(54) Title: MECHANISM FOR COMMUNICATING BETWEEN TWO APPLICATIONS ON A SAFETY MODULE

(54) Bezeichnung : MECHANISMUS ZUR KOMMUNIKATION ZWISCHEN ZWEI APPLIKATIONEN AUF EINEM SICHERHEITSMODUL

(57) Abstract: The invention describes a method for operating a safety module, having the steps of transmitting data from a first application (10) to a second application (12) in the safety module and of using the transmitted data in the second application (12). The transmitted data comprises a command for the second application (12). The second application (12) carries out the command in the use step. In the transmission step, the first application (10) calls an internal diversion to the second application (12). The safety module diverts the command internally in such a way that it is received by the second application (12) as a conventional command which is received from outside the safety module.

Erklärungen gemäß Regel 4.17:

— hinsichtlich der Berechtigung des Anmelders, ein Patent zu beantragen und zu erhalten (Regel 4.17 Ziffer ii)

— hinsichtlich der Berechtigung des Anmelders, die Priorität einer früheren Anmeldung zu beanspruchen (Regel 4.17 Ziffer iii)

Veröffentlicht:

— mit internationalem Recherchenbericht (Artikel 21 Absatz 3)

— vor Ablauf der für Änderungen der Ansprüche geltenden Frist; Veröffentlichung wird wiederholt, falls Änderungen eingehen (Regel 48 Absatz 2 Buchstabe h)
Mechanismus zur Kommunikation zwischen zwei Applikationen auf einem Sicherheitsmodul


In der Regel ist es Applikationen eines Sicherheitsmoduls nicht möglich, Daten mit anderen Applikationen des Sicherheitsmoduls auszutauschen. Dies ist jedoch wünschenswert, um die Funktionalität von Applikationen eines Sicherheitsmoduls erweitern zu können. Eine bekannte Möglichkeit, Daten zwischen zwei Applikationen auszutauschen, ist das sog. „Sharable Interface“. Die Verwendung dieser Schnittstelle weist jedoch den Nachteil auf, dass die aus einer Applikation aufgerufene Applikation eine Schnittstelle zur Verfügung stellen muss. Darüber hinaus ist die aufgerufene Funktion bei der Abarbeitung nicht selektiert, wodurch die Verwendung eines Clear on Deselect (CoD)-Speichers nicht möglich ist.


Es ist Aufgabe der vorliegenden Erfindung, ein Verfahren und ein Sicherheitsmodul anzugeben, mit denen der Austausch von Daten zwischen Applikationen des Sicherheitsmoduls auf einfachere und flexiblere Weise durchgeführt werden kann. Insbesondere soll ein Inter-Applikationsdatenaustausch auch dann möglich sein, wenn eine Applikation erst nachträglich in das Sicherheitsmodul eingebracht wird.

Diese Aufgaben werden gelöst durch ein Verfahren gemäß den Merkmalen des Patentanspruches 1 sowie ein Sicherheitsmodul gemäß den Merkmalen des Patentanspruches 8. Vorteilhafte Ausgestaltungen ergeben sich aus den abhängigen Patentansprüchen.

Die Erfindung schafft ein Verfahren zum Betreiben eines Sicherheitsmoduls, mit den Schritten: Übertragen von Daten von einer ersten Applikation zu ei-
ner zweiten Applikation in dem Sicherheitsmodul und Verwenden der übertra
genen Daten in der zweiten Applikation. Das Verfahren zeichnet sich da-
durch aus, dass die übertra
genen Daten ein Kommando für die zweite Ap-
plikation umfassen und die zweite Applikation das Kommando in dem
Schritt des Verwendens ausführt. In dem Schritt des Übertragens ruft die ers-
te Applikation eine interne Umleitung zu der zweiten Applikation auf. Das
Sicherheitsmodul leitet das Kommando intern so um, dass es durch die zwei-
te Applikation als herkömmliches, wie ein von außerhalb des Sicherheitsmo-
duls erhaltenes Kommando empfangen wird.

