(19) österreichisches patentamt

(10) **AT 526879 A1 2024-08-15**

(12) Österreichische Patentanmeldung

(21) Anmeldenummer: A 50068/2023
(22) Anmeldetag: 03.02.2023
(43) Veröffentlicht am: 15.08.2024

(51) Int. Cl.: ***G06F 21/57*** (2013.01)
G06F 21/64 (2013.01)
G06F 21/70 (2013.01)
G06F 21/71 (2013.01)
G06F 21/78 (2013.01)

(56) Entgegenhaltungen:
US 2007038851 A1
DE 102006006109 A1
DE 102020133738 A1
WO 03058409 A2

(71) Patentanmelder:
btv technologies gmbh
59423 Unna (DE)

(72) Erfinder:
Königshaus Markus
59427 Unna (DE)

AT 526879 A1 2024-08-15

(54) Verfahren zum Beschreiben von Daten auf einen IC sowie System zur Ausführung des Verfahrens

(57) Die Erfindung betrifft ein Verfahren zum Beschreiben von Daten auf einen IC 3 mit den folgenden Schritten:

1. Bereitstellen eines in verschlüsselter Form vorliegenden Datenpakets, enthaltend die auf den IC 3 zu schreibenden Daten;
2. Laden des verschlüsselten Datenpakets;
3. Entschlüsseln des verschlüsselten Datenpaketes und Ablegen des entschlüsselten Datenpaketes ausschließlich in einem flüchtigen Speicher 10;
4. Bereitstellen des entschlüsselten Datenpakets in dem flüchtigen Speicher 10 für eine Programmiermaschine 6;
5. Beschreiben eines oder mehrerer ICs 3 mit für den IC 3 bestimmten Daten des entschlüsselten Datenpakets mittels der Programmiermaschine 6;
6. Nach dem Fertigstellen des Beschreibens des bzw. der mehreren ICs 3: Löschen des flüchtigen Speichers 10.

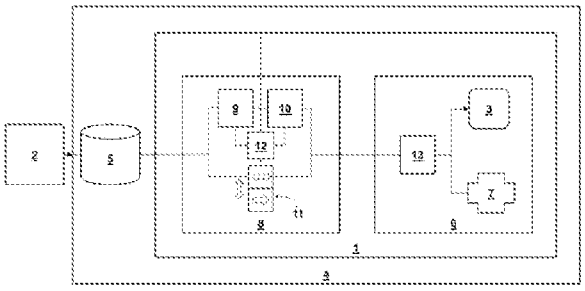

Fig. 1

Zusammenfassung

Die Erfindung betrifft ein Verfahren zum Beschreiben von Daten auf einen IC 3 mit den folgenden Schritten:

1. Bereitstellen eines in verschlüsselter Form vorliegenden Datenpakets, enthaltend die auf den IC 3 zu schreibenden Daten;
2. Laden des verschlüsselten Datenpakets;
3. Entschlüsseln des verschlüsselten Datenpaketes und Ablegen des entschlüsselten Datenpaketes ausschließlich in einem flüchtigen Speicher 10;
4. Bereitstellen des entschlüsselten Datenpakets in dem flüchtigen Speicher 10 für eine Programmiermaschine 6;
5. Beschreiben eines oder mehrerer ICs 3 mit für den IC 3 bestimmten Daten des entschlüsselten Datenpakets mittels der Programmiermaschine 6;
6. Nach dem Fertigstellen des Beschreibens des bzw. der mehreren ICs 3: Löschen des flüchtigen Speichers 10.

Fig. 1

Verfahren zum Beschreiben von Daten auf einen IC sowie System zur Ausführung des Verfahrens

Die Erfindung betrifft ein Verfahren zum Beschreiben von Daten auf einen IC sowie ein System zur Ausführung dieses Verfahrens.

In Elektronikartikeln sind ICs essenzieller Bestandteil. Sie dienen dem Ausführen unterschiedlichster Aufgaben, sie empfangen und werten Daten etwa aus und/oder führen in Abhängigkeit von Bedingungen bestimmte Aktionen aus, die die Peripherie eines ICs steuern. Auch können sie zum Speichern von Daten dienen.

Aufgrund der zunehmenden Digitalisierung übernehmen solche ICs auch sicherheitsrelevante Aufgaben, etwa das Speichern von personenbezogenen, sicherheitsrelevanten Daten, etwa Smartcards, Bankarten etc. oder auch das Ausführen von sicherheitsrelevanten und kritischen Aktionen, etwa im Automobilbereich. Zum Übernehmen dieser Aufgaben wird der IC mit einem sogenannten Image beschrieben. Diejenigen Informationen, die dazu dienen, einen IC mit den persönlichen Daten oder auch zum Ausführen der sicherheitsrelevanten Aktionen zu beschreiben, werden nachstehend allgemein als Daten angesprochen.

ICs können etwa als EPROM, EEPROM oder dazu ähnlich ausgeführt sein, wobei die Erfindung hierauf nicht beschränkt ist. Wichtig ist, dass der genutzte IC in einer Programmiermaschine mit den für ihn bestimmten Daten beschreibbar ist.

Um dem Potenzial eines möglichen Angriffes entgegenzuwirken, müssen die für den IC bestimmten Daten, die üblicherweise in einer Entwicklungsabteilung entwickelt werden, sicher an eine Programmiermaschine, die den oder die ICs beschreibt, übertragen werden. Auch sollen Daten nicht im Nachhinein – etwa nach dem Abschließen des Beschreibens des ICs – durch einen Dritten abgreifbar sein. Das ist üblicherweise der Fall, wenn die Daten auf einem nicht-flüchtigen Speicher (zwischen)gespeichert werden.

Um die für den IC bestimmten Daten sicher zu übertragen, werden diese üblicherweise verschlüsselt übertragen und sind nur mit einem entsprechenden Schlüssel entschlüsselbar.

Aus US 2007/0038851 A1 ist eine Programmiermaschine bekannt, die dazu geeignet ist, einen IC mit sicherheitsrelevanten Daten zu beschreiben. Die sicherheitsrelevanten Daten werden in einem flüchtigen Speicher der Programmiermaschine gespeichert. Anschließend wird mit diesen Daten ein IC beschrieben.

