Informatik

W. A. Halan

Herausf

durch

Informatik aktuell

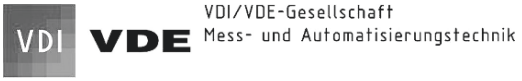
Herausgeber: W. Brauer
im Auftrag der Gesellschaft für Informatik (GI)

Wolfgang A. Halang (Hrsg.)

Herausforderungen durch Echtzeitbetrieb

Echtzeit 2011

Fachtagung des gemeinsamen Fachausschusses
Echtzeitsysteme von
Gesellschaft für Informatik e.V. (GI),
VDI/VDE-Gesellschaft für Mess- und Automatisierungstechnik (GMA) und
Informationstechnischer Gesellschaft im VDE (ITG)
Boppard, 3. und 4. November 2011



CR Subject Classification (2001): C3, D.4.7

ISBN 978-3-642-24657-9 e-ISBN 978-3-642-24658-6
DOI 10.1007/978-3-642-24658-6
Springer Heidelberg Dordrecht London New York

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über http://dnb.d-nb.de abrufbar.

Einbandentwurf: WMXDesign GmbH, Heidelberg

Gedruckt auf säurefreiem Papier

Springer ist Teil der Fachverlagsgruppe Springer Science+Business Media
(www.springer.com)

Vorwort

Als Leitthema für die Fachtagung „Echtzeit 2011“ hat das Programmkomitee die Herausforderungen gewählt, die der Echtzeitbetrieb an Entwurf, Aufbau, Programmierung, Implementierung sowie Einsatz von Rechensystemen stellt. Unter den aus den eingegangenen Vorschlägen zur Aufnahme in den vorliegenden Band ausgewählten Beiträgen beschäftigt sich in der Tat eine Gruppe mit Entwurfsverfahren für Echtzeitsysteme. Weitere Themenschwerpunkte bilden einem aktuellen Trend entsprechend Mehrkernsysteme, deren höhere Leistungsfähigkeit durch kompliziertere Handhabbarkeit erkauft wird, sowie die insbesondere unter Einsatzgesichtspunkten bedeutende Skalier- und Konfigurierbarkeit.

Die Auftaktsitzung der Tagung ist jedoch Fragen der funktionalen Sicherheit gewidmet. Ein aktueller – eigentlich unvernünftiger – Trend in der Automatisierungstechnik ist es, Kommunikationsnetze in Ethernet-Technik oder funkgestützt aufzubauen und dann noch mit dem Internet zu verbinden, so dass sie von außen leicht angegriffen oder ausgespäht werden können. Der erste Beitrag zeigt daher auf, wie sich unter Einhaltung gegebener Echtzeitbedingungen Daten abhörsicher austauschen lassen. Weil in eingebetteten Systemen häufig auch Programme mit Sicherheitsverantwortung implementiert sind, die dann gegen Fehlfunktionen des übrigen Systems geschützt werden müssen, befasst sich der zweite Beitrag mit der Frage, wie sicherheitsgerichtete Software in AUTOSAR-Systemen isoliert werden kann, ohne dass deren gesamte System-Software auch hohen Sicherheitsanforderungen genügen müsste.

Zunehmend bieten Hersteller von Mikrorechnern diese mit mehreren Prozessorkernen an. Da deren Potential in eingebetteten und Echtzeitsystemen jedoch nicht ohne weiteres genutzt werden kann, beschäftigt sich ein Beitrag der zweiten Sitzung damit, modellbasiert statische Zuteilungen sicherheitskritischer Tasks mit harten Echtzeitanforderungen zu mehreren Kernen automatisch zu erzeugen. Wie der Ablauf derartiger Tasks auf Mehrkernprozessoren unter Hinnahme gewisser Einschränkungen durch gewöhnliche, unveränderte Betriebssysteme gesteuert werden kann, zeigt ein anderer Beitrag. Schließlich wird in dieser Sitzung noch ein Beispiel für den Einsatz von Mehrkernprozessoren in eingebetteten Systemen im Zusammenspiel mit Virtualisierung gegeben, und zwar zur Integration von Multimediabordsystemen in Automobilen.