Die Erfindung schafft weiter ein Sicherheitsmodul mit einer physischen
Schnittstelle zu einem externen Terminal und einem Betriebssystem zur Aus-
führung von Applikationen, wobei das Sicherheitsmodul angepasst ist, eine
Kommunikation zwischen den Applikationen zu unterstützen. Das Sicher-
heitsmodul zeichnet sich dadurch aus, dass dieses ein internes, virtuelles
Terminal aufweist. Das Sicherheitsmodul ermöglicht die Kommunikation
zwischen den Applikationen auf dem Sicherheitsmodul durch das virtuelle
Terminal.

Durch die Erfindung wird einer Applikation in dem Sicherheitsmodul der
Aufruf eines karteninternen, virtuellen Kartenlesers ermöglicht, welcher mitt-
tels herkömmlicher Kommandos mit einer anderen Applikation kommunik-
zieren kann. Der Vorteil besteht darin, dass das Betriebssystem des Sicher-
heitselements den virtuellen Kartenleser wie jede andere Schnittstelle, bspw.
ein kontaktloses oder kontaktbehaftetes Interface, behandeln kann. Durch die
interne Umleitung eines Kommandos von einer ersten zur zweiten Applikation
ist bspw. eine Interapplikationskommunikation über die in dem Sicher-
heitselement bereits vorhandene APDU-Schnittstelle sowie die Nutzung ei-
nes Clear on Deselect (CoD)-Speichers eines Sicherheitsmoduls möglich.
Darüber hinaus ergeben sich erweiterte Funktionalitäten für den Betrieb der Applikationen, die nachfolgend näher beschrieben werden.


Es ist weiterhin zweckmäßig, wenn die APDU-Schnittstellen der ersten und der zweiten Applikation parallel aktiv sind, über welche das Kommando von der ersten an die zweite Applikation übertragen wird. Diese Ausgestaltung ermöglicht die Interapplikationskommunikation zwischen der ersten und der zweiten Applikation. Dabei kann der eingangs bereits erwähnte „Clear on Deselect“-Speicher genutzt werden.

In einer weiteren Ausgestaltung erfolgt die interne Umleitung des Kommandos über ein virtuelles Terminal des Sicherheitsmoduls, welche das von der ersten Applikation empfangene Kommando über eine virtuelle Schnittstelle an die zweite Applikation sendet. Die virtuelle Schnittstelle kann bspw. in Gestalt einer Terminal-API (Application Programmable Interface) vorliegen.
In einer Ausgestaltung des erfindungsgemäßen Verfahrens wird die erste Applikation als Dispatcher betrieben, welcher Daten mit mehreren zweiten Applikationen mit demselben Applikationskennzeichen (Application ID, AID) austauschen kann.


Das erfindungsgemäße Verfahren zeichnet sich dadurch aus, dass das virtuelle Terminal keine Unterscheidung vornimmt, ob es ein Kommando von einem externen Terminal oder ein Kommando von einer Applikation empfängt, die an eine weitere Applikation weitergeleitet werden soll. Dabei stellt das virtuelle Terminal den Applikationen eine dedizierte Funktion nur zum Senden eines zwischen zwei Applikationen übertragenen Kommandos bereit.

Eine Applikationslogik zur Bedienung der zweiten Applikation wird in der Regel in der ersten Applikation realisiert. In welcher Weise die Logik ausgebildet ist, kommt auf die Aufgaben der beiden Applikationen an. Beispielsweise kann eine erste Applikation die Aufgabe haben, eine Kommunikation für die zweite Applikation zu überwachen und zu einem bestimmten Zeitpunkt Daten aus der Kommunikation zu extrahieren und zu visualisieren. Beispielsweise könnte hierdurch ein abgebuchter Betrag einer Kreditkarte
angezeigt werden, ohne die eventuell zugelieferte und evaluierte Kreditkartenapplikation, d.h. die zweite Applikation, zu verändern.


