



(19)
Bundesrepublik Deutschland
Deutsches Patent- und Markenamt

(10) **DE 600 30 765 T2** 2007.09.13

(12) **Übersetzung der europäischen Patentschrift**

(97) **EP 1 238 393 B1**

(21) Deutsches Aktenzeichen: **600 30 765.4**

(86) PCT-Aktenzeichen: **PCT/US00/34216**

(96) Europäisches Aktenzeichen: **00 986 483.6**

(87) PCT-Veröffentlichungs-Nr.: **WO 2001/045106**

(86) PCT-Anmeldetag: **14.12.2000**

(87) Veröffentlichungstag
der PCT-Anmeldung: **21.06.2001**

(97) Erstveröffentlichung durch das EPA: **11.09.2002**

(97) Veröffentlichungstag
der Patenterteilung beim EPA: **13.09.2006**

(47) Veröffentlichungstag im Patentblatt: **13.09.2007**

(51) Int Cl.⁸: **G11C 7/24** (2006.01)

G11C 16/22 (2006.01)

G11C 16/10 (2006.01)

(30) Unionspriorität:

465665 17.12.1999 US

(73) Patentinhaber:

Qualcomm, Inc., San Diego, Calif., US

(74) Vertreter:

**WAGNER & GEYER Partnerschaft Patent- und
Rechtsanwälte, 80538 München**

(84) Benannte Vertragsstaaten:

**AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT,
LI, LU, MC, NL, PT, SE, TR**

(72) Erfinder:

**JHA, Sanjay, San Diego, CA 92126, US;
SIMMONDS, Stephen, San Diego, CA 92130, US;
ELHUSSEINI, Jalal, Poway, CA 92064, US; YU, K.,
Nicholas, San Diego, CA 92131, US; KHAN, Safi,
San Diego, CA 92131, US**

(54) Bezeichnung: **MOBILES KOMMUNIKATIONSGERÄT MIT INTEGRIERTEM EINGEBETTETEM FLASH UND SRAM
SPEICHER**

Anmerkung: Innerhalb von neun Monaten nach der Bekanntmachung des Hinweises auf die Erteilung des europäischen Patents kann jedermann beim Europäischen Patentamt gegen das erteilte europäische Patent Einspruch einlegen. Der Einspruch ist schriftlich einzureichen und zu begründen. Er gilt erst als eingelegt, wenn die Einspruchsgebühr entrichtet worden ist (Art. 99 (1) Europäisches Patentübereinkommen).

Die Übersetzung ist gemäß Artikel II § 3 Abs. 1 IntPatÜG 1991 vom Patentinhaber eingereicht worden. Sie wurde vom Deutschen Patent- und Markenamt inhaltlich nicht geprüft.

Beschreibung**I. Gebiet der Erfindung**

[0001] Die Erfindung bezieht sich allgemein auf mobile Kommunikationsgeräte, wie beispielsweise Zellulartelefone und insbesondere auf Flash-Speicher und statische Direktzugriffsspeicher (SRAM = static random access memory) zur Verwendung darin.

II. Beschreibung der verwandten Technik

[0002] Ein Flash-Speicher ist eine Art eines nicht flüchtigen Speichers, der gespeicherte Informationen beibehält, sogar dann, wenn keine Stromversorgung vorliegt. Dies steht im Gegensatz zu flüchtigen Speichergeräten, wie beispielsweise SRAM oder dynamischem RAM (DRAM), die darin gespeicherte Daten verlieren, sobald keine Stromversorgung vorliegt. Zusätzlich dazu, dass er nicht flüchtig ist, ist ein Flash-Speicher elektrisch löschtbar und wieder- bzw. neuprogrammierbar innerhalb des Systems, in das er aufgenommen ist. Dies steht im Gegensatz zu anderen nicht flüchtigen Speichergeräten, wie beispielsweise löschbarem programmierbarem Lesespeicher (EPROM = erasable, programmable read-only memory), welcher typischerweise besondere Spannungen zur Wieder- bzw. Neuprogrammierung benötigt, und daher typischerweise nur von einem Hersteller oder Servicespezialist neu programmierbar ist.

[0003] Demgemäß wird Flash-Speicher auf vorteilhafte Weise in Geräten eingesetzt, die nicht flüchtigen Speicher benötigen, der selektiv gelöscht und neu programmiert werden kann. Insbesondere ist Flash-Speicher gut geeignet zur Verwendung in Arbeitsplatzcomputern, Laptopcomputern, Videospielsteckmodulen, digitalen Sprachaufzeichnungsgeräten, persönlichen digitalen Assistenten (PDAs) und zellularen Telefonen (oder anderen Mobilstationen oder drahtlosen Kommunikationssystemen). Typischerweise wird innerhalb eines derartigen Systems der Flash-Speicher als ein einzelnes Flash-Makro konfiguriert, in welches entweder geschrieben werden oder von dem gelesen werden kann, aber nicht beides zu einem Zeitpunkt. Für die meisten Geräte ist dies ausreichend. Beispielsweise werden innerhalb der meisten Geräte, die Flash-Speicher enthalten, jegliche Daten, die in dem Flash-Speicher gespeichert werden sollen, in dem flüchtigen Speicher gespeichert, bis das Gerät ausgeschaltet werden soll, dann wird der Flash-Speicher neu programmiert mit den Daten während eines Abschalt- bzw. Herunterfahrvorgangs. Als solches ist es unwahrscheinlich, dass das Gerät jemals zur gleichen Zeit sowohl von dem Flash-Speicher lesen als auch darauf schreiben muss. Als ein Beispiel kann ein PC oder ein Laptopcomputer Änderungen an einem Basic-Input/Output-System (BIOS) in einem DRAM-Speicher speichern, bis der Computer heruntergefahren werden

soll, dann werden Änderungen am BIOS zu dem Flash-Speicher übertragen.

[0004] Probleme entstehen jedoch beim Versuch, einen Flash-Speicher innerhalb eines Zellulartelefons zu implementieren, welches viel häufiger Lesevorgänge und Schreibvorgänge in den Flashspeicher benötigen kann. Zellulare Telefone konsumieren eine beträchtliche Menge an Leistung während der Verwendung und um kommerziell wünschenswert zu sein, muss das zellulare Telefon in der Lage sein, effektiv für lange Zeitperioden zwischen dem Wiederaufladen zu arbeiten. Daraus folgt, dass zellulare Telefone typischerweise so konfiguriert werden, dass Leistung an interne Komponenten so oft wie möglich abgeschaltet wird. In CDMA-Zellulartelefonen beispielsweise werden viele Komponenten zwischen jedem aufeinander folgenden Paging-Schlitz des CDMA-Systems abgeschaltet. (Die Paging-Schlitze treten 30 Millisekunden voneinander entfernt auf.) Demgemäß ist es nicht möglich, ausstehende Schreibvorgänge in einem nicht flüchtigen Speicher zu akkumulieren, bis zu einem einzelnen abschließenden Energieabschaltvorgang. Stattdessen müssten möglicherweise Daten, die in den Flash-Speicher geschrieben werden, sofort vor jedem folgenden zeitweisen Abschalten der Leistung. Zudem ist der Bedarf, den Flash-Speicher häufig neu zu programmieren, in einem Zellulartelefon typischerweise viel größer, insbesondere in einem zellularen Smartphone, d.h. einem Zellulartelefon, das zusammen mit einem PDA konfiguriert ist, um dadurch sowohl Zellulartelefoniefunktionen als auch PDA-Funktionen vorzusehen. Insoweit es sich um Computer handelt, kann ein Flash-Speicher nur dann neu programmiert werden müssen für den Fall, dass es Änderungen an dem BIOS oder anderen Konfigurationsparametern des Systems gibt. Bei einem zellularen Smartphone kann der Flash-Speicher häufig neu programmiert werden müssen, um neue Telefonnummern, Adressen, Kalenderdaten, Verabredungstermine und Ähnliches aufzuzeichnen. Bei Smartphones, die so konfiguriert sind, dass sie Sprachnotizen bzw. Sprachmemos aufnehmen, kann der Flash-Speicher jedes Mal dann neu programmiert werden müssen, wenn der Nutzer des Telefons ein Sprachmemo aufnimmt. Demgemäß kann es sein, dass Lese- und Schreibvorgänge viel häufiger ausgeführt werden müssen in Verbindung mit zellularen Telefonen und die herkömmliche Flash-Speicher-Anordnung, in der Lesen vom und Schreiben in den Flash-Speicher nicht gleichzeitig ausgeführt werden kann, kann unangemessen sein.

[0005] Zudem kann in Zellulartelefonanwendungen auf den Flash-Speicher viel schneller zugegriffen werden müssen, als es in anderen Anwendungen erforderlich ist. Dies ist insbesondere wahr, falls die Daten, die von dem Flash-Speicher abgerufen werden sollen, benötigt werden zur Verwendung in Verbindung mit irgendwelchen Echtzeitfunktionen des Zel-

lulartelefons, wie beispielsweise Sprachtelefonanrufen. Für solche Funktionen kann jede Verzögerung, die dadurch erforderlich wird, dass darauf gewartet werden muss, dass ein vorheriger Schreibvorgang vor dem Lesen von dem Flash-Speicher beendet worden ist, signifikant sein.

[0006] Darüber hinaus können, selbst wenn ein Lesevorgang nicht bis zur Vervollständigung eines Schreibvorgangs verzögert werden muss, Lesezeiten in herkömmlichen Flash-Speichern recht langsam sein. Flash-Speicher verschlechtern sich durch den Gebrauch, so dass die Lesezeit für spezielle Flash-Zellen, die häufig neu beschrieben worden sind, relativ langsam werden kann im Vergleich zu Flash-Zellen, die nicht häufig neu beschrieben worden sind. Daher haben, nachdem eine Vorrichtung die Flash-Speicher enthält, für einige Zeit verwendet worden ist, einige der Flash-Zellen langsamere Lesezeiten als andere. Um mögliche Verschlechterung zu berücksichtigen, stellen Vorrichtungen, die Flash-Zellen beinhalten, eine Lesezeit des internen Flash-Speichers so ein, dass sie relativ langsam ist. Diesbezüglich wird ein Bussystem, das mit dem Speicher verbunden ist, um Daten von dem Flash-Speicher abzurufen, vorprogrammiert mit einer Anzahl von Wartezuständen, die ausreichen, um die möglicherweise langsamen Zugriffszeiten von Zellen, die sich verschlechtern haben können, zu berücksichtigen. Mit anderen Worten wird das Bussystem vorprogrammiert, um das Szenario des schlimmsten Falls zu berücksichtigen, insofern als die Flash-Speicherzugriffszeit betroffen ist. Im Ergebnis sind alle Lesezugriffe relativ langsam, sogar von Flash-Speicherorten, die sich noch nicht verschlechtert haben. In vielen Geräten ist die langsame Lesezeit nicht problematisch. Jedoch ist es in Verbindung mit Zellulartelefonen viel wichtiger, die Zeit, die für jeden Zugriff von dem Flash-Speicher benötigt wird, zu minimieren, insbesondere während das Zellulartelefon in Echtzeitfunktionen eingebunden ist. Demgemäß wäre es sehr wünschenswert, ein verbessertes Flash-Speichersystem vorzusehen, insbesondere zur Verwendung in Zellulartelefonen oder ähnlichen Geräten, welche die oben dargelegten Nachteile überwinden.