Traditionell ist eine Sitzung der Tagung dem Bereich Ausbildung gewidmet. Zunächst werden zum Einsatz in Informatikunterricht und -studium entwickelte anschauliche Medien vorgestellt, mit denen komplexe Echtzeitsysteme, ihr Zeitverhalten und Zusammenwirken mit der Umgebung besser begreifbar gemacht werden soll. Dem Trend hin zu individuellen Produkten folgend, wird anschließend die Ausbildung von Studenten mittels Praktika und kleinerer Entwicklungsprojekte an flexiblen, industriellen Fertigungssystemen beschrieben. Ebenfalls in dieser Sitzung werden die beiden studentischen Abschlussarbeiten präsentiert, die als Sieger aus dem erneut vom Fachausschuss ausgelobten

Graduiertenwettbewerb hervorgegangen sind. In der ersten Arbeit wird ein Echtzeitverfahren zur Orientierung und Navigation mobiler, autonomer Roboter mit vorhersagbarer Laufzeit auf einer konkreten Plattform umgesetzt und in einer realen Testumgebung evaluiert. Die zweite Arbeit stellt ein FPGA-gestütztes dediziertes System vor, das die numerische Lösung durch Differentialgleichungen beschriebener zeitkontinuierlicher Simulationsmodelle der Umgebungen von Steuergeräten bei Hardware-in-the-Loop-Tests im Echtzeitbetrieb erlaubt.

Die erste Sitzung des zweiten Workshop-Tages befasst sich mit Entwurfsverfahren. Um das Verhalten von Echtzeitsystemen transparent, genau, eindeutig, konsistent und leicht verständlich darzustellen, wird eine tabellarische Beschreibungsmethode vorgestellt, die es weiterhin erlaubt, Verhalten hierarchisch zu strukturieren, zu verfeinern und bzgl. verschiedener systemtechnischer Aspekte zu differenzieren. Die Ergebnisse einer den Zusammenhang zwischen Energiebedarf, Dienstgüte und Systemleistung bei der Substitution von Ressourcen in Software-Systemen untersuchenden Fallstudie werden im Anschluss präsentiert. Der Autor des dritten Beitrags dreht den Spieß um. Er adaptiert etablierte Verfahren aus dem Fundus der Echtzeitinformatik, um die in sog. Manufacturing Execution Systems ablaufenden produktionsnahen Geschäftsprozesse ressourcen- und kostenschonender in Software umzusetzen, zu testen und zu warten.

Die Sitzung zu Fragen der Skalier- und Konfigurierbarkeit beginnt mit der Vorstellung eines Konzepts, Echtzeitaufgaben physikalisch auf die Knoten konfigurierbarer Mehrprozessorsysteme abzubilden und zur Interprozesskommunikation blockierungsfreie, mehrstufige Netze mit vorhersagbarer Latenz einzusetzen. Es folgt ein Ansatz zur Erhöhung der Flexibilität von Fertigungsanlagen durch Einsatz rekonfigurierbarer Komponenten und echtzeitfähiger Software-Agenten. Am Beispiel der Einspannvorrichtung einer Fräsmaschine wird gezeigt, wie sie durch einen zugeordneten Agenten gesteuert und zur Laufzeit rekonfiguriert wird. Der letzte Vortrag widmet sich einem neuartigen und deutlich leistungsfähigeren Verfahren zur Simulation der Sortierung von Schüttgut und Verwendung eines programmierbaren Framegrabbers. Damit können Testabläufe flexibel gestaltet und unter Echtzeitbedingungen automatisiert abgearbeitet werden.

Zum Abschluss sei zunächst den Autoren gedankt, die ihre Beiträge meistens pünktlich in guter Qualität und in vorgegebener Länge abgeliefert haben. Damit konnte der Tagungsband erneut in einheitlichem Erscheinungsbild mit geringem redaktionellen Aufwand fertiggestellt werden. Dieser Aufgabe und der Korrektur offensichtlicher Fehler hat sich Frau Dipl.-Ing. Jutta Düring wieder mit großer Hingabe gewidmet, wofür ich ihr meinen ganz herzlichen Dank ausspreche. Für die auch in diesem Jahr gewährte finanzielle Unterstützung des Workshops in Boppard sind Programmkomitee und Leitungsgremium des Fachausschusses den langjährigen industriellen Sponsoren zu großem Dank verpflichtet.