Auch wenn an dieser bekannten Programmiermaschine selbst aufgrund des Einsetzens eines flüchtigen Speichers ein Abgreifen der Daten nach dem Beschreiben des ICs nicht möglich ist, so müssen die Daten der Programmiermaschine zum Beschreiben des ICs in entschlüsselter Form vorliegen, respektive in entschlüsselter Form an diese übertragen werden. Dies birgt die Gefahr, dass auf dem Übertragungsweg hin zur Programmiermaschine die Daten in einem nicht-flüchtigen Speicher (zwischen)gespeichert werden und im Nachhinein unbemerkt abgegriffen werden können, insbesondere da solche Programmiermaschinen üblicherweise an ein großes Computernetzwerk (etwa LAN) angeschlossen sind.

Aufgabe der Erfindung ist es vor diesem Hintergrund ein Verfahren vorzuschlagen, mit dem die Sicherheit des Prozesses zum Beschreiben von Daten auf einen IC zu erhöht ist. Zudem ist es Aufgabe der Erfindung, ein nachrüstbares System zur Ausführung des Verfahrens vorzuschlagen.

Die verfahrensbezogene Aufgabe wird gelöst durch ein Verfahren zum Beschreiben von Daten auf einen IC mit den folgenden Schritten:

1. Bereitstellen eines in verschlüsselter Form vorliegenden Datenpakets, enthaltend die auf den IC zu schreibenden Daten;
2. Laden des verschlüsselten Datenpakets;
3. Entschlüsseln des verschlüsselten Datenpaketes und Ablegen des entschlüsselten Datenpaketes ausschließlich in einem flüchtigen Speicher;

4. Bereitstellen des entschlüsselten Datenpakets in dem flüchtigen Speicher für eine Programmiermaschine;
5. Beschreiben eines oder mehrerer ICs mit für den IC bestimmten Daten des entschlüsselten Datenpakets mittels der Programmiermaschine;
6. Nach dem Fertigstellen des Beschreibens des bzw. der mehreren ICs: Löschen des flüchtigen Speichers.

Die systembezogene Aufgabe wird gelöst durch ein System zur Ausführung des erfindungsgemäßen Verfahrens, umfassend eine Programmiermaschine zum Beschreiben eines ICs und eine Computermaschine mit einem flüchtigen Speicher.

Vorteilhafte Ausgestaltungen ergeben sich aus den abhängigen Ansprüchen und der Beschreibung.

Kern der Erfindung ist es, das Datenpaket in Vorbereitung des Beschreibens des ICs zu entschlüsseln, das entschlüsselte Datenpaket jedoch ausschließlich in einem flüchtigen Speicher abzulegen und der Programmiermaschine aus diesem flüchtigen Speicher heraus bereitzustellen. Hierunter versteht sich auch das Ablegen in mehreren flüchtigen Speichern, sofern dies notwendig ist.

Um das Verständnis der Ausführung des Verfahrens zu erleichtern, wird nachstehend zunächst das System, in dem das erfindungsgemäße Verfahren durchgeführt wird, erläutert. Es versteht sich, dass die Ausführung des Verfahrens gemäß der Erfindung nicht auf ein Ausführen auf dem nachstehend beschriebenen System beschränkt ist, wenngleich dies bevorzugt ist.

Eine Programmiermaschine ist mit einer Computermaschine gekoppelt. Bevorzugt ist diese Koppelung so eingerichtet, dass die Computermaschine nur mit einer speziellen Programmiermaschine interagieren kann. Zum Sicherstellen, dass die Computermaschine nur mit einer bestimmten Programmiermaschine interagieren kann, sind dem Fachmann Verfahren und Methoden hinlänglich bekannt. Die Kopplung zwischen Computermaschine

und Programmiermaschine erhöht die Sicherheit. Der Einsatz der Computermaschine als separate Maschine zu der Programmiermaschine vereinfacht die Kontrolle über den Datenfluss zwischen Programmiermaschine und externem Netzwerk, da die Computermaschine üblicherweise kontrollierter einstellbar ist als die Programmiermaschine, etwa da auf den ausgeführten Code in der Computermaschine einfacher zugegriffen werden kann als in der Programmiermaschine. So können auch bereits bestehende Programmiermaschinen dergestalt nachgerüstet werden, dass das erfindungsgemäße Verfahren mit diesen ausgeführt werden kann. Die Computermaschine kann etwa als PC ausgeführt sein.

Die Computermaschine verfügt über eine Recheneinheit zur Ausführung von Code sowie über einen flüchtigen Speicher, etwa einen RAM.

Die Computermaschine und die Programmiermaschine sind bevorzugt in einem gemeinsamen Gehäuse untergebracht, um zu vermeiden, dass auf die Kommunikation zwischen Computermaschine und Programmiermaschine eingewirkt, bzw. diese abgegriffen werden kann. Ein physischer Zugriff auf die Computermaschine und die physischen Kommunikationsleitungen zu der Programmiermaschine ist so von außen ohne das Öffnen der Programmiermaschine nicht möglich. Das so gebildete System aus Computermaschine und Programmiermaschine wird nachstehend als Programmiermaschinensystem angesprochen.

So ist ferner bevorzugt vorgesehen, dass an dem Programmiermaschinensystem alle externen Schnittstellen, etwa USB-Anschlüsse, Laufwerke für CD's und/oder DVDs, serielle Schnittstellen, etc. deaktiviert bzw. verplombt sind.

Das Programmiermaschinensystem ist in ein Netzwerk eingebettet und hat Zugriff auf externe Netzwerkteilnehmer, etwa einen Server zum Abrufen und Speichern von Daten bzw. Datenpaketen. Bevorzugt ist vorgesehen, dass nur die Computermaschine an dieses Netzwerk angeschlossen ist; die Programmiermaschine ist lediglich an die Computermaschine angeschlossen. Die Computermaschine kann so den Zugriff der Programmiermaschine

auf das Netzwerk kontrollieren, da keine physikalische Verbindung zwischen Programmiermaschine und Netzwerk besteht; die Computermaschine ist dazwischengeschaltet. Die Kontrolle seitens der Computermaschine erfolgt mittels eines Zugriffscontrollers. Dieser Zugriffscontroller kann eine Kommunikation zwischen Programmiermaschine und Netzwerk bidirektional (dann hat die Programmiermaschine Read/Write-Zugriff auf das Netzwerk, respektive auf deren Teilnehmer) oder unidirektional (dann hat die Programmiermaschine nur Read-Zugriff auf das Netzwerk, respektive die Netzwerkteilnehmer) zulassen.