Die Erfindung wird nachfolgend näher anhand von Ausführungsbeispielen in der Zeichnung erläutert. Es zeigen:

Fig. 1 eine schematische Darstellung der grundlegenden Architektur eines erfindungsgemäßen Sicherheitselements, insbesondere in Gestalt eines tragbaren Datenträgers, in dem der erfindungsgemäße Mechanismus zur Kommunikation zwischen zwei Applikationen implementiert ist,

Fig. 2 eine schematische Darstellung eines generischen Flussdiagramms, welches das Prinzip des erfindungsgemäßen Mechanismus illustriert,
Fig. 3  eine schematische Darstellung eines Flussdiagramms, in welchem eine erste Applikation als Proxy ausgestaltet ist,

Fig. 4  eine schematische Darstellung eines Flussdiagramms, in der die erste Applikation als Dispatcher ausgestaltet ist, und

Fig. 5  eine schematische Darstellung eines Flussdiagramms, in dem eine Interapplikationskommunikation zwischen erster und zweiter Applikation dargestellt ist.

Fig. 1 zeigt eine schematische Darstellung der grundlegenden Architektur eines Sicherheitselements, in dem der erfindungsgemäße Mechanismus zur Kommunikation zwischen zwei Applikationen implementiert ist. In einer dem Fachmann bekannten Weise repräsentiert das Bezugszeichen 110 die Hardware-Plattform des Sicherheitselements. Das Bezugszeichen 110 kennzeichnet das Betriebssystem. Ein Transport-Manager des Betriebssystems 100 ist mit 111 gekennzeichnet. Als Protokolle 112 können beispielsweise T=0, T=1, USB, MMC oder T=CL eingesetzt werden. Mit 113 ist eine Speichermanagementeinheit („Memory Management“) des Betriebssystems 110 gekennzeichnet. Das Betriebssystem 110 umfasst weiter eine virtuelle Maschine 114, sowie einen kryptographischen Dienst 115. Der kryptographische Dienst 115 kann dazu ausgebildet oder eingerichtet sein, DES, AES, RSA, DSA, ECC, RNG, usw. nutzende Kryptographieverfahren zu verarbeiten.

Das Bezugszeichen 120 kennzeichnet ein in dem Sicherheitselement implementiertes Java-Card System, vorzugsweise eine JAVA-Card virtuelle Maschine. 130 repräsentiert die Applikationsschnittstelle API (Application Programmable Interface). Das Java-Card System 120 umfasst eine virtuelle Schnittstelle („Virtual Card“) 121, einen APDU-Dispatcher 122 sowie weitere


In der vorstehenden Beschreibung sind, sofern für den Fachmann gebräuchlich, die üblichen englischen Begriffe verwendet worden.


Die Übertragung von Daten von dem User Applet 142 als erster Applikation zu dem User Applet 143 als zweite Applikation erfolgt durch eine interne Umleitung der zu übertragenden Daten in dem Sicherheitselement. Bei den zu übertragenden Daten handelt es sich insbesondere um ein Kommando für
die zweite Applikation, welche durch die zweite Applikation nach Erhalt des Kommandos ausgeführt wird. Die Umleitung des Kommandos erfolgt durch das Sicherheitsmodul intern derart, dass es durch die zweite Applikation als herkömmliches, wie ein von außerhalb des Sicherheitsmoduls erhaltenes Kommando empfangen wird.


Das Umleitungsmittel umfasst insbesondere ein virtuelles Terminal 131, welches durch die erste Applikation 142 aufrufbar ist. Es ist besonders vorteilhaft, wenn das virtuelle Terminal 131 das intern umzuleitende Kommando (APDU) über eine virtuelle Schnittstelle 121 an die herkömmlichen Mittel 122, 132 zum Weiterleiten von Kommandos an Applikationen 140, 150 über- gibt. Für die Weiterleitungsmittel 122, 132 stellt die virtuelle Schnittstelle 121 eine weitere Schnittstelle dar, von welcher das Sicherheitsmodul Kommandos empfängt. Die virtuelle Schnittstelle 121 wird in dem Sicherheitsmodul wie eine (weitere) physische Schnittstelle verwaltet.

Der APDU-Dispatcher 122 wird vorzugsweise eine Antwort der Anwendung 143 auf das Kommando 184 von der virtuellen Schnittstelle 121 (bzw. den Umleitungsmitteln 121,131) entsprechend wieder an die virtuelle Schnittstelle 121 (bzw. die Umleitungsmittel 121,131) weitergeben. Die Antwort der
zweiten Applikation 143 kann das virtuelle Terminal 131 somit der ersten Applikation 142 wieder zur Verfügung stellen.