[0007] Typischerweise wird in Zellulartelefonen ein Flash-Speicher in Verbindung mit einem SRAM verwendet, wobei der Flash-Speicher nicht flüchtigen Speicher vorsieht und der SRAM flüchtigen Speicher vorsieht. Typischerweise sind der Flash-Speicher und SRAM-Vorrichtungen voneinander getrennte Vorrichtungen, und sind auch getrennt von einem primären ASIC auf dem Zellulartelefon befestigt, welches den Mikroprozessor und die verschiedenen peripheren Komponenten zur Handhabung der Zellulartelefoniefunktionen beinhaltet. In solchen Ausführungen kann, weil die Flash- und die SRAM-Speicher von dem ASIC getrennt sind, die Zeit, die benötigt wird, um auf den Flash-Speicher und den SRAM zu-

zugreifen, relativ langsam sein, und dadurch die Gesamtsystemperformance behindern. Demgemäß ist es auch wünschenswert, eine verbesserte Systemarchitektur zur Verwendung in zellularen Telefonsystemen vorzusehen, die Flash-Speicher, SRAM und einen zentralen ASIC besitzen, die einen beschleunigten Zugriff auf die Flash- und SRAM-Speicher gestattet.

[0008] Auch gibt es, weil auf den Flash-Speicher des Zellulartelefons möglicherweise häufig zugegriffen werden muss, ein Risiko, dass Teile der Daten, die innerhalb des Flash-Speichers gespeichert sind, unbeabsichtigter Weise überschrieben werden können. Dies ist insbesondere der Fall bei Zellulartelefonen des Standes der Technik, die zahlreiche Hardware-Komponenten aufweisen können, die jeweils Teile des Flash-Speichers neu programmieren können. Dieses Problem wird durch die Tatsache verschlimmert, dass eine Software zur Anwendung in Zellulartelefonen typischerweise sehr schnell entwickelt und auf den Markt gebracht werden muss, um sich an den schnell verändernden Markt anzupassen. Als eine Folge gibt es ein ziemlich signifikantes Risiko, dass die Software unbeabsichtigt bewirken kann, dass Teile der Daten in dem Flash-Speicher überschrieben oder gelöscht werden. Dies kann ein schwerwiegendes Problem sein, wenn der Teil des Flash-Speichers, der unbeabsichtigt gelöscht wurde, wichtige Telefonnummern gespeichert hatte, wie beispielsweise die Polizeinummer, die Nummern der Feuerwehr und so weiter, oder wichtige Software-Programme gespeichert hatte, wie beispielsweise Boot-Load-Programme und so weiter, die für den Betrieb des Telefons erforderlich sind. Entsprechend ist es besonders wünschenswert, ein verbessertes Flash-Speichersystem vorzusehen, welches das Risiko eines unbeabsichtigten Löschs von Teilen des Flash-Speichers minimiert. Die Schrift WO 99/35650 des Standes der Technik offenbart einen Flash-Speicher mit einer Fähigkeit, während er schreibt zu lesen bzw. einer Lese-während-des-Schreibens-Fähigkeit.

ZUSAMMENFASSUNG DER ERFINDUNG

[0009] Gemäß der vorliegenden Erfindung wird ein Flash-Speichersystem gemäß Anspruch 1 vorgesehen.

[0010] Somit werden Flash-Speicherzellen in einen Satz von Flash-Makros aufgeteilt, auf die unabhängig voneinander zugegriffen werden kann. Ein Schreibvorgang kann an einem der Flash-Makros durchgeführt werden, während ein Lesevorgang in einem anderen der Flash-Makros durchgeführt wird. Dies verbessert die Gesamtsystemantwortzeit, teilweise deshalb, weil Lesevorgänge nicht aufgeschoben werden müssen, bis ein Schreibvorgang abgeschlossen ist. Tatsächlich können, da Flash-Speicherschreibvorgänge wesentlich länger dauern als Flash-Speicher-

lesevorgänge, mehrere Lesevorgänge parallel durchgeführt werden, während eines einzigen Flash-Speicherschreibvorgangs.

[0011] Gemäß einem Aspekt der Erfindung, wird ein Flash-Speichersystem vorgesehen mit programmierbaren Wartezuständen. In einem beispielhaften Ausführungsbeispiel werden Flash-Speicherzellen als ein Satz von Flash-Makros angeordnet. Ein Flash-Speicherbus verbindet die Flash-Makros und einen Flash-Speichercontroller. Der Flash-Speichercontroller beinhaltet eine Wartezustandsregistereinheit zum Speichern einer programmierbaren Anzahl von Wartezuständen, die mit den Speichermakros assoziiert sind, mit einer programmierbaren Anzahl von Wartezuständen pro Flash-Makro. Der Flash-Speichercontroller weist auch eine Flash-Makrozugriffseinheit auf zum Zugreifen auf ein ausgewähltes Flash-Makro. Die Flash-Makrozugriffseinheit greift auf das ausgewählte Flash-Makro zu, wobei sie das Flash-Speicherbusprogramm mit der Anzahl der Wartezustände, die mit dem ausgewählten Flash-Makro assoziiert sind, verwendet. Mit dieser Anordnung kann auf Teile des Flash-Speichers, welche häufig neu programmiert werden, und daher einer Verschlechterung ausgesetzt sind, zugegriffen werden unter Verwendung einer größeren Anzahl von Wartezuständen, als bei anderen Teilen des Flash-Speichers. Als ein Ergebnis kann auf Teile des Flash-Speichers, welche nicht häufig neu programmiert werden, viel schneller zugegriffen werden als in Systemen, in denen alle Leseoperationen des Flash-Speichers um eine Anzahl von Wartezuständen verzögert werden, um eine schlimmstmögliche bzw. längste Speicherzugriffszeit, entsprechend einer größtmöglichen Verschlechterung, zu berücksichtigen.

[0012] Gemäß einem Aspekt der Erfindung ist ein Flash-Speichersystem mit einem Passwortschutz vorgesehen. In einem beispielhaften Ausführungsbeispiel weist das Flash-Speichersystem Flash-Speicherzellen auf, die als ein Satz von Flash-Makros angeordnet sind, und zwar zusammen mit einer Flash-Speichersteuervorrichtung mit Mitteln zum Speichern eines getrennten Passwortes, das mit jedem Flash-Makro und den Mitteln assoziiert ist, die auf einen Empfang eines gültigen Passwortes von einem ausgewählten der Flash-Makros ansprechen, um das Programmieren zu ermöglichen, um Flash-Zellen des ausgewählten Flash-Makros zu löschen, während sie gleichzeitig eine Programmierung zum Löschen von allen anderen Flash-Zellen verhindern. Mit dieser Konfiguration wird das Risiko eines unbeabsichtigten Löschsens von den Teilen des Flash-Speichers stark minimiert. Ein gültiges Passwort wird angenommen, bevor ein Löschvorgang ausgeführt werden kann. Daher werden unbeabsichtigte Löschvorgänge, die aus Software-Fehlern und Ähnlichem resultieren, stark verringert. Darüber hin-

aus können durch das Vorsehen von unterschiedlichen Passwörtern für unterschiedliche Teile des Flash-Speichers einzelne Computerprogramme oder einzelne Peripherie-Hardware-Komponenten begrenzt werden, dass sie nur spezielle Teile des Flash-Speichers löschen. Als eine Folge werden die Risiken eines unbeabsichtigten Löschsens oder einer Neuprogrammierung von großen Teilen des Flash-Speichers stark verringert. In einer speziellen Implementierung speichert ein Teil des Flash-Speichers einen Boot-Loader zur Anwendung beim booten bzw. Hochfahren eines Systems, in dem der Flash-Speicher angeordnet ist. Ein separates Passwort, welches mit dem Boot-Loader assoziiert ist, ist vorgesehen, um weiter das Risiko einer zufälligen Neuprogrammierung des Boot-Loaders zu minimieren.

[0013] Gemäß einem Aspekt der Erfindung ist ein Flash-Speichersystem mit einer programmierbaren Memory-Map bzw. Speicherabbildung vorgesehen. Das Flash-Speichersystem weist Flash-Speicherzellen und einen Flash-Speichercontroller auf, die mit Mitteln versehen sind, um die Flash-Speicherzellen in hohe und niedrige Speicherstellen zu partitionieren. Die Flash-Speicherzellen speichern einen Boot-Loader, beginnend an einer niedrigsten Speicheradresse des Flash-Speicherraums. Die Mittel zur Partitionierung weisen die Mittel zum Umlagern bzw. Swapping der hohen und niedrigen Speicherstellen auf, nachdem Betriebsvorgänge, die von dem Boot-Loader ausgeführt worden sind, vollendet worden sind. Mit dieser Anordnung wird der Boot-Loader, der typischerweise nur während eines anfänglichen Einschaltvorgangs aufgerufen wird, in den hohen Speicher umgelagert, so dass andere Daten und Programme, die häufiger aufgerufen werden müssen, in dem unteren Speicher gespeichert werden können, wo sie zweckmäßiger aufgerufen werden können.

[0014] Andere Ziele, Merkmale und Vorteile der Erfindung werden aus der folgenden Beschreibung in Verbindung mit den beigefügten Zeichnungen offensichtlich.

KURZE BESCHREIBUNG DER ZEICHNUNGEN

[0015] [Fig. 1](#) ist ein Blockdiagramm eines Sprach- und Daten-Modem-ASIC mit einem integrierten eingebetteten Flash- und SRAM-Speichersystem zur Verwendung in einem Mobiltelefon.

[0016] [Fig. 2](#) ist ein Blockdiagramm eines Flash-Speichersystems des ASIC der [Fig. 1](#).

[0017] [Fig. 3](#) ist ein Blockdiagramm eines Wartezustandsspeichers zur Anwendung in dem Flash-Speichersystem der [Fig. 2](#).

[0018] [Fig. 4](#) ist ein Blockdiagramm eines Pass-

wortspeichers zur Anwendung in dem Flash-Speichersystem der [Fig. 2](#).

[0019] [Fig. 5](#) ist ein Blockdiagramm eines Flash-Speicherstatusregisters zur Anwendung bei dem Flash-Speichersystem der [Fig. 2](#).

[0020] [Fig. 6](#) ist ein Blockdiagramm eines partitionierbaren Speicherraums des Flash-Speichers der [Fig. 2](#).