Hagen, im August 2011 Wolfgang A. Halang

Inhaltsverzeichnis

Sicherheit

Mehrkernsysteme

Ausbildung

Entwurfsverfahren

Skalier- und Konfigurierbarkeit

Sichere Kommunikation in der Automatisierungstechnik

Linus Schleupner

Lehrstuhl für Informationstechnik, insb. Realzeitsysteme
FernUniversität in Hagen, 58084 Hagen
Linus.Schleupner@fernuni-hagen.de

Zusammenfassung. Automatisierungstechnische Kommunikationsnetze an Maschinen und Anlagen werden vermehrt, z.B. zu Fernwartungszwecken, mit dem Internet verbunden. Auch wird Ethernet als Standard wie ein Feldbus eingesetzt. Die bisher als Insellösung ausgeführten Netze mit proprietären Bussystemen können deshalb genauso von außen angegriffen oder ausgespäht werden wie jedes Büro- oder Heimnetzwerk. Marktgängige Feldbus- oder Funknetze bieten keinen Schutz gegen diese Gefahren. Der Angriff von W.32 Stuxxnet hat gezeigt, dass die Wirkung bis zur Funktionsstörung von Atomanlagen reichen kann. Deshalb müssen Automatisierungsarchitekturen eine im Rahmen der gegebenen Echtzeitbedingungen abhörsichere Kommunikation erlauben, die gegen Einflüsse von außen unempfindlich ist.

1 Einleitung

Die ständige Verfügbarkeit von Infrastrukturen z.B. bei Verarbeitungs- und Produktionsanlagen sowie Anlagen zur Energie- oder Wasserversorgung spielt für Unternehmen, Verwaltungen und private Haushalte eine große Rolle. Dort kommen automatisierte Prozesssteuerungssysteme, IndustriePCs mit Windows-Betriebssystemen und Office-Anwendungen sowie Supervisory Control and Data Acquisition-Systeme (SCADA) zur Steuerung der verschiedenen Funktionen und Abläufe in verteilten Strukturen zum Einsatz. Zur Vernetzung ihrer Komponenten nutzen diese Systeme immer häufiger die gleiche oder ähnliche Ethernet-basierte Netzwerktechnik wie Standard-Computernetzwerke.

Diese in der Automatisierungstechnik zunehmend eingesetzte Ethernet-basierte Feldbustechnik soll wegen der hohen Bandbreite, des hohen Bekanntheitsgrades und der einfachen Anbindung von Netzwerkteilnehmern neue Möglichkeiten öffnen. Die Vorteile liegen anscheinend auf der Hand: Die Anforderung an eine bekannte und einfache Technik, die insbesondere in der Automatisierungs- und Prozesstechnik den Transport und die Verarbeitung immer größerer anfallender Datenmengen erlaubt, ist zunächst erfüllt. So können bisher gängige, in der Maschinenautomatisierung lokal eingesetzte, proprietäre Feldbussysteme abgelöst werden.

Das führt allerdings dazu, dass Automatisierungssysteme potenziell den gleichen Gefahren durch Viren, Würmer, Trojaner und unbedachte Nutzer ausge-

W.A. Halang (ed.), *Herausforderungen durch Echtzeitbetrieb*, Informatik aktuell,
DOI 10.1007/978-3-642-24658-6_1,

setzt sind wie jeder Büro- oder Heim-PC. Konzepte zur datentechnischen Einbindung aller Komponenten einer automatisierten Anlage über Ethernet-Netzwerke werden bereits diskutiert. Geschäftsmodelle zur Auslagerung von Service- und Instandhaltungsaufgaben auf externe, ggf. nicht am Standort der Anlage ansässige Unternehmen, verstärken die potenziellen Risiken zusätzlich, ebenso wie die Vernetzung verschiedener Produktionsstandorte über Enterprise Ressource Planning - Systeme (ERP-Systeme). Etablierte Schutzmaßnahmen aus der Büro-Informationstechnik lassen sich aber nicht 1:1 in die Automatisierungstechnik übertragen [1, 2]. Viele Prozesse, z.B. in Kraftwerken oder Stahlwerken, können nicht einfach angehalten werden, um notwendige Updates von Betriebssystemen oder Virenschutzprogrammen mit anschließendem Systemneustart durchzuführen.