In den nachfolgend beschriebenen Fig. 2 bis 4 sind schematisch Flussdiagramme dargestellt, die das erfindungsgemäße Vorgehen innerhalb des Sicherheitsmoduls veranschaulichen.

Fig. 2 zeigt dabei ein generisches Flussdiagramm. Als Komponenten sind als erste Applikation 10 ein Applet 1, als zweite Applikation 12 ein Applet 2, das Betriebssystem 14 (System) des Sicherheitsmoduls und ein virtuelles Terminal 16, der als Loop Back Card Reader bezeichnet ist, des Sicherheitsmoduls beteiligt. Die zwischen den beteiligten Komponenten transferierten Kommandos werden zeitlich aufeinander folgend von oben nach unten ausgeführt. Die Vorgehensweise wird dabei anhand eines Java-Card Systems, das bereits in Fig. 1 beschrieben wurde, illustriert. Das virtuelle Terminal 16 ist zugleich virtuelle Schnittstelle für das System 14. Prinzipiell sind die Erfindung und deren Vorgehensweise allgemein auf solche Sicherheitsmodule
anwendbar, in denen ein interpreterbasiertes Betriebssystem verwendet wird. Neben Java-Card ist dies bspw. MEL.

Das mit den Bezugszeichen 200 bis 222 gekennzeichnete Vorgehen beschreibt eine Selektion einer Applikation auf der JAVA-Card bzw. ein Ansprechen der Java-Card. Der eigentliche Austausch von Daten, d.h. das Übertragen eines Kommandos von der ersten Applikation 10 zur zweiten Applikation 12, erfolgt dann in den Schritten 230 bis 240.


Im Schritt 204 überträgt die erste Applikation 10 ein Kommando („CardReader.getInstance(LOOPBACK_READER)“), mit welcher die erste Applikation 10 das externe Terminal ansprechen möchte. In Schritt 206 („create“) erzeugt das System 14 das virtuelle Terminal 16, das als Loop Back Card Reader bezeichnet wird. In Schritt 208 übermittelt das System 14 einen Kennzeichner („reader_ref“) der ersten Applikation 10. In Schritt 210 spricht die erste Applikation 10 das virtuelle Terminal 16 an, indem das Kommando „reader_ref.open()“ übermittelt wird. In Schritt 212 wird die Schnittstelle des virtuellen Terminals aktiviert („OpenLoopbackInterface()“). Die Schritte 204 bis 212 sind insofern optionale Schritte als dass das virtuelle Terminal 16 nicht notwendigerweise wie dargestellt erst gestartet und/oder deren Referenz bekannt gegeben werden muss. Das virtuelle Terminal 16 kann auf dem Sicherheitsmodul alternativ ständig unter bekanntem Namen (Referenz) ausgeführt werden oder bereits zuvor gestartet worden sein.

In Schritt 230 empfängt das System 14 eine Kommando-APDU von dem externen Terminal. In Schritt 232 wird das empfangene Kommando („Applet1.process(APDU)“) an die erste Applikation 10 weitergeleitet. Zur Verarbeitung des Kommandos ist auch die zweite Applikation 12 erforderlich. In Schritt 234 ruft die erste Applikation 10 hierzu eine Umleitungsfunktion des virtuellen Terminals 16 auf „Reader_ref.sendAPDU(APDU)“. Das virtuelle Terminal 16 leitet in den Schritten 236 und 238 über das System 14 das Kommando zur zweiten Applikation 12 um. Hierzu werden die virtuelle Schnittstelle „LoopbackInterface().send(APDU)“ und dann der herkömmliche Weg zur Kommandowiederleitung „Applet2.process(APDU)“ verwendet, welche die zweite Applikation zur Verarbeitung des Kommandos veranlasst. Eine entsprechende Antwort der zweiten Applikation 12 auf das Kommando wird von dieser an das System 14 übermittelt, welche die Antwort an das virtuelle Terminal 16 für die Weiterleitung an die erste Applikation 10 über-
trägt. Eine entsprechende Antwort „response“ wird in Schritt 240 an das externe Terminal übertragen. Nach Beendigung der Transaktion wird in Schritt 250 das virtuelle Terminal 16 beendet („terminate“).