DETAILLIERTE BESCHREIBUNG VON BEISPIELHAFTEN ASPEKTEN DER ERFINDUNG

[0021] Mit Bezug auf die Figuren werden nun bevorzugte beispielhafte Aspekte der Erfindung beschrieben. Die Erfindung wird in erster Linie mit Bezug auf ein integriertes eingebettetes Flash- und SRAM-System zur Anwendung in einem Sprach- und Daten-Modem-ASIC eines CDMA-Zellulartelefons beschrieben. Jedoch sind die Prinzipien der Erfindung genauso auf andere Systeme anwendbar.

[0022] [Fig. 1](#) veranschaulicht einen Sprach- und Daten-Modem-ASIC **100** zur Anwendung in einer mobilen drahtlosen Kommunikationsvorrichtung, wie beispielsweise einem Zellulartelefon, der zur Anwendung in einem drahtlosen CDMA-Kommunikationssystem konfiguriert ist. Der Sprach- und Daten-Modem-ASIC weist eine Schaltung auf, zum Handhaben der Telefonfunktionen des Zellulartelefons. Obwohl nicht gezeigt, kann das Zellulartelefon andere ASICs oder andere integrierte Schaltungen aufweisen, die konfiguriert sind, um andere Funktionen auszuführen. Wenn beispielsweise das Zellulartelefon als ein Smartphone konfiguriert ist, um PDA-Funktionen genauso wie Drahtlostelefonfunktionen vorzusehen, kann ein separater ASIC vorgesehen sein, um die PDA-Funktionen zu steuern. Alternativ können alle Funktionen in einem einzigen ASIC integriert sein.

[0023] Um die Drahtlostelefonfunktionen des Zellulartelefons handzuhaben, weist der ASIC einen Mikroprozessor **102** auf, um die Sprach- und Daten-Modem-Funktionen zu steuern. Der Mikroprozessor kann beispielsweise ein Mikroprozessor mit reduziertem Anweisungssatz (RISC-Mikroprozessor, RISC = reduced instruction set computing) sein, wie beispielsweise der ARM 7TDMI-Mikroprozessor, der von der Arm, Inc. geliefert wird, wobei ARM 7TDMI und Arm, Inc. beides Handelsmarken der Arm, Inc. sind. Bei anderen Einrichtungen werden andere Mikroprozessoren eingesetzt, die beispielsweise Mikroprozessoren mit komplexem Anweisungssatz (CISC-Mikroprozessoren, CISC = complex instruction set computing) aufweisen. Verschiedene Peripheriekomponenten, die im Allgemeinen mit **104** bezeichnet werden, sind in dem ASIC vorgesehen, um spezifische CDMA-Drahtlostelefonfunktionen auszuführen.

[0024] Ein Systembus **106** verbindet den Mikroprozessor und die verschiedenen CDMA-Peripheriekomponenten. Bei der Anwendung steuert der Mikroprozessor die verschiedenen CDMA-Peripheriekomponenten über den Systembus, um verschiedene Funktionen auszuführen, die auf die Verarbeitung von CDMA-Drahtloskommunikationen gerichtet sind, wie beispielsweise die Konvertierung von CDMA-Signalen, die von einer (nicht gezeigten) Basisstation empfangen werden, in Sprachsignale zur Ausgabe durch einen Lautsprecher des Zellulartelefons, oder die Umwandlung von Sprachsignalen, die von einem Mikrofon des Zellulartelefons empfangen wurden, in CDMA-Signale zur Übertragung an die Basisstation. Um diese und andere Funktionen auszuführen, speichern der Mikroprozessor und die Peripheriekomponenten Daten oder andere Informationen entweder in einem internen Speichersystem **108**, welches auf dem ASIC ausgebildet ist, oder in einem externen Speichersystem **110**, welches einen oder mehrere SRAM-, DRAM- oder Flash-Speicherchips aufweisen kann, die in dem Zellulartelefon außerhalb des ASIC angeordnet sind. Im Allgemeinen sind Daten oder andere Informationen, die schnell aufgerufen werden müssen, wie beispielsweise Daten, die in Verbindung mit einer Echtzeitverarbeitung von Telefonanrufen und Ähnlichem verwendet werden, in dem internen Speichersystem für einen zweckmäßigen Zugriff gespeichert. Daten, die nicht so schnell aufgerufen werden müssen, wie beispielsweise Daten zur Anwendung in Nicht-Echtzeitfunktionen, sind in dem externen Speichersystem gespeichert.

[0025] Das interne Speichersystem weist ein Flash-Speichersystem **112** und einen SRAM **113** integriert als ein einzelnes Speichersystem auf. Bei der beispielhaften Implementierung der [Fig. 1](#) sind die Schaltungen des einzelnen Speichersystems und die Schaltungen des Mikroprozessors und der CDMA-Peripheriekomponenten alle auf einem einzigen Silizium-Die integriert. Bei anderen Anwendungen sind die Schaltungen des einzelnen Speichersystems auf einem Die vorgesehen, und der Mikroprozessor und die CDMA-Peripheriekomponenten sind auf einem anderen Die vorgesehen. Vorzugsweise sind bei diesem Aufbau die zwei getrennten Silizium-Dies in einem einzelnen Chip integriert. Auf jeden Fall verbindet eine interne Speicherschnittstelleneinheit **115** das interne Speichersystem mit dem Mikroprozessor und den CDMA-Peripheriekomponenten über den Systembus **106**. Eine externe Speicherschnittstelleneinheit **117** verbindet das externe Speichersystem mit dem Mikroprozessor und den CDMA-Peripheriekomponenten auch über den Systembus **106**.

[0026] Der Flash-Speicher des Flash-Speichersystems **112** des internen Speichersystems wird in erster Linie zur Speicherung von CDMA-Softwarecode zur Anwendung durch den Mikroprozessor oder die CD-

MA-Peripheriekomponenten verwendet. Typischerweise weist der Flash-Speicher auch einen Boot-Loader auf, der während eines anfänglichen Einschaltvorgangs vom Mikroprozessor aufgerufen und laufen gelassen wird. Der Boot-Loader weist Anweisungen auf, um auf andere CDMA-Programme zuzugreifen und sie auszuführen, die in dem Flash-Speicher gespeichert sind. Der Flash-Speicher wird auch eingesetzt, um eine nicht flüchtige Speicherung von Daten vorzusehen, die in Verbindung mit dem Zellulartelefon verwendet werden, wie beispielsweise die Speicherung von Namen, Telefonnummern, Adressen und Ähnlichem. Wenn das Zellulartelefon als ein Smartphone konfiguriert ist, um zusätzlich die Betriebsvorgänge eines PDA auszuführen, dann speichert der Flash-Speicher zusätzlich Daten, die vom dem PDA verwendet werden, wie beispielsweise Verabredungsdaten, Kalender, Zeitpläne, Sprachmemos und so weiter. Im Allgemeinen wird irgendeine Art von Software oder Daten, die gespeichert bleiben muss, auch während das Zellulartelefon vollständig abgeschaltet ist, vorzugsweise in entweder dem Flash-Speichersystem gehalten, welches in dem ASIC eingebettet ist, oder in dem Flash-Speichersystem, falls vorhanden, welches in dem externen Speicher vorgesehen ist.

[0027] Der eingebettete SRAM **113** sieht eine flüchtige Speicherung von anderen Arten von Daten oder Softwareprogrammen vor, die vom Mikroprozessor oder den Peripheriekomponenten in Verbindung mit ihren Funktionen eingesetzt werden. Beispielsweise, wenn CDMA-Signale von einer Basisstation während einer Zellulartelefonunterhaltung empfangen werden, können die verschiedenen Peripheriekomponenten die Signale, oder verarbeitete Versionen davon, in dem SRAM speichern. Als ein spezielles Beispiel werden, nachdem der Viterbi-Decodierer die Eingangssignale verarbeitet hat, die decodierten Symbole für Sprache in dem SRAM zur folgenden Verarbeitung durch den Vocoder gespeichert. Der externe Speicher kann auch als ein flüchtiger Speicher zur Speicherung von Daten eingesetzt werden, insbesondere von Daten, auf die nicht mit hohen Raten zugegriffen werden muss.

[0028] Nun unter kurzer Betrachtung der CDMA-Peripheriekomponenten des ASIC **100** ist für die Übertragung von Signalen ein Vocoder **114** vorgesehen, der einen DSP zur Umwandlung von Sprachsignalen aufweisen kann, die durch ein (nicht gezeigtes) Mikrofon aufgenommen wurden, und zwar in digitalisierte Symbole oder andere Pakete von Information. Ein CDMA-Codierer **116** codiert die Symbole, die von dem Vocoder erzeugt wurden, zur Fehlerkorrektur und Detektierungszwecken. Ein CDMA-Interleaver bzw. -Verschachteler **118** verschachtelt die codierten Signale, um Zeit-Diversity vorzusehen, um dadurch eine Verringerung der Sendeleistung zu gestatten. Ein CDMA-Modulator **120** moduliert die verschach-

telten Signale zur darauf folgenden Sendung über eine Antenne (nicht gezeigt). Für die Verarbeitung von empfangenen Signalen demoduliert ein CDMA-Demodulator **122** die Signale, ein Entschachteler **124** entschachtelt die Signale, um den Effekt einer vorherigen Verschachtelung zu entfernen, und ein CDMA-Decodierer **126** decodiert Signale, um Sprach- oder Datensignale extrahieren, die darin codiert sind. Für Sprachkommunikation werden die decodierten Sprachsignale durch einen (nicht gezeigten) Lautsprecher für den Nutzer des Telefons ausgegeben. Für Datenkommunikationsvorgänge werden die decodierten Datensignale weiter durch andere Komponenten des Telefons verarbeitet, wie beispielsweise für eine Anzeige (auf einer nicht gezeigten Anzeige) unter Verwendung eines Webbrowserprogramms, eines Emailprogramms oder durch Ähnliches.