Die Computermaschine verfügt ferner über einen Zustandsautomaten, der von der Recheneinheit angesteuert wird. Der Zustandsautomat wechselt auf Befehl zwischen einem Normal-Mode und einen Secure-Mode. Der Zustandsautomat schaltet den Zugriffscontroller entsprechend des Modus um: im Normal-Mode ist die bidirektionale Kommunikation, jedenfalls auf relevante Dienste und andere Netzwerkteilnehmer möglich, in dem Secure-Mode ist nur die genannte unidirektionale Kommunikation möglich. Durch die Computermaschine wird der Zugang der Programmiermaschine zu dem Netzwerk maskiert. Des Weiteren wird beim Zurückschalten vom Secure-Mode in den Normal-Mode seitens des Zustandsautomaten der der Computermaschine zugehörige flüchtige Speicher gelöscht.

Es versteht sich, dass der hier angesprochene Zugriffscontroller und/oder der Zustandsautomat in der Recheneinheit physikalisch integriert und/oder softwaretechnisch implementiert sein kann bzw. die Aufgaben des Zugriffscontrollers und/oder des Zustandsautomaten durch die Recheneinheit mit ausgeführt werden können.

Im Detail lässt sich der erfindungsgemäße Prozess – teilweise unter Bezugnahme auf das vorstehend beschriebene System – wie folgt beschreiben:

In einem ersten Schritt (Schritt 1) wird ein Datenpaket bereitgestellt. Das Datenpaket liegt in verschlüsselter Form vor, etwa verschlüsselt mittels einer asymmetrischen Verschlüsselung. So kann eine Entwicklungsabteilung oder ein Kunde diejenigen Daten, die auf den IC geschrieben werden sollen,

von seiner Seite aus verschlüsseln. Der Betreiber der Programmiermaschine, respektive das Programmiermaschinensystem, verfügt grundsätzlich über die notwendigen Schlüssel, damit er dieses Datenpaket entschlüsseln kann.

Das Datenpaket enthält jedenfalls diejenigen Daten, die zum Beschreiben des ICs bestimmt sind. Dieses sind üblicherweise ein Image, das als Grundstruktur zum Beschreiben des bzw. der ICs bestimmt ist und ggf. prozessabhängige Individualisierungsdaten, mit denen das Image ergänzt werden kann. In diesem Fall können auch Teil des Datenpaketes Informationen sein, auf welche Art und Weise die prozessabhängigen Individualisierungsdaten in das Image eingesetzt werden sollen.

Die Individualisierungsdaten können spezifisch bereitgestellt werden; sie können jedoch auch lokal mittels einer Kundenapplikation generiert werden. Auch ist denkbar, dass diese Daten von einem externen Server abgegriffen werden, bevorzugt über eine gesicherte Verbindung.

Zum Betreiben einer Programmiermaschine sind üblicherweise zusätzliche Konfigurationsdaten notwendig, die maschinenspezifisch und ggf. auf die für den IC bestimmten Daten abgestimmt sind. Auch diese maschinenspezifischen Konfigurationsdaten können Teil des Datenpaketes sein.

Teil des Datenpaketes können auch weitere Daten sein. Das Datenpaket kann etwa in Form eines Archives vorliegen.

In einem zweiten Schritt (Schritt 2) wird das bereitgestellte Datenpaket in den Prozess geladen. Dies erfolgt mittels einer Recheneinheit, sei es einer der Programmiermaschine physikalisch zugeordneten Recheneinheit oder – bevorzugt – mittels der Recheneinheit der in dem vorstehend beschriebenen Programmiermaschinensystem genannten Computermaschine.

Das verschlüsselte Datenpaket kann etwa auf einem Server in einer Netzwerkumgebung liegen. Zum Laden des Datenpaketes wird dieses von dem Server abgerufen.

Im dritten Schritt (Schritt 3) wird das geladene, verschlüsselte Datenpaket entschlüsselt. Das entschlüsselte Datenpaket wird ausschließlich in einem flüchtigen Speicher abgelegt. Ein Schreiben auf einen nicht-flüchtigen Speicher ist ausgeschlossen, etwa das Schreiben auf eine Festplatte. Denn: Auch wenn auf einem nicht-flüchtigen Speicher Speicherplatz freigegeben/gelöscht wird, so ist es für einen Angreifer möglich bzw. besteht die Gefahr, dass die vormals gespeicherten Daten rekonstruiert werden können, insbesondere dann, wenn nicht der gesamte Speicher überschrieben wird, sondern nur ein Teil desselben. Dies ist bei nicht-flüchtigen Speichersystemen üblich.

Bevorzugt ist der flüchtige Speicher als RAM ausgebildet, etwa als Teil der vorstehend genannten Computermaschine. So kann die Entschlüsselung lokal auf der Computermaschine erfolgen; das Ergebnis wird dann ausschließlich im RAM gehalten.

Vor dem Schritt 3 oder nach dem Schritt 3 ist bevorzugt vorgesehen, das Programmiermaschinensystem in den oben bereits erläuterten Secure-Mode zu schalten, um zu verhindern, dass entschlüsselte Daten über die Programmiermaschine an das Netzwerk gesendet werden.

Im vierten Schritt (Schritt 4) wird das entschlüsselte Datenpaket in dem flüchtigen Speicher für die Programmiermaschine bereitgestellt. Der Programmiermaschine wird somit Zugriff auf den flüchtigen Speicher gewährt.

Dies kann etwa dadurch umgesetzt werden, dass in dem flüchtigen Speicher ein Laufwerk emuliert wird, welches für die Programmiermaschine freigegeben wird. Durch das Emulieren eines Laufwerkes in dem flüchtigen Speicher, auf das der Programmiermaschine Zugriff gewährt wird, ist eine einfache Einbindung des erfindungsgemäßen Verfahrens auch in bestehende Produktionslinien ermöglicht. Programmiermaschinen greifen üblicherweise auf externe Laufwerke zu, um die erforderlichen Datenpakete, respektive Daten abzurufen, um einen oder mehrere ICs zu beschreiben.

Bevorzugt ist vorgesehen, dass das emulierte Laufwerk nur an der computermaschinenseitigen Schnittstelle zu der Programmiermaschine freigegeben ist und nicht an der computermaschinenseitigen Schnittstelle zu einem externen Netzwerk.

Sollte der flüchtige Speicher dennoch in ein Netzwerk eingebunden sein, der auch Netzwerkteilnehmer außerhalb des Programmiermaschinensystems umfasst, kann vorgesehen sein, das emulierte Laufwerk mit einem nur der Programmiermaschine bekannten Schlüssel, etwa in Form eines Passwortes, zu sichern bzw. zu verschlüsseln.