In den Fig. 3 bis 5 werden nun spezifische Anwendungen des erfindungsge- mäßen Prinzips beschrieben. Wie in Verbindung mit Fig. 2 erläutert, erfolgt in den Beispielen der Fig. 3 und Fig. 4 dabei zunächst ein Auswahlvorgang zur Aktivierung der APDU-Schnittstelle. Anschließend werden jeweilige Aktionen durchgeführt. Zur Vereinfachung der Beschreibung werden nachfolgend jeweils nur die Unterschiede in den zwischen den einzelnen Komponenten übertragenen Kommandos im Vergleich zur Fig. 2 erläutert, welche das grundsätzliche Vorgehen zeigt.


Im Ausführungsbeispiel der Fig. 4 wird die erste Applikation 10 als Dispatcher betrieben, welcher selektiv das empfangene Kommando an eine von mehreren gleichartigen zweiten Applikationen 12 verteilt. Im Gegensatz zu dem bereits im System 14 vorhandenen Dispatcher, der das Kommando an die vom externen Terminal ausgewählte und mit der AID als Applikationskennzeichen adressierte Applikation leitet, wählt die Dispatcher-Applikation selbst eine der Applikationen 12 aus. Die mehreren Applikationen 12 sind alle geeignet die aplikationsspezifischen Kommandos, für die

In Schritt 414 wählt die Dispatcher-Applikation 10 eine der mehreren Applikationen aus und leitet das Kommando, in diesem Fall zunächst das Select-Kommando, an die ausgewählte Applikation 12 weiter. In den Schritten 416, 418, 420 und 422 wird entsprechend das Select-Kommando an die ausgewählte Applikation 12 mit dem lokalen (oder internen) Applikationsbezeichner AID_lokal2 übertragen.


Alle weiteren vom externen Terminal empfangenen Kommandos 430, wie das Kommando APDU2, werden von der Dispatcher-Applikation 10 transparent, d.h. unverändert, an die ausgewählte Applikation 12 weiter geleitet.
Dieses Vorgehen ist bspw. dann hilfreich, wenn mehrere gleichartige Applikationen in dem Sicherheitsmodul vorhanden sind. Ihnen wird jeweils ein unterschiedlicher lokaler Applikationskennzeichen „AID_lokal x“ zugeordnet, wobei ein x eine bestimmte Applikation kennzeichnet. In diesem Fall ist es für das externe Terminal nicht erforderlich, Kenntnis über die für den (Bezahl-)Vorgang zu verwendennde Applikation zu haben, da die entsprechende Auswahl durch die erste Applikation 10 erfolgt.

Im Ausführungsbeispiel gemäß Fig. 5 erfolgt eine Interapplikationskommunikation.


Im Gegensatz zu den vorangegangenen Ausführungsbeispielen in den Fig. 2 bis 4 erfolgt hier kein zweistufiger Ablauf mit einem vorgeschalteten Auswahlvorgang zur Aktivierung der APDU-Schnittstelle der zweiten Applikation, dessen Erfolg dem externen Terminal bestätigt wird (siehe 222, 326, 426). Nach Erhalt einer Kommando-APDU1 in Schritt 500 wird durch das System 14 und das virtuelle Terminal 16 eine interne Umleitung der zwischen den beiden Applikationen 10, 12 übertragenen Daten vorgenommen,
sodass (erst) in Schritt 540 eine Antwort auf die Kommando-APDU1 an das externe Terminal übertragen wird („Response to APDU1“).

In dem in Fig. 5 dargestellten Ausführungsbeispiel erfolgt dabei beispielhaft eine zweimalige für das externe Terminal nicht sichtbare Interkommunikation mittels APDU2 und APDU3 zwischen den Applikationen 10, 12. Die zwischen der ersten und der zweiten Applikation 12 für die APDU2 ausgetauschten Kommandos sind mit den Bezugszeichen 522 bis 528 gekennzeichnet. Die die APDU3 betreffenden Kommandos sind zusätzlich mit einem „‘“ bei gleicher Nummerierung gekennzeichnet.