[0029] Somit veranschaulicht [Fig. 1](#) neben weiteren Merkmalen einen ASIC mit einem eingebetteten Flash und einem SRAM. Durch Einbettung des Flash-Speichers und des SRAM auf dem ASIC sind der Mikroprozessor und der Flash- und SRAM-Speicher dadurch physisch nahe aneinander. Daher erfahren Datenbus- und Steuersignale vom Mikroprozessor zum Speicher und umgekehrt, weniger Weiterleitungs- bzw. Routing- und Kapazitätsverzögerungen, wie dies auf gedruckten Leiterplatten zu finden ist, wenn Off-Chip-Speicher oder nicht auf dem Chip vorhandener Speicher verwendet wird. Dies verbessert die Gesamtspeicherzugriffszeit und sieht schnellere Gesamtverarbeitungsgeschwindigkeiten vor. Auch kann durch Einbettung des Flash und des SRAM in dem ASIC die Gesamtgröße der Elektronik des Zellulartelefons verringert werden, und zwar im Vergleich zu Systemen, die erfordern, dass der gesamte Speicher außerhalb des ASIC angeordnet ist. Dies gestattet, dass das Zellulartelefon unter Verwendung einer schlanken und kompakten Konstruktion konfiguriert wird. Darüber hinaus werden signifikante Leistungseinsparungen erreicht durch Einbettung des Flash-Speichers und des SRAM in dem ASIC. Insbesondere wenn der Flash-Speicher und der SRAM außerhalb des Chips vorgesehen sind, wird beträchtliche Leistung und Energie bei der Umschaltung von verschiedenen Eingabe/Ausgabe-Pins verbraucht, die erforderlich sind, um auf den Flash-Speicher und den SRAM zuzugreifen, und für Signalbahnen auf der gedruckten Leiterplatte. Hier wird durch Speicherung von Daten und Programmen, die oft aufgerufen werden müssen, in dem eingebetteten Flash-Speicher und dem SRAM der Leistungsverbrauch verringert, wodurch die Standby-Zeit des Zellulartelefons verbessert wird, so dass das Telefon länger durch den Kunden betrieben werden kann, bevor ein Aufladungsvorgang erforderlich ist. Es gibt auch einen niedrigeren Gesamtstromverbrauch, wenn die Vorrichtungen aktiv sind, wodurch weiter der Leistungsverbrauch verringert wird. Vorzugswei-

se sind der Systembus, der Mikroprozessor und das interne Speichersystem alle konfiguriert, um 32-Bit-Datenpfade zu verarbeiten. Der externe Flash und der SRAM werden typischerweise unter Verwendung von nur einem 8-Bit-Datenpfad aufgerufen. Dadurch werden durch Aufnahme eines 32-Bit-Datenpfades Zugriffszeiten ebenfalls verbessert. Darüber hinaus können, weil eine größere Datenmenge während jedes Taktzyklus verarbeitet werden kann, die Gesamttaktraten der Komponenten des ASIC reduziert werden, wodurch noch weiter der Leistungsverbrauch verbessert wird. Für den Speicher außerhalb des Chips wird typischerweise kein 32-Bit-Datenpfad verwendet, und zwar wegen Pin- und Weiterleitungsbeschränkungen. Als noch ein weiterer Vorteil reduziert das Vorsehen eines eingebetteten Flash-Speichers und eines SRAM in dem ASIC die Gesamtanzahl der Chips, die auf dem Motherboard bzw. der Hauptplatine des Zellulartelefons erforderlich sind, und zwar durch Erhöhen der Zuverlässigkeit als eine Folge von vereinfachten Herstellungsprozessen.

[0030] Somit bietet das Vorsehen eines eingebetteten Flash- und SRAM-Speichers in dem ASIC **100** zahlreiche Vorteile. Im Folgenden werden spezielle Merkmale der Flash- und SRAM-Speicher des internen Speichersystems genauer beschrieben. Diese Merkmale sehen beispielsweise eine verbesserte Zugriffszeit, einen verbesserten Schutz vor unbeabsichtigten Überschreibvorgängen oder Ähnliches vor.

[0031] Nun mit Bezug auf [Fig. 2](#) weist das eingebettete Flash-Speichersystem eine Flash-Speicheranordnung **130** und einen Flash-Speichercontroller **132** auf. Die Flash-Speicheranordnung **130** weist Flash-Speicherzellen auf, die bis zu N unterschiedlichen Flash-Makros definieren, auf die, wie beschrieben wird, unabhängig zugegriffen werden kann. Anders gesagt, kann ein Lesevorgang an einem der Flash-Makros ausgeführt werden, während ein Schreibvorgang an einem anderen der Flash-Makros ausgeführt wird. In einem speziellen Beispiel ist N auf 4 gesetzt, was gestattet, dass der Flash-Speicher in eine Gesamtzahl von vier getrennten Flash-Makros unterteilt werden kann. Ein Flash-Speichersystembus **134** verbindet die Flash-Speicheranordnung und den Flash-Controller, was gestattet, dass der Flash-Controller auf die Flash-Speicheranordnung zugreift, um Lese- oder Schreibvorgänge auszuführen. Vorzugsweise sind alle drei Komponenten konfiguriert, um 32-Bit-Datenpfade aufzunehmen. Der Flash-Controller **132** weist eine Flash-Makrozugriffseinheit **136** und verschiedene Speicherregister auf, einschließlich eines programmierbaren Wartezustandsregisters **138**, eines programmierbaren Passwortregisters **140** und eines Statusregisters **142**. Die Flash-Speicherzugriffseinheit verwendet Information, die in den verschiedenen Registern gespeichert ist, um einen Zugriff auf die Flash-Makros der Flash-Speicheranordnung zu steuern. Die Informati-

on, die in den Registern gespeichert ist, und die Weise, in der die Informationen verwendet werden, werden genauer unten beschrieben. Die Flash-Speicherzugriffseinheit **136** weist eine Flash-Speicher-Leseeinheit **144** und eine Flash-Speicher-Schreibeinheit **146** auf. Die Leseeinheit wird verwendet, um Daten oder Software-Programme zu lesen, die in der Flash-Speichereinheit gespeichert sind. Die Schreibeinheit wird zum Neuprogrammieren von Teilen des Flash-Speichers verwendet, um neue Daten oder neue Programme zu speichern. Zusammen sehen die Flash-Lese- und Schreibeinheiten Lesen-während-des-Schreibens-Mittel vor, um zu gestatten, dass ein Lesevorgang auf einem der Flash-Makros ausgeführt wird, während ein Schreibvorgang simultan an einem anderen der Flash-Makros ausgeführt wird. Somit kann, als ein spezielles Beispiel, die Leseeinheit **144** einen Lesevorgang an dem Flash-Makro #1 ausführen, während die Schreibeinheit einen Schreibvorgang an dem Flash-Makro #2 ausführt. Ein Lesevorgang wird ausgeführt durch Weiterleiten eines Lesebefehls über den Flash-Bus **134** zu der Flash-Speicheranordnung **130**, wobei eine spezielle Speicheradresse in der Gesamtspeicheranordnung festgelegt wird. Im Fall eines Lesevorgangs von Flash-Makro #1 wird die Adresse eine 32-Bit-Speicheradresse in den Adressräumen des Flash-Makros #1 spezifizieren. Genauso wird für einen Schreibvorgang ein Schreibbefehl über den Flash-Bus zum Flash-Speicher weitergeleitet. Der Schreibbefehl spezifiziert die Adresse in dem Flash-Makro und spezifiziert zusätzlich die Daten, die darin gespeichert werden sollen. Für eine 32-Bit-Datenpfadimplementierung wird der Schreibbefehl 32-Bit-Daten entlang der Speicheradresse vorsehen. Auch, wie unten beschrieben wird, wird ein Schreibvorgang nur ausgeführt, wenn ein korrektes Passwort von dem Mikroprozessor oder der Peripheriekomponente als Teil des Schreibbefehls empfangen wird.

[0032] Flash-Speicherschreibvorgänge brauchen eine wesentlich größere Zeitdauer als ein Flash-Lesevorgang. Ein Lesevorgang kann beispielsweise in nur einem einzigen Taktzyklus ausgeführt werden. Ein Schreibvorgang kann Hunderte von Taktzyklen erfordern. Entsprechend kann eine sehr große Anzahl von einzelnen Lesevorgängen ausgeführt werden, während ein einziger Schreibvorgang ausgeführt wird. Lesevorgänge werden nur an irgendwelchen anderen Flash-Makros ausgeführt als an dem Makro, das dem Schreibvorgang unterworfen ist. Wenn eine Leseanforderung vom Mikroprozessor empfangen wird, die eine Adresse in dem Flash-Makro spezifiziert, das gegenwärtig einem Schreibvorgang unterworfen ist, wird der Lesevorgang von der Flash-Makrozugriffseinheit verzögert, bis der Schreibvorgang vollendet worden ist. Zusätzlich liefert die Flash-Speicherzugriffseinheit ein Signal an den Mikroprozessor, was den Mikroprozessor steuert, um den Betrieb anzuhalten, und zwar wartend auf

die Vollendung des Schreibvorgangs. Dies kann unter Verwendung eines Interrupt- bzw. Unterbrechungssignals oder einer anderen herkömmlichen Technik erreicht werden. In anderen Implementierungen, wie beispielsweise bei einer Implementierung, die einen Mikroprozessor einsetzt, der eine spekulative Verarbeitung bzw. Speculative Processing ausführen kann, kann es Wünschenswert sein, zu gestatten, dass der Mikroprozessor weiter arbeitet, während der Schreibvorgang ausgeführt wird.

[0033] Daher wird ein Lesen-während-des-Schreibens-Mechanismus vorgesehen. Der Lesen-während-des-Schreibens-Mechanismus gestattet, dass schnellere Lesezugriffszeiten erreicht werden, im Durchschnitt, weil Lesevorgänge nicht bis zur Vollen- dung eines laufenden Schreibvorgangs verzögert werden müssen, außer wenn sie an das gleiche Flash-Makro ausgegeben werden. Als solches ist das Gesamtsystem effizienter, wobei entweder ge- stattet wird, dass das System mehr Vorgänge in einer gegebenen Zeitperiode ausführt, oder gestattet wird, dass das System mit einer im Allgemeinen niedrige- ren Taktrate laufen gelassen wird, was somit den Leistungsverbrauch reduziert.

[0034] Die Flash-Speicherzugriffseinheit weist zu- sätzlich eine Löscheinheit **147** auf, um Teile des Flash-Speichers zu löschen. Die Löscheinheit nimmt einen Löschbefehl vom Mikroprozessor oder von ei- ner anderen Komponente auf, der eine Anfangsa- dresse festlegt, und der weiter die Datenmenge fest- legt, die zu löschen ist. Das Löschen des Befehls kann das Löschen von entweder einem einzigen Wort (32 Bits), eines gesamten Pages bzw. einer ge- samten Seite oder einer gesamten Flash-Speichera- nordnung festlegen. Wie bei Schreibvorgängen wird ein Löschvorgang nur ausgeführt, wenn ein korrektes Passwort vom Mikroprozessor oder von einer Peri- pheriekomponente als Teil des Löschbefehls emp- fangen wird. Während eines Betriebsvorgangs, bei dem die gesamte Flash-Speicheranordnung gelöscht wird, was hier als "Massenlöschvorgang" bezeichnet wird, werden alle anderen Lese- oder Schreibanfor- derungen an den Flash-Speicher angehalten, und zwar beim Warten auf die Vollendung des Massen- löschvorgangs. Bei dem beispielhaften Aufbau wer- den auch alle Flash-Speichervorgänge angehalten, wenn eine Seite des Flash-Speichers gelöscht wird, auch wenn der Betrieb als ein Flash-Makro getrennt von der gerade gelöschten Seite ausgeführt wird. In anderen Implementierungen sind Flash-Speichervor- gänge bezüglich anderen Flash-Makros gestattet, die nicht von der gerade gelöschten Seite beeinflusst werden. Auch kann abhängig von der speziellen Im- plementierung des Flash-Speichers ein Löschvor- gang vor irgendeinem Schreib-(oder Programmie- rungs-)Vorgang ausgeführt werden. Insbesondere wenn der Flash-Speicher so konfiguriert ist, dass ein Schreibvorgang nur bewirken kann, dass sich ein

"Eins"-Bit in ein "Null"-Bit verändert, dann wird ein Löschvorgang an dem Wort oder an den Worten aus- geführt werden müssen, die geschrieben werden, und zwar vor dem tatsächlichen Schreibvorgang. Der Löschvorgang wandelt alle Bits, die zuvor auf "Null" programmiert waren, zurück auf "Eins", so dass der darauf folgende Schreibvorgang ausgewählte Bits in dem Wort auf "Null" umprogrammieren kann. Ent- sprechend steuert bei einer solchen Implementierung die Flash-Speicherschreibereinheit **146** beim Empfang eines Schreibbefehls zuerst die Löscheinheit **147**, um das Wort oder die Worte zu löschen, die dem Schreibvorgang unterworfen sind, bevor der Schreib- vorgang ausgeführt wird. In einer Alternative führt die Schreibereinheit **146** nicht automatisch einen Lösch- vorgang vor einem Schreibvorgang aus. Bei einer solchen Implementierung müssen die Software-Pro- grammierer sicherstellen, dass die Software, die auf den Flash-Speicher zugreift, Löschvorgänge vor Schreibvorgängen ausführt.