In einem fünften Schritt (Schritt 5) werden mittels der Programmiermaschine ein oder mehrere ICs mit den für den bzw. die ICs bestimmten Daten des entschlüsselten Datenpaketes beschrieben. Das Beschreiben erfolgt auf übliche Weise seitens der Programmiermaschine und ist dem Fachmann hinlänglich bekannt.

So ist bevorzugt vorgesehen, dass das entschlüsselte Datenpaket bzw. hieraus extrahierte, entschlüsselte Daten nur in flüchtigen Speichern der Programmiermaschine abgelegt werden. Andere Maßnahmen zum Sicherstellen, dass die entschlüsselten Daten nicht aus der Programmiermaschine durch einen Dritten abgegriffen werden können, sind möglich.

Je nach Programmiermaschine besteht zudem die Möglichkeit, dass das Datenpaket ausschließlich aus denjenigen Daten besteht, mit denen der IC beschrieben wird; etwaige zusätzliche Konfigurationsdaten können auch auf separatem Wege der Programmiermaschine zur Verfügung gestellt werden. Bevorzugt ist vorgesehen, dass, sollten die Konfigurationsdaten der Programmiermaschine auf separatem Wege zur Verfügung gestellt werden, dieses analog zu dem hier beschriebenen Prozess erfolgt, nämlich, dass auch die Konfigurationsdaten grundsätzlich verschlüsselt gespeichert sind und nach dem hier vorgeschlagenen Verfahren in einen flüchtigen Speicher entschlüsselt werden und nur aus dem flüchtigen Speicher heraus der Programmiermaschine zur Verfügung gestellt werden.

In einem sechsten Schritt (Schritt 6) ist vorgesehen, dass nach dem Fertigstellen des Beschreibens des bzw. der mehreren ICs der flüchtige Speicher, in dem entschlüsselte Daten vorliegen, gelöscht wird. Mit Löschen der Daten ist gemeint, dass die Daten durch einen Angreifer nicht mehr aus dem Speicher abrufbar bzw. rekonstruierbar sind. Dies erfolgt etwa durch ein Überschreiben des vollständigen flüchtigen Speichers mit anderem Inhalt. Auch kann ein Erden des flüchtigen Speichers vorgesehen sein, sodass dieser stromlos geschaltet wird. Dies ist insbesondere automatisch der Fall, sollte ein Angreifer physisch auf den flüchtigen Speicher zugreifen und diesen ausbauen. Bei einem Ausbau wird die Stromversorgung unterbrochen, sodass der Inhalt des flüchtigen Speichers verloren geht.

Nach dem Löschen des Speichers wird das System wieder in den Normal-Mode geschaltet. Das Schalten in den Normal-Modus kann das Löschen des flüchtigen Speichers auch beinhalten (s. o.).

Üblicherweise werden die Daten, mit denen der IC beschrieben werden soll, ggf. ergänzt durch prozessabhängige Individualisierungsdaten, getrennt von dem Schreibvorgang der Programmiermaschine entwickelt, um ein Beschreiben eines ICs auf verschiedenen Programmiermaschinentypen zu ermöglichen. Die zum Betrieb eines Programmiermaschinentyps notwendigen maschinentypspezifischen Konfigurationsdaten werden daher getrennt im Rahmen eines Mustererstellungsprozesses an einem Individuum des jeweiligen Programmiermaschinetyps entwickelt. Hierzu werden die auf den IC zu schreibenden Daten testweise mit der Programmiermaschine auf den IC geschrieben und der IC anschließend validiert, nämlich, ob der Schreibvorgang zu dem gewünschten Ergebnis geführt hat. Ist dies nicht der Fall, werden die Konfigurationsdatenangepasst, ein IC wird abermals testweise beschrieben und validiert etc.

An diesen Mustererstellungsprozess schließt sich ein Serienfertigungsprozess, in dem eine Vielzahl gleicher bzw. sehr ähnlicher ICs beschrieben werden (letzteres ist der Fall, wenn Teil der auf den IC zu beschreibenden Daten prozessabhängige Individualisierungsdaten sind), an. Bevorzugt ist vorgesehen, dass auch während dieses Mustererstellungsprozesses die entschlüsselten Daten zum Beschreiben des ICs als auch die angepassten

Konfigurationsdaten nur in einem flüchtigen Speicher abgelegt werden und nicht auf einen nicht-flüchtigen Speicher gespeichert werden.

Im Detail ist in dem Mustererstellungsprozess zunächst vorgesehen (Schritt A), die vorstehend beschriebenen Schritte 1 bis 4 durchzuführen, mit der Maßgabe, dass es sich bei dem angesprochenen Datenpaket um die auf den oder die ICs zu schreibenden Daten handelt, mithin das Image, ggf. ergänzt um prozessabhängige Individualisierungsdaten.

In einem nächsten Schritt (Schritt B) werden die Konfigurationsdaten, etwa maschinentypspezifische Konfigurationsdaten, bereitgestellt, mithin ggf. erstellt. Auch diese Konfigurationsdaten werden nur in dem flüchtigen Speicher abgelegt, und zwar dergestalt, dass sie der Programmiermaschine bereitgestellt werden. Das Bereitstellen kann analog zum Schritt 4 erfolgen.

In einem weiteren Schritt (Schritt C) schließt sich üblicherweise ein Funktionstest an, nach dem in Abhängigkeit des Ausganges die im Schritt B bereitgestellten Konfigurationsdaten angepasst werden. Sind die Konfigurationsdaten final bereitgestellt, werden die Konfigurationsdaten und die für das Beschreiben des ICs bestimmten Daten zusammengeführt. Diese zusammengeführten Daten werden auch als Job angesprochen. Die Zusammenführung kann etwa in einem Archiv erfolgen.

In einem nächsten Schritt (Schritt D) wird der Job verschlüsselt. Dieser verschlüsselte Job wird in einem nicht-flüchtigen Speicher abgelegt. Dies kann etwa in einer Netzwerkumgebung auf einem Server sein. Auf diese Weise ist der verschlüsselte Job von allen Netzwerkteilnehmern, etwa auch von anderen Programmiermaschinensystemen, abrufbar. Sollte der Job auch von anderen Programmiersystemen abrufbar sein sollen, ist es notwendig, den Job auf eine Art und Weise zu verschlüsseln, dass auch andere Programmiersysteme diesen entschlüsseln können.