Prinzipiell kann zur Verarbeitung der Kommando-APDU1 eine beliebige Anzahl an Kommunikationen zwischen den Applikationen 10, 12 erfolgen.

**Patentansprüche**

1. Verfahren zum Betreiben eines Sicherheitsmoduls, mit den Schritten:
   - Übertragen von Daten von einer ersten Applikation (10) zu einer zweiten Applikation (12) in dem Sicherheitsmodul;
   - Verwenden der übertragenen Daten in der zweiten Applikation (12);
   dadurch gekennzeichnet, dass
   - die übertragenen Daten ein Kommando für die zweite Applikation (12) umfassen;
   - die zweite Applikation (12) das Kommando in dem Schritt des Verwendens ausführt; und
   - in dem Schritt des Übertragens
   -- die erste Applikation (10) eine interne Umleitung zu der zweiten Applikation (12) aufruft,
   -- das Sicherheitsmodul das Kommando intern so umleitet, dass es durch die zweite Applikation (12) als herkömmliches, wie ein von außerhalb des Sicherheitsmoduls erhaltenes Kommando empfangen wird.

2. Verfahren nach Anspruch 1, dadurch gekennzeichnet, dass das von der zweiten Applikation (12) empfangene Kommando wie das herkömmliche Kommando durch die zweite Applikation (12) verarbeitet wird.

3. Verfahren nach Anspruch 1 oder 2, dadurch gekennzeichnet, dass das Kommando an einer APDU-Schnittstelle der zweiten Applikation (12) empfangen wird.
4. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, dass APDU-Schnittstellen der ersten und der zweiten Applikation (10, 12) parallel aktiv sind, über welche das Kommando von der ersten an die zweite Applikation (10, 12) übertragen wird.

5. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, dass die interne Umleitung des Kommandos über ein virtuelles Terminal (16) des Sicherheitsmoduls erfolgt, welche das von der ersten Applikation (10) empfangene Kommando über eine virtuelle Schnittstelle an die zweite Applikation (12) sendet.

6. Verfahren nach einem der vorhergehenden Ansprüche, dadurch gekennzeichnet, dass die erste Applikation (10) als Dispatcher betrieben wird, welcher eine von mehreren zweiten Applikationen (12) auswählt und Kommandos an die ausgewählte zweite Applikation (12) weiterleitet.

7. Verfahren nach einem der Ansprüche 1 bis 5, dadurch gekennzeichnet, dass die erste Applikation (10) als Proxy betrieben wird, welcher aus einer Antwort der zweiten Applikation (12) Daten extrahiert.

8. Verfahren nach einem der Ansprüche 1 bis 5 oder 7, dadurch gekennzeichnet, dass die erste Applikation (10) als Proxy betrieben wird, welcher ein empfangenes Kommando für die zweite Applikation (12) ändert.

9. Verfahren nach einem der Ansprüche 1 bis 5, 7 oder 8, dadurch gekennzeichnet, dass die erste Applikation (10) als Proxy betrieben wird, welcher ein Kommando für die zweite Applikation (12) erzeugt.
10. Sicherheitsmodul mit einer physischen Schnittstelle zu einem externen Terminal und einem Betriebssystem (14) zur Ausführung von Applikationen (10, 12), wobei das Sicherheitsmodul angepasst ist, eine Kommunikation zwischen den Applikationen zu unterstützen, dadurch gekennzeichnet, dass
- das Sicherheitsmodul ein internes, virtuelles Terminal (16) aufweist;
- das Sicherheitsmodul die Kommunikation zwischen den Applikationen (10, 12) auf dem Sicherheitsmodul durch das virtuelle Terminal (16) ermöglicht.