[0035] Die Flash-Speicherzugriffseinheit weist auch eine Speicherumlagerungseinheit bzw. Speicherswa- peinheit **149** zur Anwendung bei der Umlagerung von hohen und niedrigen Speicherteilen der Gesamtspei- cheradressphase der Speicheranordnung **133** auf. Der Betrieb der Speicherumlagerungseinheit wird ge- nauer unten in Verbindung mit [Fig. 5](#) beschrieben.

[0036] [Fig. 3](#) veranschaulicht ein Wartezustand- speicherregister **138** für Flash-Bus-Wartezustände zur Anwendung mit unterschiedlichen Teilen der Flash-Speicheranordnung. Das Wartezustandspei- cherregister weist M getrennte Speicherregister zum Speichern von Wartezustandswerten auf, die mit M getrennten Teilen der Flash-Speicheranordnung as- soziiert sind. Bei einer Implementierung ist M gleich N gesetzt, so dass es ein getrenntes Wartezustand- register pro Flash-Makro gibt. Bei anderen Imple- mentierungen kann es mehrere Wartezustandregis- ter pro Flash-Makro oder mehrere Flash-Makros pro Wartezustandregister geben. Bei einer speziellen Im- plementierung ist N auf 4 gesetzt und M ist auf 16 ge- setzt. Jedes Wartezustandregister speichert einen Wartezustandwert zur Anwendung während Lesevor- gängen in irgendeine Speicheradresse in einem ent- sprechenden Teil der Speicheranordnung. Der War- tezustand legt die Anzahl der Zyklen fest, für die der Flash-Speicherbus der Flash-Speicherzugriffseinheit bis zum Abruf von Daten über einen Lesevorgang warten muss. Wenn der Wartezustand auf Null ge- setzt ist, dann greift die Flash-Speicherzugriffseinheit auf den Flash-Bus während des Taktzyklus sofort fol- gend auf einen Taktzyklus zu, in dem ein Lesevor- gang an der Flash-Anordnung vorgebracht wurde. Wenn der Wartezustand auf 3 gesetzt ist, dann wartet die Flash-Speicherzugriffseinheit drei Takte nachfol- gend auf das Vorbringen eines Lesevorgangs.

[0037] Unterschiedliche Wartezustandswerte kön-

nen für unterschiedliche Teile der Flash-Speicheranordnung vorgesehen werden, um unterschiedliche Grade der Verschlechterung der Flash-Speicheranordnungen zu berücksichtigen. Kurz gesagt, je öfter eine Flash-Zelle beschrieben wird, desto langsamer spricht die Flash-Zelle auf darauf folgende Lesevorgänge an. Wenn unterschiedliche Teile der Flash-Anordnung unterschiedliche Arten von Daten oder Programmen speichern, treten unterschiedliche Grade der Lesezeitverschlechterung auf. Insbesondere Teile des Flash-Speichers, die oft neu beschrieben werden, wie beispielsweise Teile, die verwendet werden, um Sprachmemos zu speichern, sind einem größeren Ausmaß an Verschlechterung unterworfen als Teile des Flash-Speichers, die nicht oft wiederbeschrieben werden, wie beispielsweise Teile, die den Boot-Loader speichern. (In einer Implementierung, die unten beschrieben wird, wird der Boot-Loader in den hohen Speicher und aus diesem heraus umgelagert, was einen gewissen Grad an Verschlechterung der Speicherzellen zur Folge hat, die zum Speichern des Boot-Loaders verwendet werden).

[0038] Somit wird ein getrennter Wartezustandswert in dem Wartezustandsregister für jeden Teil des Speichers abhängig vom Grad der erwarteten Verschlechterung von diesem Speicher gespeichert. Für Teile des Speichers, vor denen erwartet wird, dass sie nur wenig häufig wieder programmiert werden, ist ein Wartezustand von Null in dem entsprechenden Wartezustandsspeicherregister vorprogrammiert. Für Teile der Anordnung, bei denen erwartet wird, dass sie oft neu programmiert werden, ist vorzugsweise ein Wartezustandswert von 3 vorprogrammiert. Das Ausmaß der erneuten Programmierung, welches erwartet wird, wird zuvor basierend auf der gesamten Systemhardware- und Software-Konstruktion bestimmt. Wenn gewisse Hardware- und Software-Komponenten, die oft in den Flash-Speicher schreiben, wie beispielsweise Sprachnachrichtenkomponten, ausgelegt sind, um nur das Flash-Makro #3 zu verwenden, dann ist somit der Wartezustand für das Flash-Makro #3 auf 3 voreingestellt. Alternativ ist ein Wartezustand von 0 für das Flash-Makro #1 voreingestellt; ein Wartezustand von 1 ist für das Flash-Makro #2 voreingestellt; ein Wartezustand von 2 ist für das Flash-Makro #3 voreingestellt, und so weiter. Software- und Hardwaredesigner sind angewiesen, Systeme zum Schreiben in die Flash-Makros basierend auf der erwarteten Schreibfrequenz zu konstruieren. Konstrukteure, die Hardware oder Software entwickeln, die häufige Schreibvorgängen in den Flash-Speicher ausführen, werden ihre Systeme so konfigurieren, dass sie in das Flash-Makro #4 schreiben. Konstrukteure, die Hardware oder Software entwickeln, die nur selten in den Flash-Speicher schreiben, werden ihre Systeme konfigurieren, so dass sie in das Flash-Makro #1 schreiben. Als noch eine weitere Alternative ist die Flash-Speicherzugriffseinheit konfiguriert, um die An-

zahl der Schreibvorgänge zu verfolgen, die an jedem Flash-Makro ausgeführt wird, und selektiv die Anzahl der assoziierten Wartezustände zu steigern. Die Wartezustandsregister sind alle auf Null voreingestellt, wenn das Zellulartelefon anfänglich hergestellt wird. Danach werden, abhängig von dem Ausmaß, in dem die unterschiedlichen Flash-Makros während der Anwendung des Zellulartelefons neu beschrieben werden, die einzelnen Werte, die in den Wartezustandsregistern gespeichert sind, selektiv auf 1 zurückgesetzt bzw. neugesetzt, dann vielleicht auf 2, dann auf 3 oder mehr. Abhängig von dem speziellen verwendeten Flash-Speicher und vom Ausmaß der Anwendung des Zellulartelefons kann es ein Jahr oder zwei dauern, dass irgendeines der Wartezustandsregister von Null auf 1 neu gesetzt wird und keines kann jemals einen Wartezustandswert von 3 erreichen. In anderen Fällen können die Werte öfter neu gesetzt werden müssen. Bei noch weiteren Anwendungen ist eine noch größere Anzahl von Wartezustandswerten vorgesehen, beispielsweise Wartezustandswerte von Null bis 10. Die Gesamtanzahl der Wartezustandswerte und die Anzahl der Neuprogrammierungsvorgänge, die ausgeführt werden, bevor die Anzahl der Wartezustände vergrößert wird, hängt stark von der speziellen Flash-Speicheranordnung, vom Flash-Bus und vom Flash-Controller ab, genauso wie die Gesamtfrequenz des Taktes, der verwendet wird, um die verschiedenen Komponenten zu steuern. Für jede Auswahl von Komponenten und für jede Auswahl eines Taktsignals kann jedoch ein Routineexperiment ausgeführt werden, um akzeptable oder optimale Werte für die Wartezustandswerte zu bestimmen, und um zu bestimmen, wie oft die Wartezustandswerte inkrementiert werden müssen. Auf jeden Fall kann durch Vorsehen von programmierbaren Wartezustandswerten eine durchschnittliche Speicherzugriffszeit verringert werden, wodurch entweder eine größere Verarbeitungsgeschwindigkeit vorgesehen wird oder die Anwendung eines langsameren Taktes gestattet wird, um ähnliche Verarbeitungsgeschwindigkeiten zu erreichen.

[0039] [Fig. 4](#) veranschaulicht ein Flash-Speicherpasswortregister **140**, um Passworte zu speichern, die mit unterschiedlichen Teilen der Flash-Speicheranordnung assoziiert sind. Das Flash-Speicherpasswortregister speichert P Passworte, jeweils eines assoziiert mit einem entsprechenden Teil der Flash-Speicheranordnung. In einem Beispiel ist P auf 16 eingestellt. Zur Vereinfachung der Programmierung kann P auf den gleichen Wert gesetzt werden wie M. Jedes einzelne Passwortregister speichert ein einzigartiges Passwort, das erforderlich ist, bevor ein Schreib- oder Löschvorgang an dem entsprechenden Teil der Speicheranordnung ausgeführt werden kann. Somit legt jeder Schreibbefehl, der vom Mikroprozessor oder von einer anderen Komponente des ASIC empfangen wurde, ein Passwort fest, und zwar zusammen mit einer Speicheradresse und den Da-

ten, die auf diese Adresse geschrieben werden sollen. Jeder Löschbefehl legt auch ein Passwort fest, und zwar zusammen mit einer Identifikation der zu löschenden Speicheradressen. Auf den Empfang eines Schreibbefehls hin vergleicht die Schreibeinheit **146** ([Fig. 2](#)) das empfangene Passwort in dem Schreibbefehl mit dem Passwort in dem Passwortregister für die Speicheradresse, die von dem Schreibbefehl festgelegt wurde. Wenn die Passwörter übereinstimmen, geht die Schreibeinheit weiter voran, um den Schreibbefehl auszuführen, und zwar durch Liefern des Schreibbefehls an den Flash-Speicher. Wenn die Passwörter nicht übereinstimmen, liefert die Schreibeinheit ein Fehlersignal zurück zu der Komponente, die den Schreibbefehl ausgibt, wobei sie die Komponente informiert, dass der Schreibbefehl wegen einem ungültigen Passwort nicht ausgeführt werden konnte. Löschbefehle werden in der gleichen Weise behandelt.