Anschließend (Schritt E) wird der flüchtige Speicher, in dem die entschlüsselten Daten zuvor abgelegt waren, gelöscht. Dies kann analog zu Schritt 6 erfolgen.

Es kann vorgesehen sein, dass im Schritt D der verschlüsselte Job nicht sofort auf einen externen Netzwerkteilnehmer gespeichert wird, sondern zunächst nur lokal. Anschließend wird der Schritt E ausgeführt (Löschen des flüchtigen Speichers). Erst nachdem der flüchtige Speicher gelöscht worden ist, wird eine Verbindung zu einem Netzwerkteilnehmer, etwa einem Server, aufgebaut, um den verschlüsselten Job zu speichern.

Zum Durchführen des Serienfertigungsprozesses werden dann die vorstehend beschriebenen Schritte 1 bis 6 durchgeführt, wobei es sich bei dem dort angesprochenen Datenpaket um den im Schritt C des Mustererstellungsprozesses erstellten Job handelt.

Auch in dem Mustererstellungsprozess kann vorgesehen sein, dass bevor die verschlüsselten Daten entschlüsselt werden, das Programmiermaschinensystem in den Secure-Mode geschaltet wird und nach dem Schritt E wieder in den Normal-Mode geschaltet wird.

Die Erfindung wird anhand der beiliegenden Figur näher erläutert. Sie zeigt:

Fig. 1: Ein schematisches Schaubild des Programmiermaschinensystems, eingebettet in ein Netzwerk.

Figur 1 zeigt ein Programmiermaschinensystem 1, eingebettet in einen Produktionsprozess. In einer Entwicklungsabteilung 2, etwa bei einem Kunden, werden Daten, mit denen ein IC 3 beschrieben werden soll, entwickelt. Diese Daten werden auch als Image angesprochen. Zusätzlich können prozessabhängige Individualisierungsdaten vorgesehen sein, die das Image, mit dem ein spezieller IC 3 beschrieben werden soll, in Details abgewandelt wird, etwa wenn es sich bei den Daten um personenindividuelle Daten handelt. So können prozessabhängige Individualisierungsdaten etwa auf die Vielzahl des ICs zu schreibende, verschiedene Pin-Codes sein. Diese Daten werden an ein Produktionssystem 4 übergeben, und zwar in verschlüsselter Form. Als Verschlüsselungsmethode ist eine asymmetrische Verschlüsselung vorgesehen. Verschlüsselt werden die Daten mit dem seitens der Produktion 4 herausgegebenen Public Key.

Die verschlüsselten Daten werden an das Produktionssystem 4 übertragen und in dem Produktionssystem 4 auf einem Server 5 gespeichert. Hierfür verfügt der Server 5 über nicht-flüchtige Speicher zum sicheren Speichern der verschlüsselten Daten.

In dem Produktionssystem 4 sollen eine Vielzahl von ICs 3 mit den seitens der Entwicklungsabteilung 2 entwickelten Daten in einem Serienfertigungsprozess beschrieben werden. Hierfür muss zunächst in einem Mustererstellungsprozess eine Programmiermaschine 6 entsprechend eingerichtet werden. Die Einrichtung erfolgt über Konfigurationsdaten, die durch einen Einrichter vorgegeben werden. Zum Konfigurieren der Programmiermaschine 6 ist es notwendig, testweise ein oder mehrere ICs 3 mit den für den IC 3 bestimmten Daten zu beschreiben, und zwar unter Verwendung von bestimmten Konfigurationsdaten, um zu prüfen, ob die eingestellten Konfigurationsdaten in Kombination mit den für den IC 3 bestimmten Daten zu dem gewünschten Ergebnis führen. Mit den Konfigurationsdaten werden etwa der Programmiermaschine 6 zugehörige Peripherie 7 angesteuert, etwa Steuergeräte zum Beschreiben des ICs 3, zum Durchführen von Validierungen, etc.

Um die auf dem Server 5 gespeicherten, von der Entwicklungsabteilung 2 übermittelten Daten, die auf dem Server 5 in verschlüsselter Form vorliegen, entschlüsselt der Programmiermaschine 6 zum Beschreiben des ICs 3 zur Verfügung zu stellen, ist eine Computermaschine 8 Teil des Programmiermaschinensystems 1. Die Computermaschine 8 ist mit der Programmiermaschine 6 gekoppelt. Die Computermaschine 8 ist dergestalt eingerichtet, dass sie nur mit einer ganz speziellen Programmiermaschine 6 arbeiten kann; das Anschließen einer anderen Programmiermaschine 6 führt zu einer Nicht-Funktion der Computermaschine 8, jedenfalls in dem Funktionsumfang, wie er in dem hier beschriebenen Verfahren vorgesehen ist. Für die Kopplung notwendige Parameter und Verfahren sind dem Fachmann hinreichend bekannt.

Eine Kommunikation zwischen der Computermaschine 8 und der Programmiermaschine 6 erfolgt kabelbasiert, etwa über LAN. Um einen Abgriff der

ausgetauschten Daten zwischen Programmiermaschine 6 und Computermaschine 8 zu unterbinden, ist die Computermaschine 8 physikalisch Teil des Programmiermaschinensystems 1. Das bedeutet, dass die Computermaschinen 8 in ein und demselben Gehäuse untergebracht sind und gegen einen Zugriff von außen geschützt sind.

Die Programmiermaschine 6 hat über die Computermaschine 8 ebenfalls Zugriff auf das Netzwerk außerhalb des Programmiermaschinensystems 1, etwa auf den Server 5. Der Zugriff von Seiten der Programmiermaschine 6 auf das Netzwerk außerhalb des Programmiermaschinensystems 1 wird jedoch durch die Computermaschine 8 mittels eines Zugriffcontrollers 11 kontrolliert. Der Zugriffcontroller 11 ist angeschlossen an einen Zustandsautomaten 12. Es versteht sich, dass der Zugriffscontroller 11 und der Zustandsautomat 12 physikalisch auch Teil der Recheneinheit 9 sein können; zu Erklärungszwecken sind diese getrennt in der Figur 1 gezeichnet.

Der Zustandsautomat 12 schaltet das Programmiermaschinensystem 1 zwischen zwei Zuständen: dem Normal-Mode und dem Secure-Mode.