11. Sicherheitsmodul nach Anspruch 10, dadurch gekennzeichnet, dass das interne, virtuelle Terminal (16) für eine erste der Applikationen, welche im Rahmen der Kommunikation Daten, insbesondere umfassend ein Kommando für eine zweite der Applikationen, an das virtuelle Terminal sendet, wie ein außerhalb des Sicherheitsmodul angeordnetes Terminal ansprechbar ist.

12. Sicherheitsmodul nach Anspruch 10 oder 11, dadurch gekennzeichnet, dass das Sicherheitsmodul für die Kommunikation zwischen den Applikationen (10, 12) auf dem Sicherheitsmodul eine virtuelle Schnittstelle (121) umfasst.

13. Sicherheitsmodul nach Anspruch 12, dadurch gekennzeichnet, dass die virtuelle Schnittstelle (121) in dem Betriebssystem (14) wie eine physische Schnittstelle des Sicherheitsmoduls verwaltet wird.
14. Sicherheitsmodul nach einem der Ansprüche 10 bis 13, dadurch gekennzeichnet, dass dieses dazu eingerichtet ist, die Schritte des Verfahren gemäß einem der Ansprüche 1 bis 9 durchzuführen.
Fig. 1
Fig. 2
Fig. 3
Fig. 4
Fig. 5
**INTERNATIONAL SEARCH REPORT**

**INV. G06F21/85**

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

**B. FIELDS SEARCHED**

Minimum documentation searched (classification system followed by classification symbols)

G06F

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

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

EP0-Internal

**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

<table>
<thead>
<tr>
<th>Category</th>
<th>Citation of document, with indication, where appropriate, of the relevant passages</th>
<th>Relevant to claim No.</th>
</tr>
</thead>
</table>

**Date of the actual completion of the international search**

10 December 2012

**Date of mailing of the international search report**

18/12/2012

**Name and mailing address of the ISA/Authorized officer**

Kleiber, Michael
<table>
<thead>
<tr>
<th>Patent document cited in search report</th>
<th>Publication date</th>
<th>Patent family member(s)</th>
<th>Publication date</th>
</tr>
</thead>
<tbody>
<tr>
<td>JP 2008530936 A</td>
<td>07-08-2008</td>
<td>US 2008163247 A1</td>
<td>03-07-2008</td>
</tr>
<tr>
<td>WO 2006087657 A1</td>
<td>24-08-2006</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**INTERNATIONALER RECHERCHENBERICHT**

**Inv. G06F21/85**

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

**B. RECHERCHIERTE GEBIETE**

Recherchierte Mindestprüfstoff (Klassifikationssystem und Klassifikationssymbole)

G06F

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

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

EP0-Internal

**C. ALS WESENTLICH ANGESCHENE UNTERLAGEN**

<table>
<thead>
<tr>
<th>Kategorie</th>
<th>Bezeichnung der Veröffentlichung, soweit erforderlich unter Angabe der in Betracht kommenden Teile</th>
<th>Betr. Anspruch Nr.</th>
</tr>
</thead>
</table>

Weitere Veröffentlichungen sind der Fortsetzung von Feld C zu entnehmen

Siehe Anhang Patentfamilie

Datum des Abschlusses der internationalen Recherche

10. Dezember 2012

Abschiedsdatum des internationalen Recherchenberichts

18/12/2012

Name und Postanschrift der Internationalen Recherchenbehörde

Europäisches Patentamt, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk
Tel. (+31-70) 340-3400, Fax: (+31-70) 340-3016

Bevollmächtigter Bediensteter

Kleiber, Michael

Formblatt PCT/ISA210 (Blatt 2, April 2005)
<table>
<thead>
<tr>
<th>Im Recherchenbericht angeführtes Patentdokument</th>
<th>Datum der Veröffentlichung</th>
<th>Mitglied(er) der Patentfamilie</th>
<th>Datum der Veröffentlichung</th>
</tr>
</thead>
<tbody>
<tr>
<td>JP 2008530936 A</td>
<td>07-08-2008</td>
<td></td>
<td></td>
</tr>
<tr>
<td>US 2008163247 A1</td>
<td>03-07-2008</td>
<td></td>
<td></td>
</tr>
<tr>
<td>WO 2006087657 A1</td>
<td>24-08-2006</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>