[0040] Somit wird ein Passwortschutz vorgesehen. Ein Passwortschutz wird teilweise vorgesehen, um ein unbeabsichtigtes Löschen oder eine Neuprogrammierung von Teilen des Flash-Speichers vielleicht als ein Ergebnis eines Softwarefehlers oder von Ähnlichem zu verhindern. In dieser Hinsicht ist es ziemlich unwahrscheinlich, dass ein unbeabsichtigter Schreib- oder Löschbefehl das korrekte Passwort aufweisen könnte. Wenn das Gesamtsystem so konfiguriert ist, dass unterschiedliche Peripheriekomponenten aus unterschiedlichen Teilen des Flash-Speichers lesen und dorthin schreiben, hilft dann der Passwortschutz auch dabei, zu verhindern, dass eine Peripheriekomponente unbeabsichtigt in einen Speicherbereich schreibt, der für eine andere Komponente reserviert ist. Beispielsweise kann der Vocoder nur vom Flash-Makro #2 lesen oder auf dieses schreiben. Andere Passwortprotokolle können alternativ vorgesehen werden. Anstatt ein anderes Passwort mit unterschiedlichen Teilen des Speichers zu assoziieren, können beispielsweise unterschiedliche Passwörter direkt unterschiedlichen Ausgabekomponenten zugeordnet sein. Beispielsweise kann somit ein erstes Passwort mit Befehlen assoziiert sein, die vom Mikroprozessor empfangen wurden, während ein zweites Passwort mit Befehlen assoziiert ist, die vom Vocoder empfangen wurden. Bei dieser Implementierung kann jede Vorrichtung potentiell aus irgendeinem der Flash-Makros lesen oder in irgendeines schreiben, jedoch wird trotzdem das Risiko von unbeabsichtigten Schreibvorgängen minimiert. Jedes Passwort kann beispielsweise eine einzigartige binäre Sequenz von 16 Bits sein. Durch Vorsehen eines Passwortschutzes wird ein Verlust von wichtigen Daten vermieden, wie beispielsweise der Verlust von Telefonnummern für die Polizei, die Feuerwehr oder Ähnliches. Auch könnte das Risiko, dass Teile des CDMA-Codes, die in dem Flash-Speicher gespeichert sind, fehlerbehaftet werden, verringert werden. In einem extremen Fall könnte die Störung von einem

oder von mehreren der CDMA-Programme, die in dem Flash-Speicher gespeichert sind, zu einem Versagen des gesamten Zellulartelefons führen, was erfordert, dass der Kunde das Zellulartelefon zum Ersatz oder zur Reparatur zurückbringt. In noch weiteren Implementierungen kann der Nutzer spezielle Passwörter zur Verwendung mit Daten aufzeichnen, die vom Nutzer gespeichert werden. Wenn beispielsweise eines der Flash-Makros ausschließlich zur Speicherung von Sprachnachrichten verwendet wird, kann der Nutzer ein einzigartiges Passwort über die Tastatur des Zellulartelefons eingeben, wodurch ein nicht autorisierter Zugriff auf die Sprachnachricht durch einen anderen Anwender des Telefons verhindert wird. Genauso können spezielle Teile des Speichers zum Speichern von Telefonnummern, Adressen und so weiter ein vom Anwender eingegebenes Passwort haben, welches damit assoziiert ist, um genauso damit einen nicht autorisierten Zugriff zu verhindern. Wie erkenntlich ist, kann eine große Vielzahl von unterschiedlichen Passwortszenarios vorgesehen sein.

[0041] Wie bemerkt, kann ein Teil der Flash-Speicheranordnung eingesetzt werden, um einen Boot-Loader zu speichern. Ein getrenntes zusätzliches Passwortregister **150** ist für den Teil des Speichers vorgesehen, der den Boot-Loader speichert. Daher wird kein Schreib- oder Löschbefehl, der den Teil des Speichers beeinflusst, der den Boot-Loader speichert, ausgeführt, bis dieser Befehl zusätzlich das zusätzliche Boot-Loader-Passwort liefert. Dies sieht daher ein zusätzliches Niveau des Passwortschutzes für den kritischen Boot-Loader vor, um das Risiko zu minimieren, dass der Boot-Loader unbeabsichtigt fehlerbehaftet wird.

[0042] [Fig. 5](#) veranschaulicht das Flash-Speicherstatusregister **142** zum Speichern des Status der Flash-Makros der Flash-Speicheranordnung. Das Statusregister weist N einzelne Speicherelemente auf, um separat den Status von jedem der N Flash-Makros zu speichern. Jedes Statusregister legt unter anderen Merkmalen fest, ob das entsprechende Flash-Makro einem gerade anhängigen Lese- oder Schreibvorgang unterworfen ist. Zusätzlich kann das Statusregister weiter die Software- oder Hardware-Einheit festlegen, die den Lese- oder Schreibvorgang ausführt. Durch Speicherung der Identität der Vorrichtung, die den Lese- oder Schreibvorgang ausgibt, kann der Mikroprozessor unter Verwendung von geeigneten Priorisierungstabellen, die vom Mikroprozessor gespeichert sind, bestimmen, ob ein neu angeforderter Flash-Speichervorgang von höherer Priorität ist, und falls dies so ist, kann der Mikroprozessor den Flash-Controller unterbrechen, was bewirkt, dass er den Lese- oder Schreibvorgang abbricht, um zu gestatten, dass ein Flash-Speichervorgang mit höherer Priorität sofort ausgeführt wird.

[0043] Zusätzlich können die Statusregister weiter irgendeine Anzeige des Status eines gegenwärtigen Lese- oder Schreibvorgangs speichern. Beispielsweise kann das Statusregister den Zeitpunkt aufzeichnen, zu dem ein Lese- oder Schreibbefehl zu der Flash-Speicheranordnung geliefert wurde. Somit kann der Mikroprozessor auf den Zeitwert zugreifen und daraus bestimmen, wann der Befehl wahrscheinlich vollendet sein wird. Wenn ein Schreibbefehl typischerweise 100 Taktzyklen braucht, kann der Mikroprozessor bestimmen, ob der anhängige Schreibbefehl erst kürzlich ausgegeben wurde, oder ob er schon fast vollendet ist. Dies kann gestatten, dass der Mikroprozessor bestimmt, ob der Flash-Controller zu unterbrechen ist, um einen Schreibvorgang mit höherer Priorität auszugeben. Insofern, als Lesevorgänge betroffen sind, kann das Statusregister die Anzahl der Wartezyklen festlegen, die in Verbindung mit dem speziellen Lesevorgang verwendet werden. Wie offensichtlich ist, kann eine große Vielzahl von Statusinformationen in einem Statusregister abhängig von der Vorprogrammierung des Systems gespeichert sein. Im Allgemeinen ist eine Statusinformation vorgesehen, um den gesamten Wirkungsgrad des Systems zu verbessern, um zu gestatten, dass eine größere Anzahl von Betriebsvorgängen in einer gegebenen Zeitperiode ausgeführt wird, oder um zu gestatten, dass eine niedrigere Taktrate verwendet wird, während immer noch die gleiche Anzahl von Operationen pro Sekunde erreicht wird.

[0044] [Fig. 6](#) veranschaulicht den Betrieb der Speicherlagerungseinheit **149** ([Fig. 2](#)), die dahingehend arbeitet, dass sie hohe und niedrige Speicherteile eines gesamten Speicheradressraums **152** der Speicheranordnung **130** umlagert bzw. swappt (auch [Fig. 2](#)). Im Gebrauch wird ein Boot-Loader anfänglich an einer niedrigsten Speicheradresse des Speicheradressraums gespeichert, beispielsweise an der Adresse 0X00. Hauptsoftwareprogramme sind beginnend an einer hohen Speicheradresse gespeichert, wie beispielsweise 0X05800000. Der Boot-Loader weist Befehle auf, die festlegen, welche der Hauptsoftwareprogramme in dem Flash-Speicher als ein Teil eines Einschalt- oder Reset-Vorgangs ausgeführt müssen. Sobald der Boot-Loader seine Operationen ausgeführt hat, lagert die Speicherumlagerungseinheit den hohen und niedrigen Speicher um. Nach dem Umlagerungs- bzw. Swap-Vorgang sind somit die primären Flash-Software-Programme beginnend an der Speicherstelle 0X0 gespeichert. Der Boot-Loader ist an einer hohen Speicheradresse gespeichert, wie beispielsweise 0X05800000. Daher gestattet ein Sprung zum niedrigen Speicher nun einen Zugriff auf die Haupt-Software der Flash-Makros. Dies gestattet, dass die Haupt-Flash-Software zweckmäßiger aufgerufen wird. Wenn danach das Zellulartelefon ausgeschaltet wird und dann wieder angeschaltet wird, oder wenn irgendein anderer Reset-Vorgang ausgeführt wird, werden die Speicheradressen wieder um-

gelagert, so dass der Boot-Loader an der untersten Speicheradresse liegt, um einen angemessenen Anfangsbetrieb des Boot-Loaders zu gestatten.

[0045] Was beschrieben worden ist, sind beispielhafte Aspekte eines CDMA-basierten Mikroprozessorsystems mit einem eingebetteten Flash- und SRAM-Speicher. Prinzipien der Erfindung sind genauso auf andere Systeme anwendbar.

Patentansprüche

1. Ein Flash-Speichersystem (**112**), das Folgendes aufweist:

Flash-Speicherzellen angeordnet als ein Satz von Flash-Makros (**130**) und Lese-während-des-Schreibens-Mittel (**144**, **146**) zum Schreiben in eines der Flash-Makros während gleichzeitig aus einem anderen der Flash-Makros ausgelesen wird; wobei das System verbunden ist mit einem Mikroprozessor (**102**) und wobei die Lese-während-des-Schreibens-Mittel (**144**, **146**) Folgendes beinhalten:

Schreibmittel (**146**) zum Schreiben von Signalen, die von dem Mikroprozessor empfangen werden, in ein ausgewähltes Flash-Makro aus den Flash-Makros (**130**) und gekennzeichnet dadurch, dass die Lese-während-des-Schreibens-Mittel weiterhin Unterbrechungsmittel aufweisen, ansprechend auf dem Empfang eines Lese-Befehls von dem Mikroprozessor gerichtet an das ausgewählte eine Flash-Makro der Flash-Makros, und zwar zum Unterbrechen des Betriebes des Mikroprozessors bis die Schreibmittel (**146**) ihren Betrieb abgeschlossen haben, und dann Ausführen des Lesebefehls.

2. System nach Anspruch 1, das weiterhin Speichermittel (**142**) zum Speichern eines Wertes repräsentativ für den momentanen Status der Lese-während-des-Schreibens-Mittel (**144**, **146**) aufweist, wobei auf den Wert durch den Mikroprozessor zugegriffen werden kann.