Die Sicherheit gegen Angriffe von außen wie Sabotage oder Manipulation ist in automatisierten Anlagen in jeder Hinsicht elementarer Bestandteil zur Sicherstellung von Verfügbarkeit, Zuverlässigkeit und Authentizität. Eine Unterbrechung der Produktion aufgrund sabotierter oder manipulierter Anlagen kann schwerwiegende Folgen nach sich ziehen. Vertragsstrafen können bei falsch produzierter Menge oder verzögerter Lieferung greifen oder Rückrufaktionen können bei mangelhafter Qualität die Folge sein. Auch können Anlagen beschädigt oder unbrauchbar werden, was mit Imageschäden oder hohem Geldverlust einhergeht. Angriffe auf Kraftwerke können zudem notwendige Energie- oder Stromversorgungen ausschalten.

Die erstmals im Juni 2010 bekannt gewordene Attacke der Schadsoftware W32.Stuxnet zeigt, dass die bisher autark und in sich geschlossen betriebenen Automatisierungsnetze durch gezielte Angriffe von außen verwundbar sind. Grund dafür ist die oben beschriebene Anbindung der Automatisierungsnetze an das Internet, z.B. zu Fernwartungszwecken und die damit verbundene Möglichkeit, Schadsoftware einzuschleusen. Das Ziel von W32.Stuxnet war und ist das Ausspionieren und die Umprogrammierung vorhandener Software speziell in den Steuerungssystemen der Automatisierungstechnik zur Sabotage von Kraftwerken, chemischen Fabriken und industrieellen Produktionsanlagen. Über eine vorhandene Internetverbindung wird zuerst die PC- und dann gezielt die SPS-Ebene (Speicherprogrammierbare Steuerung) in einer Automatisierungsarchitektur infiziert. Als Schaden wurde bisher veröffentlicht, dass in iranischen Atomanlagen Uranzentrifugen manipuliert und beschädigt wurden [3, 4]. In [5] ist die Angriffsstrategie von „W32.Stuxnet“ detailliert beschrieben.

An automatisierungstechnische Netze können neue Teilnehmer ohne oder nur mit sehr wenigen Sicherheitsprüfungen geschaltet werden. Das bezieht sich nicht nur auf die in Automatisierungsarchitekturen üblichen Feldgeräte wie Umrichter, Steuerungen, Sensoren oder Karten mit Ein- und Ausgängen, sondern auch auf Programmiergeräte. Letztere sind üblicherweise als tragbare Rechner ausgeführt und dienen zur Parametrierung und Konfigurierung der Teilnehmer sowie zur Erstellung von Ablaufprogrammen speicherprogrammierbarer Steuerungen. Weiterhin können mit Programmiergeräten Diagnosedaten und Programme ausgelesen und beeinflusst werden. Auf diese Weise ist es für Wirtschaftsspione

oder Saboteure sehr einfach, vertrauliche oder sicherheitsrelevante Prozess- oder Programmdaten auszulesen und zu verändern.

Vertraulichkeit kann mit aufwändigen physikalischen Mitteln gewährleistet werden. Es ist jedoch günstiger und effektiver, dafür geeignete kryptographische Methoden zu verwenden.

Nach dem für die Informationstheorie grundlegenden Satz von Shannon in [6] gilt ein Verschlüsselungssystem dann als perfekt sicher, wenn die Anzahl der möglichen Schlüssel mindestens so groß ist wie die Anzahl der möglichen Nachrichten. Damit ist die Anzahl der Schlüssel ebenfalls mindestens so groß wie die Anzahl der möglichen Chiffrate, die ihrerseits mindestens so groß wie die Anzahl der möglichen Klartexte sein muss.