Im Normal-Mode ist der Zugriffscontroller 11 dergestalt eingestellt, dass die Programmiermaschine 6 bidirektional mit dem Netzwerk außerhalb des Programmiermaschinensystems 1, etwa mit dem Server 5, kommunizieren kann; es besteht somit ein Read/Write-Zugriff.

Erhält der Zustandsautomat 12 von der Recheneinheit 9 die Information, dass in den Secure-Mode umgeschaltet werden soll, übermittelt der Zustandsautomat 12 dem Zugriffscontroller 11, dass die Zugriffsrechte der Programmiermaschine 6 dahingehend einzuschränken sind, sodass die Programmiermaschine 6 nicht mehr außerhalb des Programmiermaschinensystems 1 schreiben darf; für die Programmiermaschine 6 wird nur ein Read-Zugriff auf das Netzwerk außerhalb des Programmiermaschinensystems 1 gewährt.

Zum Abrufen der Daten von dem Server 5 werden diese von Seiten der Computermaschine 8 mittels einer Recheneinheit 9 abgerufen und in einen

flüchtigen Speicher 10, hier einem RAM der Computermaschine 8, abgelegt.

Dann wird der Secure-Mode des Programmiermaschinensystems 1 eingeschaltet. Anschließend werden die in dem flüchtigen Speicher 10 abgelegten verschlüsselten Daten durch die Recheneinheit 9 entschlüsselt.

Die entschlüsselten Daten werden ebenfalls ausschließlich in dem flüchtigen Speicher 10 abgelegt. Durch das Vermeiden des Speicherns der entschlüsselten Daten auf einem nicht-flüchtigen Speicher wird der Gefahr begegnet, dass ein Angreifer die entschlüsselten Daten rekonstruieren kann, auch wenn die Daten in dem nicht-flüchtigen Speicher als gelöscht gelten.

Die in dem flüchtigen Speicher 10 entschlüsselten Daten werden alsdann der Programmiermaschine 6 bereitgestellt. Hierfür wird ein Laufwerk in dem flüchtigen Speicher 10 emuliert, auf das die Programmiermaschine 6 mit ihrer Recheneinheit 13 zugreifen kann. Zuvor an die Computermaschine 8 ebenfalls übergebene und in dem flüchtigen Speicher 10, respektive in dem emulierten Laufwerk abgelegte Konfigurationsdaten werden ebenfalls durch die Recheneinheit 13 der Programmiermaschine 6 abgerufen.

Ausgestattet mit diesen Daten beschreibt die Programmiermaschine 6 den IC 3 mit den für den IC 3 bestimmten Daten; für die Steuerung 7 der Programmiermaschine 6 werden die Konfigurationsdaten genutzt.

Haben die Konfigurationsdaten zu dem gewünschten Ergebnis (der IC 3 ist ordnungsgemäß beschrieben worden) geführt, werden die für das Beschreiben des ICs 3 bestimmten Daten und die Konfigurationsdaten, die beide in dem flüchtigen Speicher 10 der Computermaschine 8 abgelegt sind, zu einem Job zusammengefasst, und zwar in einem Archiv.

Anschließend wird der Job respektive das Archiv verschlüsselt und seitens der Computermaschine 8 auf dem Server 5 abgelegt.

Anschließend sendet die Recheneinheit 9 an den Zustandsautomaten 12 den Befehl, in den Normal-Mode zurückzuschalten. Daraufhin löscht der

Zustandsautomat 12 den flüchtigen Speicher 10, etwa indem er diesen erdet, und sendet an den Zugriffscontroller 11 einen Befehl, dass ein Schreibzugriff für die Programmiermaschine 6 für das Netzwerk außerhalb des Programmiermaschinensystems 1 zuzulassen ist. Auf diese Weise sind die entschlüsselten Daten nicht abgreifbar durch einen Dritten.

Angedeutet ist auch eine Verbindung des Zustandsautomaten 12 zu dem schematischen Gehäuse des Programmiermaschinensystems 1: Wird das Gehäuse geöffnet oder tritt ein anderer unerwarteter Fehler auf, erkennt dies der Zustandsautomat 12 und schaltet automatisch in den Normal-Mode um (Löschen des flüchtigen Speichers 10 und Umschalten des Zugriffscontrollers 11 in den bidirektionalen Modus.

Der Serienfertigungsprozess erfolgt auf die gleiche Art und Weise mit dem einzigen Unterschied, dass seitens der Computermaschine 8 nicht nur die für den IC 3 bestimmten Daten seitens des Servers 5 abgerufen werden, sondern insgesamt den in dem Mustererstellungsprozess erstellten Job. Der Job umfasst die für das Beschreiben des ICs 3 bestimmten Daten sowie die Konfigurationsdaten für die Steuerung 7 der Peripherie der Programmiermaschine 6.

Dieser Job wird – nachdem der Secure-Mode durch den Zustandsautomaten 12 aktiviert wurde – in dem flüchtigen Speicher 10 entschlüsselt abgelegt und der Programmiermaschine 6 zur Verfügung gestellt, sodass diese unter Verwendung der bereitgestellten Konfigurationsdaten den IC 3 mit den für ihn bestimmten Daten beschreiben kann.

Die Erfindung ist anhand eines Ausführungsbeispiels beschrieben worden. Ohne den Schutzbereich, beschrieben durch die Ansprüche, zu verlassen, ergeben sich für den Fachmann zahlreiche weitere Ausgestaltungen, den Erfindungsgedanken zu verwirklichen, ohne dass diese im Rahmen dieser Ausführungen näher erläutert werden müssten.

Bezugszeichenliste

1	Programmiermaschinensystem
2	Entwicklungsabteilung
3	IC
4	Produktionssystem
5	Server
6	Programmiermaschine
7	Steuerung, der Programmiermaschine
8	Computermaschine
9	Recheneinheit, der Computermaschine
10	Flüchtiger, Speicher für Computermaschine
11	Zugriffscontroller
12	Zustandsautomat
13	Recheneinheit der Programmiermaschine

Patentansprüche

1. Verfahren zum Beschreiben von Daten auf einen IC (3) mit den folgenden Schritten:

1. Bereitstellen eines in verschlüsselter Form vorliegenden Datenpakets, enthaltend die auf den IC (3) zu schreibenden Daten;
2. Laden des verschlüsselten Datenpakets;
3. Entschlüsseln des verschlüsselten Datenpaketes und Ablegen des entschlüsselten Datenpaketes ausschließlich in einem flüchtigen Speicher (10);
4. Bereitstellen des entschlüsselten Datenpakets in dem flüchtigen Speicher (10) für eine Programmiermaschine (6);
5. Beschreiben eines oder mehrerer ICs (3) mit für den IC (3) bestimmten Daten des entschlüsselten Datenpakets mittels der Programmiermaschine (6);
6. Nach dem Fertigstellen des Beschreibens des bzw. der mehreren ICs (3): Löschen des flüchtigen Speichers (10).