3. System nach Anspruch 1, das weiterhin Folgendes aufweist:

einen Flash-Speichercontroller (**132**); einen Flash-Speicher-Bus (**134**) verbunden mit den Flash-Speichermakros und dem Flash-Speichercontroller;

wobei der Flash-Speichercontroller (**132**) Speichermittel (**138**) aufweist zum Speichern einer programmierbaren Anzahl von Warte-Zuständen, die den Flash-Makros zugeordnet sind, und zwar mit einer programmierbaren Anzahl von Warte-Zuständen pro Flash-Makro, und Zugriffsmittel zum Zugreifen auf ein ausgewähltes Flash-Makro, die die Lese-während-des-Schreibens-Mittel aufweisen, wobei die Zugriffsmittel auf das ausgewählte Flash-Makro zugreifen unter Verwendung des Flash-Speicher-Buss programmiert mit

der Anzahl von Warte-Zuständen, die dem ausgewählten Flash-Makro zugeordnet sind.

4. System nach Anspruch 3, wobei die Speichermittel-Warte-Zustandsregister-Einheiten beinhalten, wobei jede der Warte-Zustandsregister-Einheiten programmierbar ist, um eine ganze Zahl von Warte-Zuständen darzustellen.

5. System nach Anspruch 1, das weiterhin Folgendes aufweist:
einen Flash-Speichercontroller (**132**), der Folgendes aufweist:

Speichermittel (**140**) zum Speichern eines separaten Passworts, dass jedem Flash-Makro zugeordnet ist, und

Freigabe-Mittel ansprechend auf dem Empfang eines gültigen Passworts, für ein ausgewähltes Flash-Makro der Flash-Makros, und zwar zum Freigeben des Programmierens oder Löschens von Flash-Zellen des ausgewählten Flash-Makros während gleichzeitig das Programmieren oder Löschen von allen anderen Flash-Zellen verhindert wird.

6. Flash-Speichersystem nach Anspruch 5, wobei ein Teil der Flash-Zellen von einem der Flash-Makros Boot-Loader-Software speichert.

7. Flash-Speichersystem nach Anspruch 6, das weiterhin Folgendes aufweist:

Speichermittel zum Speichern eines Passworts, das dem Boot-Loader zugeordnet ist; und

Freigabe-Mittel ansprechend auf dem Empfang eines gültigen Passworts für den Boot-Loader, zum Freigeben der Programmierung oder des Löschens der Flash-Zellen des Boot-Loaders.

8. Flash-Speichersystem nach Anspruch 6, das weiterhin Löschmittel (**147**) aufweist zum Löschen von ausgewählten Flash-Zellen der Flash-Makros, wobei die Löschmittel selektiv eine Wort-Löschung, eine Page-Löschung oder eine Massen-Löschoperation (masse erase operation) ausführen.

9. Flash-Speichersystem nach Anspruch 6, wobei die Speichermittel (**140**) zum Speichern eines separaten Passworts, das jedem Flash-Makro zugeordnet ist, die Passwörter in einem festverdrahteten bzw. Festspeicher speichern.

10. System nach Anspruch 1, das weiterhin Folgendes aufweist:

ein Flash-Speichercontroller (**132**), der Partitionierungsmittel aufweist zum Partitionieren der Flash-Speicherzellen in hohe und niedrige Speicherorte;

wobei die Flash-Speicherzellen einen Boot-Loader speichern, und zwar beginnend mit einer niedrigsten Speicheradresse des Flashspeicherraumes; und wobei die Partitionierungsmittel Umlagerungsmittel

(**149**) zum Umlagern bzw. Tauschen der hohen und niedrigen Speicherorte nachdem die Operationen, die von dem Boot-Loader ausgeführt werden, abgeschlossen sind.

11. System nach Anspruch 1, das weiterhin Folgendes aufweist:

eine Flash-Speicherzugriffs-Einheit (**136**) mit den Lesewährend-Schreibens-Mitteln (**144**, **146**) zum Schreiben in ein Flash-Makro der Flash-Makros während gleichzeitig aus einem anderen der Flash-Makros gelesen wird.