Demnach ist von den heute bekannten Verschlüsselungsmethoden als Einzige die Einmalverschlüsselung wegen der Einmaligkeit der Schlüsselverwendung als perfekt sicher bewiesen [7, S. 11, S. 40ff.].

Als Grundlage der Schlüsselerzeugung werden Zufallszahlen verwendet, und zwar in der Regel Pseudozufallszahlen, weil sie schnell und einfach generiert werden können. Diese sind jedoch deterministisch und müssen für die kryptographische Verwendung mit großem Aufwand statistisch nachbereitet werden. Im Unterschied zu Pseudozufallszahlen besitzen echte Zufallszahlen die Eigenschaften der Unvorhersagbarkeit, der Gleichverteilung in einer Zahlenfolge und der Unabhängigkeit von Anfangswerten oder Randbedingungen. Echte Zufallszahlen sind in Aufgaben der Kryptographie den Pseudozufallszahlen vorzuziehen.

Die Zykluszeiten der Datenkommunikation in der Automatisierungstechnik liegen aktuell bei unter 1 ms, wodurch sich hohe Echtzeitanforderungen ergeben. Echtzeitfähigkeit bedeutet hier, dass Prozessdaten oder -befehle zu vordefinierten Zeitpunkten von einem Empfänger verarbeitet sein müssen. Viele Anwendungsfälle unterliegen dabei der sogenannten harten Echtzeitbedingung. Bei solchen Systemen muss der zeitliche Rahmen eingehalten werden, da sonst Prozesse nicht mehr zufriedenstellend abgeschlossen werden können, sei es, dass ein Schweißbalken Kunststoffnähte bei Verpackungen nicht mit der richtigen Festigkeit zusammenschweißt oder dass Nahrungsmittelrezepturen zu geschmacksverfälschten Produkten führen.

Die aus der quantenphysikalischen Übertragung von Bits her bekannten Protokolle BB84 und E91 erlauben zwar die sichere quantenphysikalische Übertragung von Bits. Allerdings ist die Anzahl der sicher übertragbaren Bits bei Weitem nicht ausreichend, um allein diese nach den Anforderungen der Automatisierungstechnik zur Schlüsselerzeugung zu verwenden. Heute in [8, S. 79] untersuchte Systeme erreichen eine Datenrate von 95 kBd, womit nur etwa 8 Ethernet-Pakete mit je 12 kBit Nutzlast pro Sekunde verschlüsselt werden können. Es würden jedoch nach [9] bereits in einem kleinen Netzwerk ca. 3000 Pakete pro Sekunde benötigt. Diese Methode allein genügt also nicht den Anforderungen der Automatisierungstechnik.

Nach diesen Überlegungen sollte die Verschlüsselung der Bitströme mittels Einmalverschlüsselung sowie die Verteilung der erzeugten echten Zufallszahlen mittels quantenphysikalischem Protokoll BB84 bzw. besser E91 erfolgen. Gegen

beide Verfahren ist unter der Voraussetzung der Null-Fehler-Toleranz bisher kein erfolgreicher Angriff bekannt. Lediglich die Anzahl der nach den Ansprüchen der Automatisierungstechnik zu verteilenden Zufallsbits ist bei weitem zu gering. Gelöst werden muss also die ausreichende Erzeugung echter Zufallsbits und die permanente Generierung und zeitgerechte Bereitstellung von Einmalschlüsseln.

2 Ergänzung der Automatisierungsarchitektur

Zunächst muss eine klassische Automatisierungsarchitektur mit Hardware ergänzt werden.

Zur Wahrnehmung von übergeordneten, zentralen Netzwerkaufgaben wird ein *Leitknoten* eingesetzt, welcher als Zusatzkarte zur SPS gesteckt oder separat als eigener Hardwarebaustein montiert werden kann. Einige Aufgaben dieses Bausteins sind die Koordination der sicheren internen Kommunikation mit den Kryptomodulen, Klärung des externen Zugriffs auf das Netzwerk (lokal oder über Fernzugriff), die Erzeugung und der quantentechnische Versand neuer, echter Zufallsbits innerhalb des Automatisierungssystems für die Schlüsselerzeugung und die zeitliche Synchronisierung aller Teilnehmer.

