# Eine Alternative zum amerikanischen DES-Chiffrier-Code : neuer VLSI-Chip für neuen Blockchriffrieralgorithmus

Autor(en): **Curiger, Andreas**

Objekttyp: **Article**

Zeitschrift: **Bulletin des Schweizerischen Elektrotechnischen Vereins, des Verbandes Schweizerischer Elektrizitätsunternehmen = Bulletin de l'Association Suisse des Electriciens, de l'Association des Entreprises électriques suisses**

Band (Jahr): **83 (1992)**

Heft 9

PDF erstellt am: **10.07.2024**

Persistenter Link: https://doi.org/10.5169/seals-902821

# Eine Alternative zum amerikanischen DES-Chiffrier-Code

## Neuer VLSI-Chip für neuen Blockchiffrieralgorithmus

Andreas Curiger

***In einem Gemeinschaftsprojekt von ETH Zürich und Ascom Tech AG wurde ein neuer Blockverschlüsselungsalgorithmus (Idea) entwickelt, der auf dem Prinzip der gleichzeitigen Verwendung von mehreren unvereinbaren algebraischen Gruppenoperationen basiert. Mit dem Algorithmus wird ein Verfahren zur Verfügung gestellt, das sowohl hinsichtlich Sicherheit als auch Datendurchsatz den heutigen DES-Implementierungen ebenbürtig, wenn nicht sogar überlegen ist. Durch freie Zugänglichkeit soll erreicht werden, dass Idea zu einem De-facto-Standard wird.***

***Un nouvel algorithme de chiffrage de bloc (Idea) a été développé dans le cadre d'un projet commun entre l'EPF de Zurich et Ascom Tech SA. L'originalité de cet algorithme découle du principe de l'utilisation concomitante de plusieures opérations algébriques incompatibles entre elles-mêmes. Comme résultat nous proposons un procédé de chiffrage équivalent, sinon supérieur, aux implantations actuelles du DES du point de vue de la sécurité ainsi que du débit.***

**Adresse des Autors**

*Andreas Curiger*, Dipl. El.-Ing. ETH, Institut für Integrierte Systeme, Gloriastrasse 35, ETH-Zentrum, 8092 Zürich.

Die Kryptographie befasst sich mit der Umwandlung oder Verschlüsselung von Information (Klartext) in Code (Chiffrat), der nur für autorisierte Empfänger verständlich sein soll, um Geheimhaltung, Echtheit und Unverfälschtheit dieser Information sicherzustellen. Während diese Methoden früher fast ausschliesslich auf diplomatischem und militärischem Gebiet Verwendung fanden, sind sie heute überall von Bedeutung, wo vertrauliche Informationen bei der Übertragung, Speicherung und Verarbeitung geschützt werden müssen. Die Methoden der Kryptographie versuchen, Chiffrierungen zu implementieren, die genügend komplex sind, um möglichen Angriffen (Kryptoanalyse) erfolgreich zu widerstehen. Solche Attacken können durch statistische Analyse des verschlüsselten Textes oder durch Brute Force-Entschlüsselungsmethoden, wie systematisches Ausprobieren aller möglichen Schlüssel, erfolgen.

Eine wichtige Eigenschaft der modernen Kryptographie ist die Verwendung von öffentlich bekannten, das heisst publizierten Algorithmen. Die Geheimhaltung liegt somit nicht im Verfahren der Verschlüsselung, sondern im jeweils verwendeten Schlüssel. Grundsätzlich werden zwei Arten der Verschlüsselung unterschieden, nämlich die sogenannte symmetrische Chiffrierung mittels eines vollständig geheimen Schlüssels und die asymmetrische Chiffrierung mittels eines teilweise öffentlich bekannten Schlüssels. Für weiterführende Informationen sei auf [1] verwiesen. Die Chiffrierungen mit geheimen Schlüsseln haben den Vorteil, dass deren Implementierungen grundsätzlich mit bedeutend höheren Datenraten als die mit öffentlichen Schlüsseln zu arbeiten vermögen [2].

## Verfahren der Blockverschlüsselung

Eine Blockverschlüsselung ist dadurch charakterisiert, dass die zu verschlüsselnde Information, der Klartext, in eine Folge von Blöcken fester Länge aufgeteilt und jeder Block nach der gleichen Vorschrift verschlüsselt wird. Ist die Länge des Klartextes kein Vielfaches der Blocklänge, so muss der Klartext nach einem vereinbarten Schema auf ein Vielfaches der Blocklänge aufgefüllt werden. Typisch sind heute Blocklängen von 64 oder 128 Bit für Klartext- und Chiffratblöcke.

Gleiche Klartextblöcke $m$ werden bei gleichbleibendem Schlüssel somit zu gleichen Chiffratblöcken $c$ transformiert. Diese Schwäche wurde überwunden, indem mehrere Betriebsarten (Modi) der Blockverschlüsselung definiert wurden [3]. Mit einer solchen standardisierten Verkettung des chiffrierten Texts, dem Cipher Block Chaining (CBC), kann erreicht werden, dass identische Klartextblöcke in der Regel auf verschiedene Chiffratblöcke abgebildet werden.