Es folgen 5 Blatt Zeichnungen

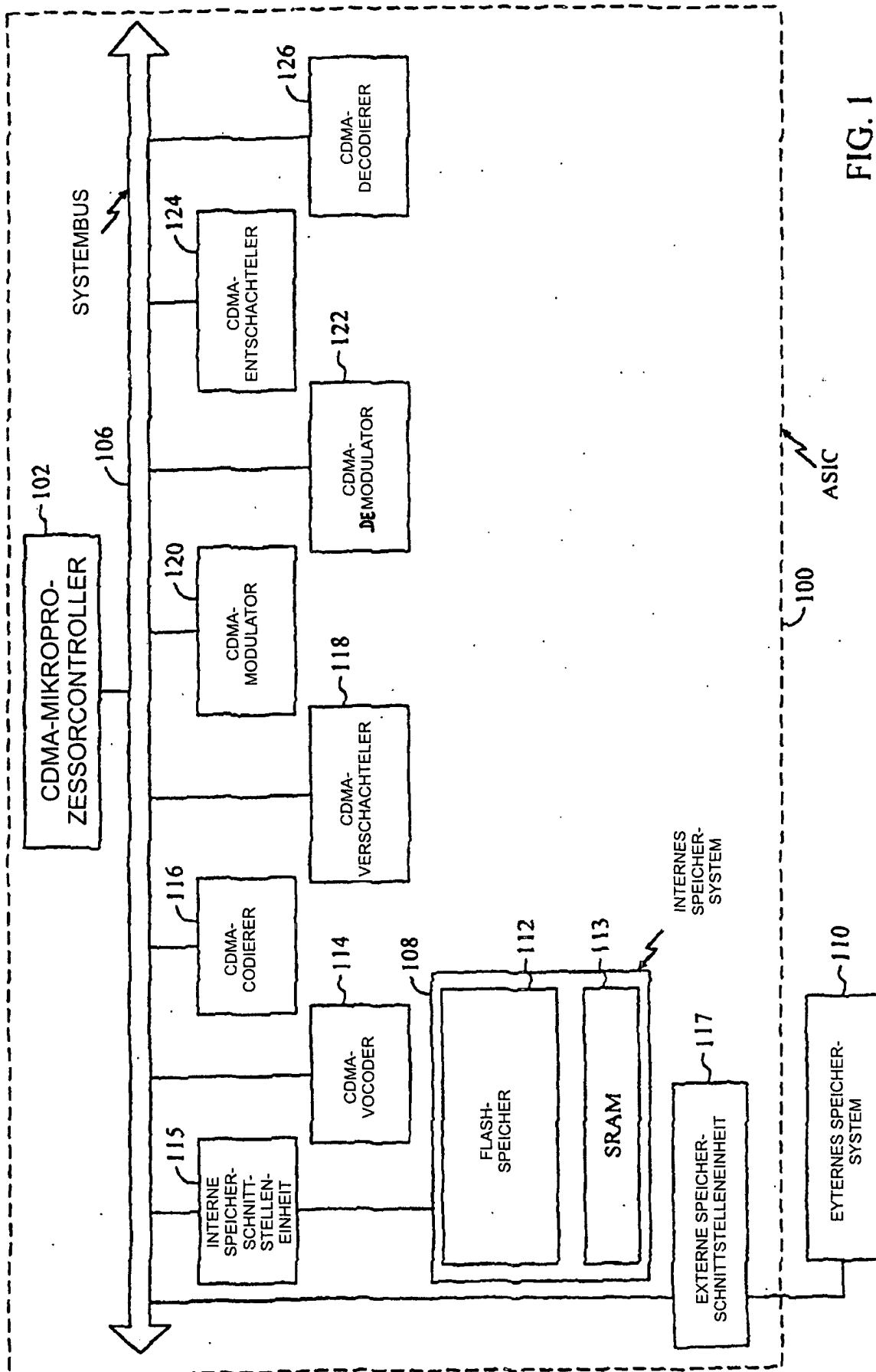


FIG. 1

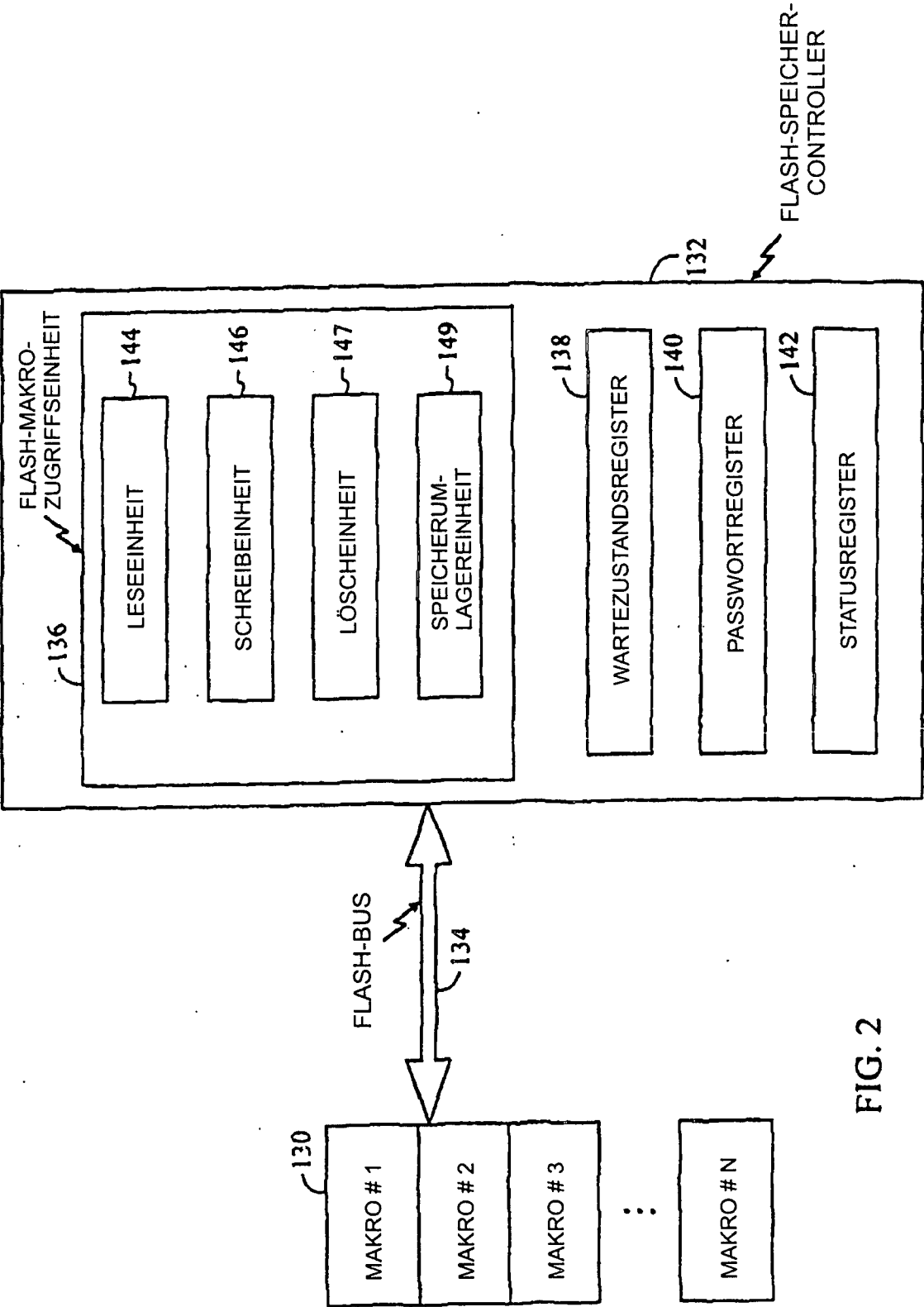


FIG. 2

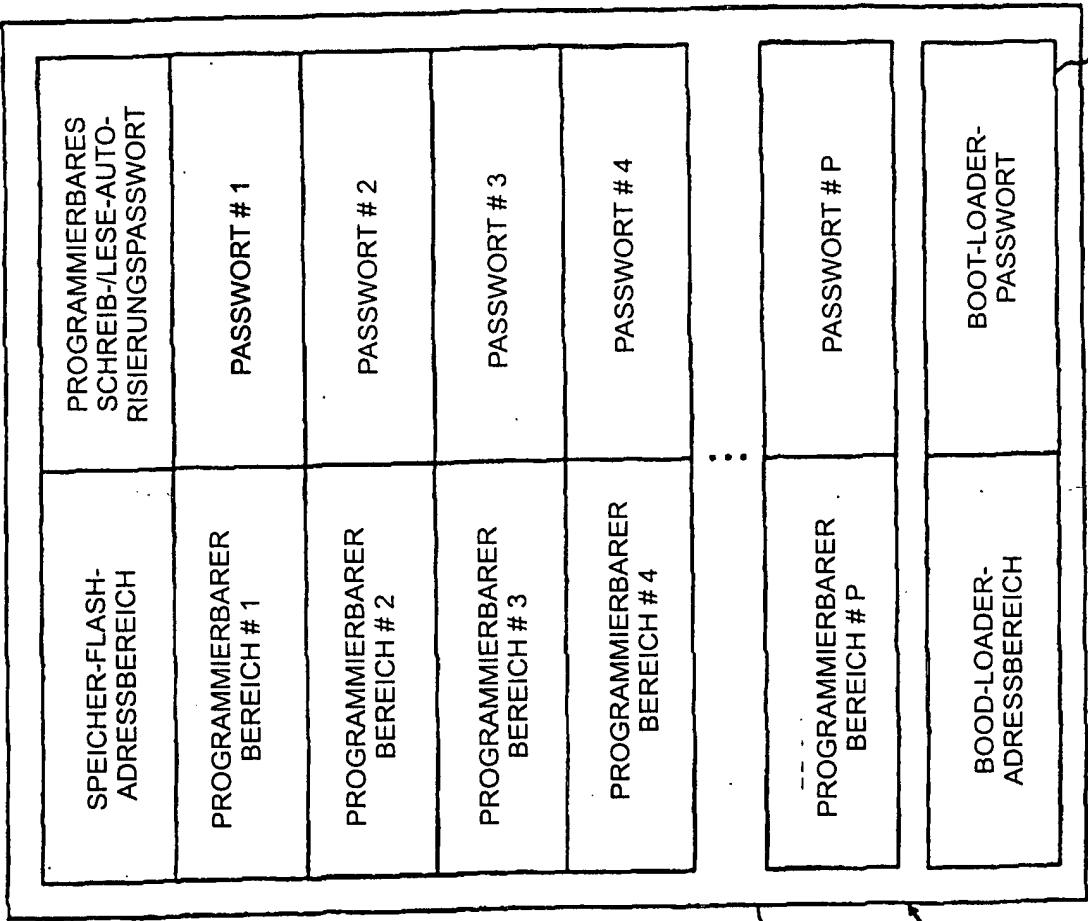


FIG. 4

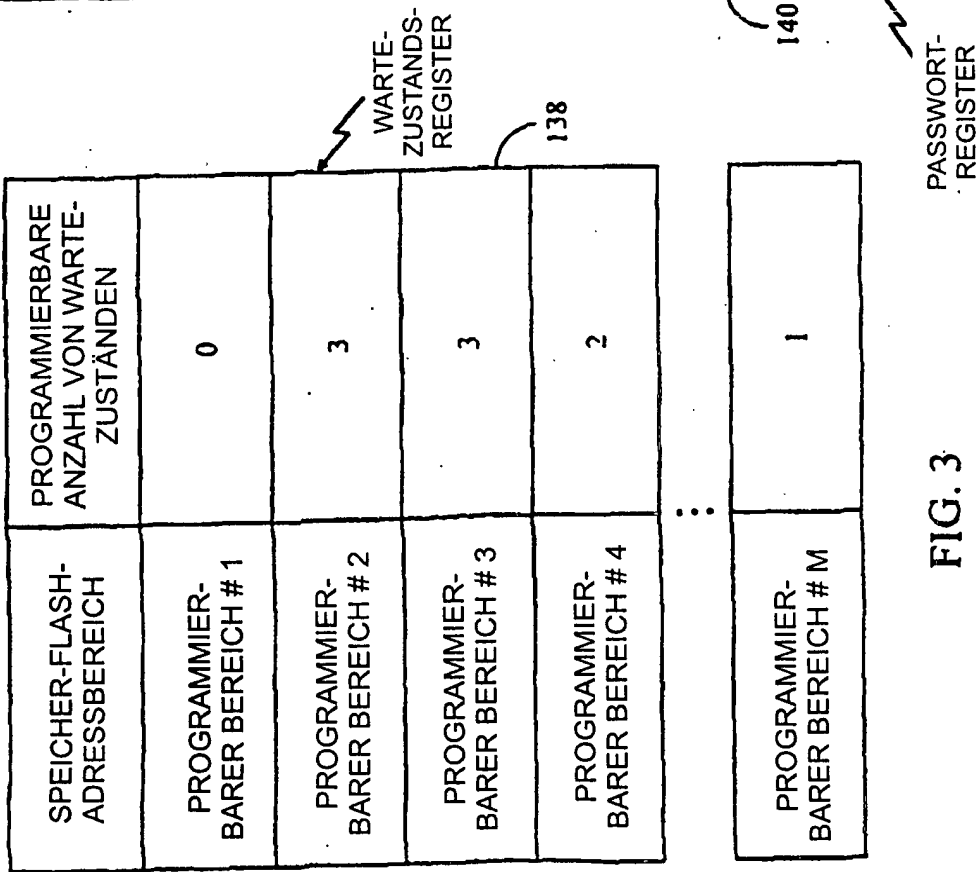


FIG. 3

FLASH-MAKRO	AKTUELLER FLASH-MAKRO-STATUS
FLASH-MAKRO # 1	LESEVORGANG IN BETRIEB
FLASH-MAKRO # 2	NICHT IN VERWENDUNG
FLASH-MAKRO # 3	NICHT IN VERWENDUNG
FLASH-MAKRO # 4	SCHREIBVORGANG IN BETRIEB
⋮	
FLASH-MAKRO # N	NICHT IN VERWENDUNG

STATUS-REGISTER

142

FIG. 5

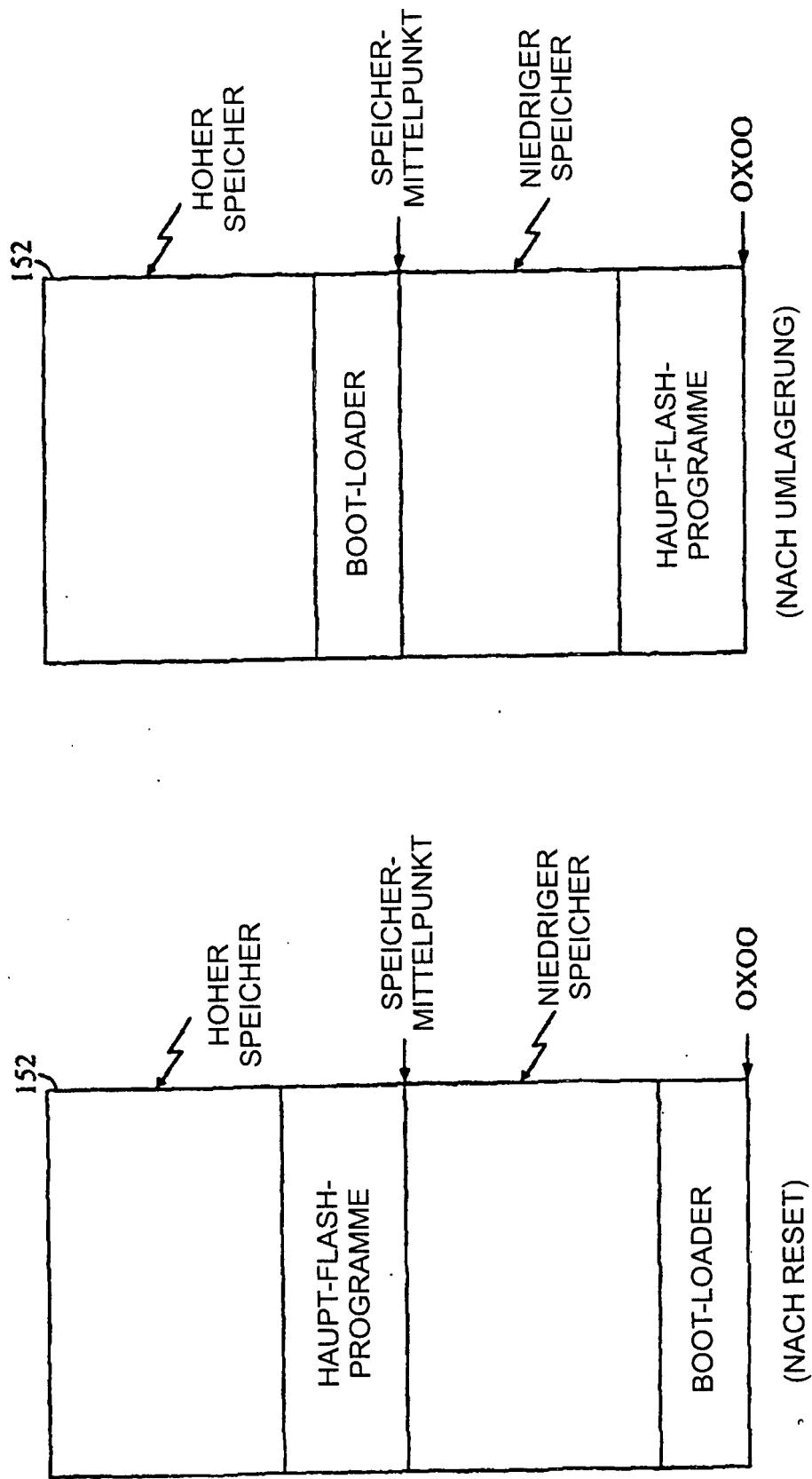


FIG. 6