Weiter wird ein *Kryptomodul* eingeführt, das als Netzknoten jeweils zwischen ein automatisierungstechnisches Gerät, allgemein einen Knoten, und den Feldbus geschaltet wird und eine Schnittstelle zu diesem Feldbus (110), eine Ver-/Entschleierungseinheit (114), ein Ver-/Entschlüsselungsgerät (106), einen Schlüsselgenerator (108), einen Speicherbereich aus einem oder mehreren Speichern (101, 102, 103, 104) mit wahlfreiem Zugriff, einen als EEPROM ausgeführten Algorithmenspeicher (105), eine Schnittstelle zum Feldgerät (109), einen Generator echter Zufallsbits (107), eine Quantenschlüsselverteilungseinheit QKD (112) mit Schnittstellen für Lichtwellenleiter (113) sowie einen Mikroprozessor mit Systemuhr (111) enthält (vgl. Abbildung 1 und 2).

Die notwendigen echten Zufallszahlen bzw. -bits werden von einer Verschaltung der Ausgangssignale mehrerer Signalgeneratoren erzeugt, von denen einer ein chaosbasierter Chua-Schaltkreis ist. Ein solcher Schaltkreis erzeugt einen sogenannten „Seltsamen Attraktor“, welcher als Signal genutzt werden kann. Die entsprechende Signalüberlagerung führt zu statistischem Rauschen, aus dem entweder direkt Zufallsbits entnommen werden können, oder das zur Erhöhung der Konfusion und Diffusion als Abtastsignal einer natürlichen Rauschquelle verwendet wird.

Die Kryptomodule übernehmen als Netzknoten die gesamte Kommunikation im Netz und sind untereinander durch zwei Leitungen verbunden: Zum einen durch den Feldbus (205) zur Kommunikation der Prozessdaten und zum anderen durch den Lichtwellenleiter (206), über den die zeitliche Synchronisierung erfolgt und kryptographische Daten und die Zufallsbits quantenphysikalisch übermittelt werden.

Da mit Laserdioden arbeitende Lichtwellenleitersysteme in der Lage sind, durch Modenmodifikation unterschiedliche Kanäle gleichzeitig im Lichtwellenleiter zu nutzen, können sowohl quantenphysikalisch als auch nicht-quantenphysi-

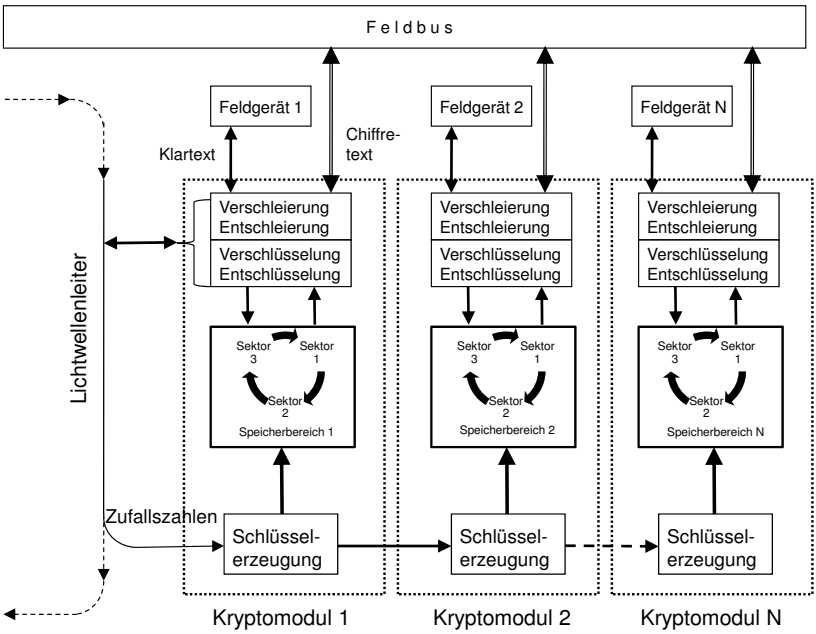


Abb. 1. Ablauf des Verfahrens

kalisch erzeugte Daten über einen einzigen Lichtwellenleiter auf unterschiedlichen Kanälen gesendet werden.