## Der DES-Algorithmus

Der bekannteste und am weitesten verbreitete Algorithmus, welcher eine Blockchiffrierung mittels geheimem Schlüssel verwendet, ist der Data Encryption Standard (DES). Er ist sowohl durch das American National Bureau of Standards (NBS, 1977) für US-Bundesbehörden als auch durch das American National Standards Institute (Ansi-Norm X3.92, 1982) für den kommerziellen Bereich adaptiert worden. Die Fabrikation von DES-Chips erfolgte bis vor wenigen Jahren ausnahmslos in Nordamerika, und der Export wurde vom U.S. State Department generell nur für finanzielle

Transaktionssysteme erlaubt. Patentgründe haben Hersteller ausserhalb den USA bis vor kurzem abgeschreckt, einen eigenen DES-VLSI-Chip zu entwickeln. Dazu kommen noch die folgenden technischen Probleme:

- Der verwendete Schlüssel umfasst eine Wortlänge von nur 56 Bit. Es gibt somit $2^{56} = 7{,}2 \cdot 10^{16}$ verschiedene Schlüssel. Obwohl bei einer Entschlüsselungsdauer von 2 $\mu$s pro Datenblock, was aktuelle Chips zu leisten imstande sind [4], ein systematisches Ausprobieren aller möglichen Schlüssel mehr als 4500 Jahre in Anspruch nehmen würde, so kann doch durch parallele Verwendung von einigen Tausend solcher Verschlüsselungseinheiten der benötigte Zeitraum auf einige Monate gesenkt werden. Rechnet man zudem mit weiteren Fortschritten im Bereich der Verarbeitungsgeschwindigkeit der Hardware, gibt es erhebliche Bedenken, ob der DES-Algorithmus mittelfristig noch genügend sicher ist.
- Es existieren sogenannte schwache Schlüssel (wenn auch wahrscheinlich nur sehr wenige), welche eine Kryptoanalyse erleichtern.
- Die Prinzipien der DES-Entwicklung bleiben von der National Security Agency (NSA) klassifizierte Information, so dass nichts über Auswahlkriterien, welche zur konkreten Ausgestaltung des DES geführt haben, bekannt ist.

Trotzdem hat sich der DES-Algorithmus als De-facto-Weltstandard der Blockverschlüsselung etablieren können. Der DES ist bis heute der einzige standardisierte Verschlüsselungsalgorithmus. Die Nachfolgealgorithmen der National Security Agency (NSA) sind nicht öffentlich und als Chiprealisierung nur teilweise in den USA erhältlich. Diese besonders für Europa unbefriedigende Situation macht den Wunsch verständlich, ein öffentlich verfügbares Blockverschlüsselungsverfahren zu entwickeln, welches höheren Sicherheitsanforderungen als der DES-Algorithmus genügt.

## Der Internationale Datenverschlüsselungsstandard Idea

In einem Gemeinschaftsprojekt der Ascom Tech und des Instituts für Signal- und Informationsverarbeitung der ETH Zürich (ISI) wurde von Xuejia Lai und J.L. Massey ein neuer Blockverschlüsselungsalgorithmus entwickelt [5;6]. Die Anforderungen an den Algorithmus waren:

- hohe kryptographische Sicherheit,
- Möglichkeit einer effizienten Hardware-Implementierung,
- Möglichkeit einer einfachen Software-Implementierung auf den heute erhältlichen Mikroprozessoren,
- problemlose Chip-Herstellung in Europa.

Um die Entwurfsprinzipien für diesen neuen Algorithmus zu beschreiben, sei an die von Shannon in [7] definierten zwei Anforderungen an gute praktische Blockchiffren erinnert:

*Konfusion:* Die Statistik des Chiffrats sollte von der Statistik des Klartexts möglichst kompliziert abhängen, so dass ein Kryptoanalyst nicht mit Hilfe der Chiffrat-Statistik auf den Klartext schliessen kann.
*Diffusion:* Jedes Bit des Schlüssels und jedes Bit des Klartexts sollen möglichst viele Bits des Chiffrats beeinflussen, um Divide and Conquer-Attacken abzuwehren.

Im DES-Algorithmus wurden diese beiden Prinzipien folgendermassen realisiert: Konfusion wird durch den Gebrauch nichtlinearer Funktionen innerhalb einer Runde, den f-Functions, erhalten. Die Implementierung basiert auf Look-up-Tabellen, welche S-Boxes genannt werden. Diffusion wird durch Transposition von Worten zwischen den Runden durch die T-Operation erreicht. Im neuen Algorithmus, der im folgenden Idea (Inter-

### Arithmetik in algebraischen Strukturen

Eine Vielzahl wichtiger kryptologischer Systeme und Codes beruht auf algebraischen Strukturen wie polynomialen Ringen und Galoiskörpern. Die Algebra klassifiziert die vielen arithmetischen Systeme, die sie behandelt, gemäss ihrer mathematischen Stärke:

1. Die mathematischen Objekte einer Abelschen Gruppe können «addiert» und «subtrahiert» werden.
2. Die mathematischen Objekte eines Rings können «addiert, «subtrahiert» und «multipliziert» werden.
3. Die mathematischen Objekte eines Körpers können «addiert», «subtrahiert», «multipliziert» und «dividiert» werden.