2. Verfahren nach Anspruch 1, **dadurch gekennzeichnet**, dass das bereitgestellte verschlüsselte Datenpaket einerseits ein Image, das zum Beschreiben des bzw. der ICs (3) bestimmt ist, ggf. ergänzt um prozessabhängige Individualisierungsdaten, und andererseits Konfigurationsdaten zum Betrieb der Programmiermaschine (6) während des Beschreibens umfasst.

3. Verfahren nach Anspruch 1, **dadurch gekennzeichnet**, dass es sich bei dem flüchtigen Speicher (10) um einen RAM einer Computermaschine (8) handelt, die mit der Programmiermaschine (6) gekoppelt ist.

4. Verfahren nach Anspruch 1, **dadurch gekennzeichnet**, dass das Bereitstellen der Daten in Schritt 4 durch das Emulieren eines Laufwerkes in dem flüchtigen Speicher (10) erfolgt.

5. Verfahren nach Anspruch 1, **dadurch gekennzeichnet**, dass die Programmiermaschine (6) in eine Netzwerkumgebung eingebunden ist und wobei in einem Normal-Mode die Programmiermaschine (6) Daten auf einen Netzwerkteilnehmer schreiben kann (Read/Write-Zugriff) und in einem Secure-Mode ein Schreiben nicht möglich ist (Read-Only-Zugriff) und wobei zu Beginn des Verfahrens nach Anspruch 1 der Normal-Mode aktiviert ist, vor Schritt 3 in den Secure-Mode gewechselt wird und nach Schritt 6 in den Normal-Mode zurückgeschaltet wird.

6. Verfahren nach Anspruch 1, **dadurch gekennzeichnet**, dass zum Einrichten eines Serienfertigungsprozesses in einem vorbereitenden Schritt ein Mustererstellungsprozess durchgeführt wird, aufweisend die folgenden Schritte:

A) Durchführen der Schritte 1 bis 4 des Anspruchs 1, wobei es sich bei dem genannten Datenpaket um die Daten handelt, die zum Beschreiben des oder der ICs (3) bestimmt sind;
B) Bereitstellen von Konfigurationsdaten zum Betrieb der Programmiermaschine (6) im flüchtigen Speicher (10) für die Programmiermaschine (6);
C) Zusammenführen der Konfigurationsdaten und der Daten, die zum Beschreiben des oder der ICs (3) bestimmt sind, in einen Job;
D) Verschlüsseln des Jobs und Ablegen des verschlüsselten Jobs in einen nicht-flüchtigen Speicher;
E) Löschen des flüchtigen Speichers (10);

und wobei anschließend zum Durchführen des Serienfertigungsprozesses die Schritte des Anspruchs 1 durchgeführt werden, wobei es sich bei dem genannten Datenpaket um den im Schritt C) erstellten Job handelt.

7. System zur Ausführung des Verfahrens nach einem der Ansprüche 1 bis 6, umfassend eine Programmiermaschine (6) zum Beschreiben eines ICs (3) und eine Computermaschine (8) mit einem flüchtigen Speicher (10).

8. System nach Anspruch 7, **dadurch gekennzeichnet**, dass die Computermaschine (8) in dem Gehäuse der Programmiermaschine (6) angeordnet ist und die Computermaschine (8) dazu eingerichtet ist, wenn unerwartet in das Gehäuse eingedrungen wird oder ein anderes unerwartetes Verhalten detektiert wird, den flüchtigen Speicher (10) zu löschen.

9. System nach Anspruch 7 oder 8, **dadurch gekennzeichnet**, dass die Computermaschine (8) über eine erste Schnittstelle und eine weitere, zweite Schnittstelle zum Datenaustausch verfügt und wobei die Computermaschine (8) und die Programmiermaschine (6) über die erste Schnittstelle miteinander gekoppelt sind und die Computermaschine (8) dazu eingerichtet ist, dass externe Daten, die der Programmiermaschine (6) zur Verfügung gestellt werden sollen, über die zweite Schnittstelle durch die Computermaschine (8) abgerufen werden und über die erste Schnittstelle an die Programmiermaschine (6) weitergegeben werden und Daten, die die Programmiermaschine (6) externen Maschinen zur Verfügung stellen soll, über die erste Schnittstelle an die Computermaschine (8) gegeben werden und über die zweite Schnittstelle an die externe Maschine weitergegeben werden und wobei die Computermaschine (8) ferner dazu eingerichtet ist, in einem Secure-Mode zu unterbinden, dass Daten, die die Programmiermaschine (6) externen Maschinen zur Verfügung stellen möchte, an die externe Maschine weitergegeben werden.