3 Inbetriebnahmemodus; Initialisierung des Automatisierungssystems und Schlüsselgenerierung

Neu eingeschaltete Kryptomodule (202, 203, 204) befinden sich im *Inbetriebnahmemodus* und die Speicher (101, 102, 103, 104) aller dieser Knoten sind dann leer (vgl. Abbildung 2). Lediglich das Betriebssystem und die Programme in den Algorithmenspeichern sind fest und auslesesicher implementiert. Der Verbindungsaufbau für den Versand der Zufallsbits vom Leitknoten zu den Kryptomodulen erfolgt als 3-Wege-Verfahren. Dabei senden bei der Erstinbetriebnahme die Kryptomodule aller Teilnehmer bzw. das Kryptomodul eines neu aufzunehmenden Teilnehmers im laufenden Betrieb der Anlage eine Anfrage an den Leitknoten und fordern dort durch einen Protokollbefehl echte Zufallsbits sowie die Systemzeit an. Der Leitknoten antwortet und meldet die Bereitschaft zur Aufnahme. Die Kryptomodule quittieren den Empfang der Aufnahmebereitschaft.

Der Leitknoten startet daraufhin die Synchronisierung der Systemzeit, erzeugt die Zufallsbits Z_k und verteilt sie quantenphysikalisch über den Lichtwellenleiter (206) an die jeweiligen Kryptomodule (202, 203, 204). Der Index k

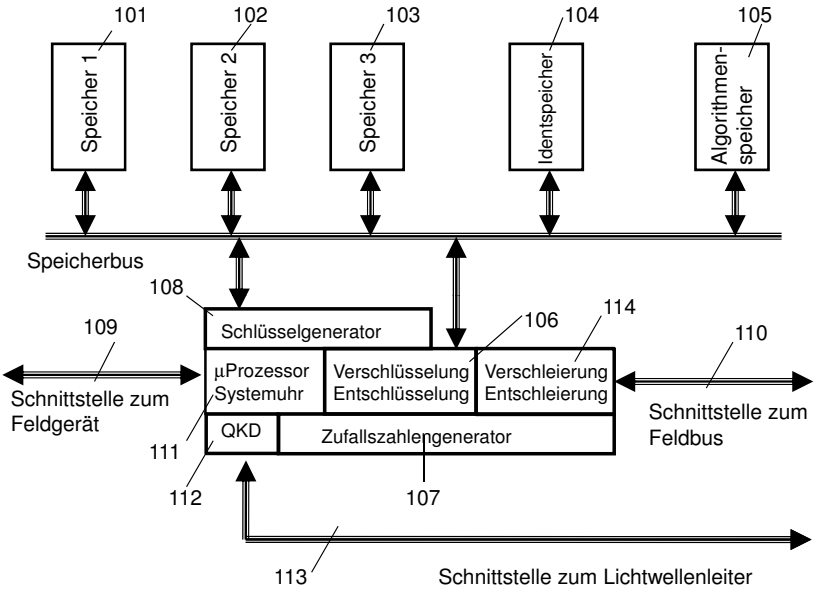


Abb. 2. Aufbau eines Kryptomoduls

bezeichnet dabei die Reihenfolge, in der die Zufallsbits von den empfangenden Kryptomodulen verarbeitet werden müssen.

Weiterhin teilt der Leitknoten allen Kryptomodulen die Gesamtzahl N der im Netz vorhandenen Kryptomodule mit und weist jedem Kryptomodul eine Ordnungsnummer N_m aufsteigend von $m = 1$ bis N zu. Das Kryptomodul des Leitknotens erhält immer die Ordnungsnummer N_0. In jedem Kryptomodul (202, 203, 204) werden dann die vorhandenen und ausreichend dimensionierten Speicher (101, 102, 103) in so viele Speicherbereiche S_N eingeteilt, wie Kryptomodule im Netz vorhanden sind. Diese Speicherbereiche werden nochmals in je drei gleichgroße Sektoren unterteilt.

Mit den übertragenen Zufallsbits Z_k erzeugen dann die Kryptomodule Kommunikationsschlüssel K_{k_j} nach einem durch den Leitknoten zufällig ausgewählten, für alle Kryptomodule identischen und im Algorithmenspeicher (105) vorgehaltenen Algorithmus j und schreiben diese in die vorbereiteten sektorisierten Speicherbereiche der Speicher (101, 102, 103) und in den Identspeicher (104), so dass anschließend die einander entsprechenden Speicherbereiche sowie die Identspeicher aller Kryptomodule identischen Inhalt aufweisen.

Die vorgenommene Organisation der Speicher führt zwar dazu, dass alle Kryptomodule identische Informationen über die vorhandenen Schlüssel besitzen. Jedes Kryptomodul verwendet jedoch individuell für die Verschlüsselung der eigenen Daten ausschließlich und eineindeutig den Speicherbereich mit genau den drei Sektoren, der seiner Ordnungsnummer entspricht. Für jedes Kryptomodul N_m ist das der Speicherbereich S_m. Die anderen Bereiche dienen ausschließlich

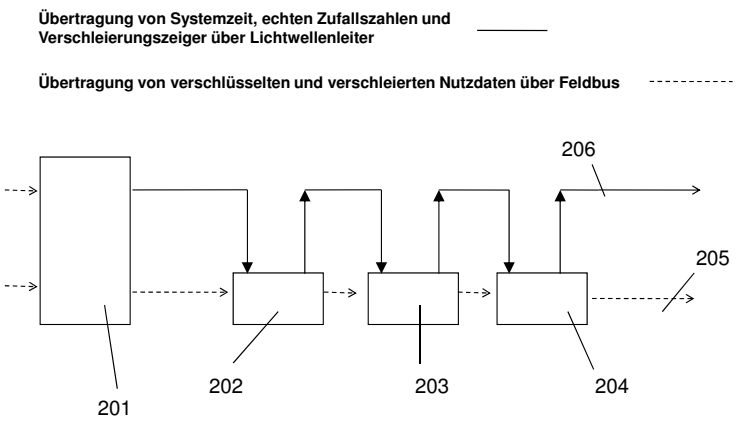


Abb. 3. Anbindung der Kryptomodule durch Lichtwellenleiter

der Entschlüsselung der von den jeweils anderen Kryptomodulen verschickten Daten (vgl. Abbildung 4).

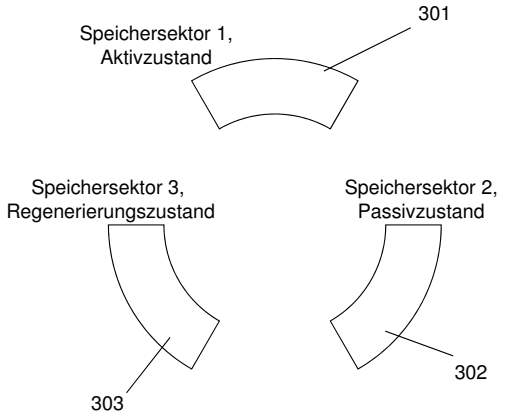


Abb. 4. Initialisierungsphase

Jeder individuell genutzte Speicherbereich besteht aus genau drei Sektoren, beziffert mit 1, 2 und 3 (vgl. Abbildung 4). Jeder dieser Sektoren (301, 302, 303) kann sich in genau einem Betriebszustand befinden. Im Aktivzustand (A) befindet sich ein mit Schlüsseln gefüllter und aktiv arbeitender Sektor, ein weiterer im Passivzustand (P) befindlicher Sektor ist ebenfalls mit Schlüsseln gefüllt und so vorbereitet, dass bei Bedarf umgeschaltet und sofort Schlüssel entnommen werden können. Durch den laufenden Betrieb werden aus dem jeweils im Aktivzustand befindlichen Sektor Schlüssel entnommen. In diesem Sektor ste-