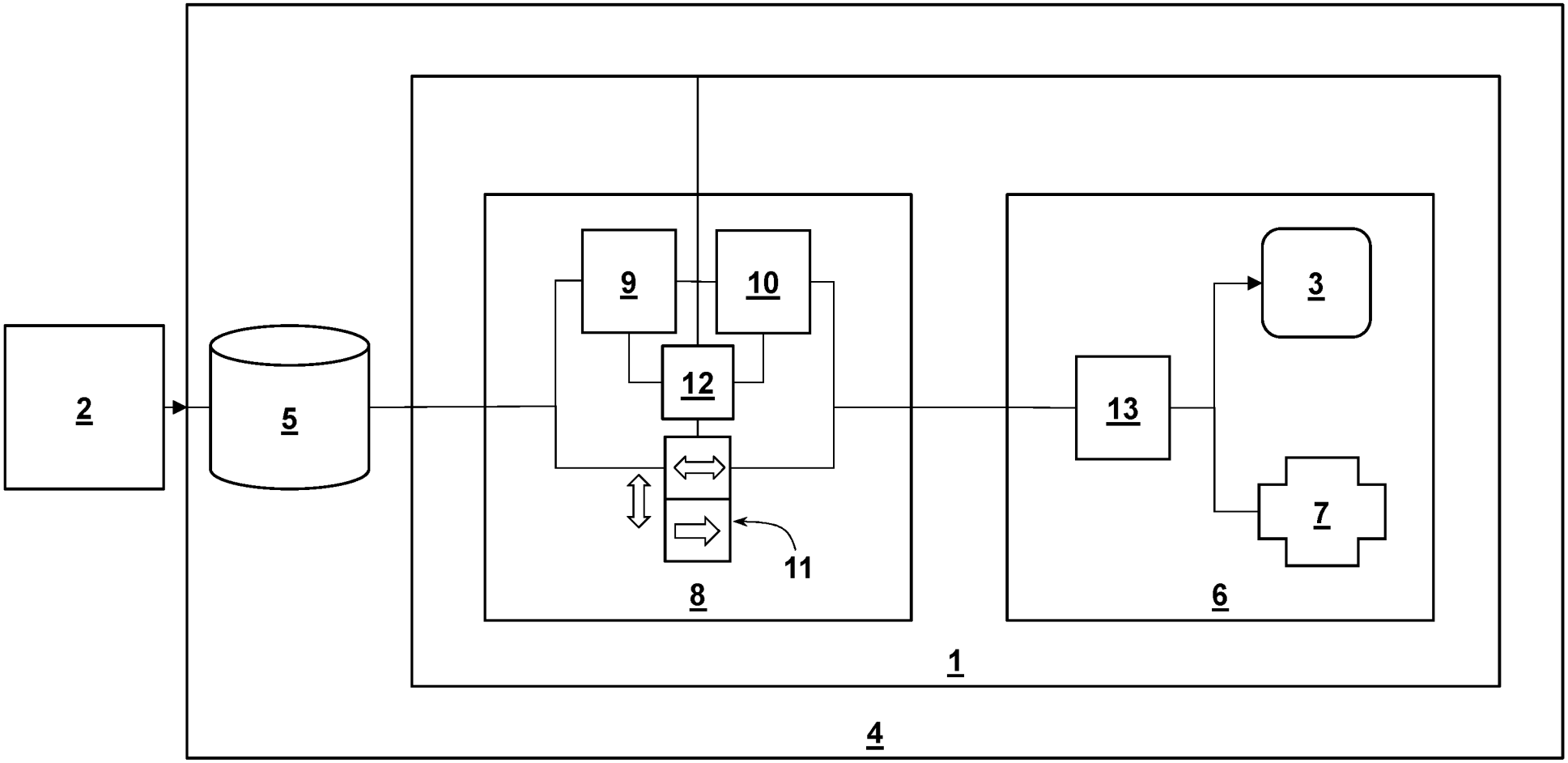

Fig. 1

Klassifikation des Anmeldungsgegenstands gemäß IPC:
G06F 21/57 (2013.01); ***G06F 21/64*** (2013.01); ***G06F 21/70*** (2013.01); ***G06F 21/71*** (2013.01); ***G06F 21/78*** (2013.01)

Klassifikation des Anmeldungsgegenstands gemäß CPC:
G06F 21/57 (2022.02); ***G06F 21/64*** (2015.11); ***G06F 21/70*** (2015.11); ***G06F 21/71*** (2015.11); ***G06F 21/78*** (2015.11)

Recherchierter Prüfstoff (Klassifikation):
G06F

Konsultierte Online-Datenbank:
WPI; EPODOC; TXTEN; TXTDE; NPL; INSPEC; Internet

Dieser Recherchenbericht wurde zu den am **03.02.2023** eingereichten Ansprüchen **1-9** erstellt.

Kategorie*)	Bezeichnung der Veröffentlichung: Ländercode, Veröffentlichungsnummer, Dokumentart (Anmelder), Veröffentlichungsdatum, Textstelle oder Figur soweit erforderlich	Betreffend Anspruch
Y	US 2007038851 A1 (BABARIA AMIT G. et al.) 15. Februar 2007 (15.02.2007) Siehe besonders Zusammenfassung; Fig. 1-3 und die dazugehörigen Figurenbeschreibungen; Paragraphen 1-7, 14-20; Patentansprüche 1-3, 8 und 9	1-4, 7, 8
A		5, 6, 9
Y	DE 102006006109 A1 (BOSCH GMBH ROBERT) 16. August 2007 (16.08.2007) Siehe besonders Zusammenfassung; Fig. 1 und die dazugehörige Figurenbeschreibung; Paragraphen 15, 24, 25, 36-42; Patentansprüche 1 bis 15	1-4, 7, 8
A		5, 6, 9
A	DE 102020133738 A1 (INTEL CORP.) 28. Oktober 2021 (28.10.2021) Siehe besonders Zusammenfassung; Fig. 1 und die dazugehörige Figurenbeschreibung, Paragraphen 4, 5, 7, 25, 30 und 36	1, 2, 4, 5
A	WO 03058409 A2 (SCM MICROSYSTEMS GMBH) 17. Juli 2003 (17.07.2003) Siehe Zusammenfassung; Fig. 1; Seite 2, Zeile 9 bis Seite 4, Zeile 11	1

Datum der Beendigung der Recherche:	Seite 1 von 1	Prüfer(in):
25.08.2023		KÖGL Christian

*) **Kategorien** der angeführten Dokumente:

- **X** Veröffentlichung **von besonderer Bedeutung**: der Anmeldungsgegenstand kann allein aufgrund dieser Druckschrift nicht als neu bzw. auf erfinderischer Tätigkeit beruhend betrachtet werden.
- **Y** Veröffentlichung **von Bedeutung**: der Anmeldungsgegenstand kann nicht als auf erfinderischer Tätigkeit beruhend betrachtet werden, wenn die Veröffentlichung mit einer oder mehreren weiteren Veröffentlichungen dieser Kategorie in Verbindung gebracht wird und diese **Verbindung für einen Fachmann naheliegend** ist.
- **A** Veröffentlichung, die den allgemeinen **Stand der Technik** definiert.
- **P** Dokument, das von **Bedeutung** ist (Kategorien **X** oder **Y**), jedoch **nach dem Prioritätstag** der Anmeldung veröffentlicht wurde.
- **E** Dokument, das **von besonderer Bedeutung** ist (Kategorie **X**), aus dem ein „**älteres Recht**“ hervorgehen könnte (früheres Anmeldedatum, jedoch nachveröffentlicht, Schutz ist in Österreich möglich, würde Neuheit in Frage stellen).
- **&** Veröffentlichung, die Mitglied der selben **Patentfamilie** ist.