Die Namen der Operationen sind zwischen Anführungs- und Schlusszeichen gesetzt, da die Operationen im allgemeinen nicht mit den uns vertrauten arithmetischen Operationen identisch sind, diesen jedoch gleichen.

#### Zum Beispiel Körper

Das einfachste Beispiel eines Körpers besteht aus zwei Elementen, z.B. 0 und 1. Addition und Multiplikation seien folgendermassen definiert:

| | |
|---|---|
| $0+0=0$ | $0\cdot 0=0$ |
| $0+1=1$ | $0\cdot 1=0$ |
| $1+0=1$ | $1\cdot 0=0$ |
| $1+1=0$ | $1\cdot 1=1$ |

Die so definierten Operationen werden modulo-2-Addition (oder XOR) und modulo-2-Multiplikation (oder AND) genannt.
Körper mit einer nichtunendlichen Anzahl Elemente werden endlich genannt. Ein Körper mit $q$ Elementen heisst auch Galoiskörper (Engl. Galois Field, nach Evariste Galois, 1811–1832) und wird mit GF($q$) bezeichnet. Damit die Körpereigenschaften erfüllt sind, muss $q$ eine Primzahl ($q = p$) oder eine Potenz einer Primzahl ($q = p^m$) sein. Das Alphabet der zwei Symbole 0 und 1 formt zusammen mit der modulo-2-Addition und der modulo-2-Multiplikation einen endlichen Körper mit zwei Elementen, der mit GF(2) bezeichnet wird. Der Körper GF(216 + 1) wird durch die Elemente {0, 1, 2,..., 65536} und die Operationen modulo-65537-Addition und modulo-65537-Multiplikation gebildet.

#### Modulo-Arithmetik

Das Resultat der Operation $a$ mod $p$ entspricht dem Rest $r$ der Division $a/b$, während mit $a$ div $p$ der Quotient $q$ der Division ermittelt wird. Somit gilt:

$$\begin{aligned} a/p &= qp + r \\ q &= a \operatorname{div} p \\ r &= a \bmod p \end{aligned}$$

Beispiel: 142 mod 17 = 6; 142 div 17 = 8; $142 = 8 \cdot 17 + 6$.

national Data Encryption Algorithm) genannt wird, ist Konfusion durch die Kombination von Operationen dreier verschiedener algebraischer Strukturen realisiert [8]. Bei den drei Operationen handelt es sich um

- Addition modulo $2^{16}$ in der Abelschen Gruppe der ganzen positiven Zahlen $[0,\ldots,2^{16}-1]$,
- Multiplikation modulo $2^{16}+1$ in der multiplikativen Gruppe des endlichen Körpers $GF(2^{16}+1)$ und
- Bit-für-Bit-Addition modulo 2 von 16-Bit-Teilblöcken, was einer Bit-für-Bit Exklusiv-Oder-Verknüpfung entspricht.

Die Konfusion wird durch eine geeignete Abfolge der Operationen erreicht, so dass im Verschlüsselungsfluss die gleichen Gruppenoperationen nie direkt aufeinanderfolgen. Diffusion entsteht durch die Anordnung dieser drei Gruppenoperationen, woraus sich die wünschbaren statistischen Eigenschaften zwischen dem zu verschlüsselnden Klartext, dem Schlüssel und dem resultierendem Chiffrat ergeben. Das resultierende Gesamtsystem ist in hohem Masse nichtlinear. Das Zusammenwirken der verschiedenen algebraischen Gruppenoperationen ersetzt die bisher bei Blockchiffren üblichen Wertetabellen zur Substitution von Bitblöcken. Dadurch werden Tabellen-Lookups überflüssig.

Die graphische Darstellung des Verschlüsselungsprozesses ist in Bild 1 gezeigt. Die Struktur weist acht Teilstrukturen, sogenannte Runden, auf und enthält in ihren verschiedenen 16-Bit-Datenpfaden mehrfach die drei erwähnten Operationen. Die acht Runden und eine abschliessende Ausgabetransformation werden kaskadiert und bilden dadurch die Chiffre. Die Funktion $f(x)$, welche die Chiffre beschreibt, hat involutorische (umkehrbare) Eigenschaft, da folgender Zusammenhang gilt:

$$f[f(x)] = x \qquad (1)$$

Die Ausgabetransformation ist nötig, um die involutorische Eigenschaft zu gewährleisten.

Der 64 Bit lange Klartextblock wird in vier 16-Bit-Teilblöcke zerlegt, da alle für den Chiffrierprozess eingesetzten algebraischen Operationen 16-Bit-Operanden verwenden. Aus dem 128 Bit langen Schlüssel werden mit Hilfe einer festgelegten Prozedur [6] für jede Verschlüsselungsrunde sechs

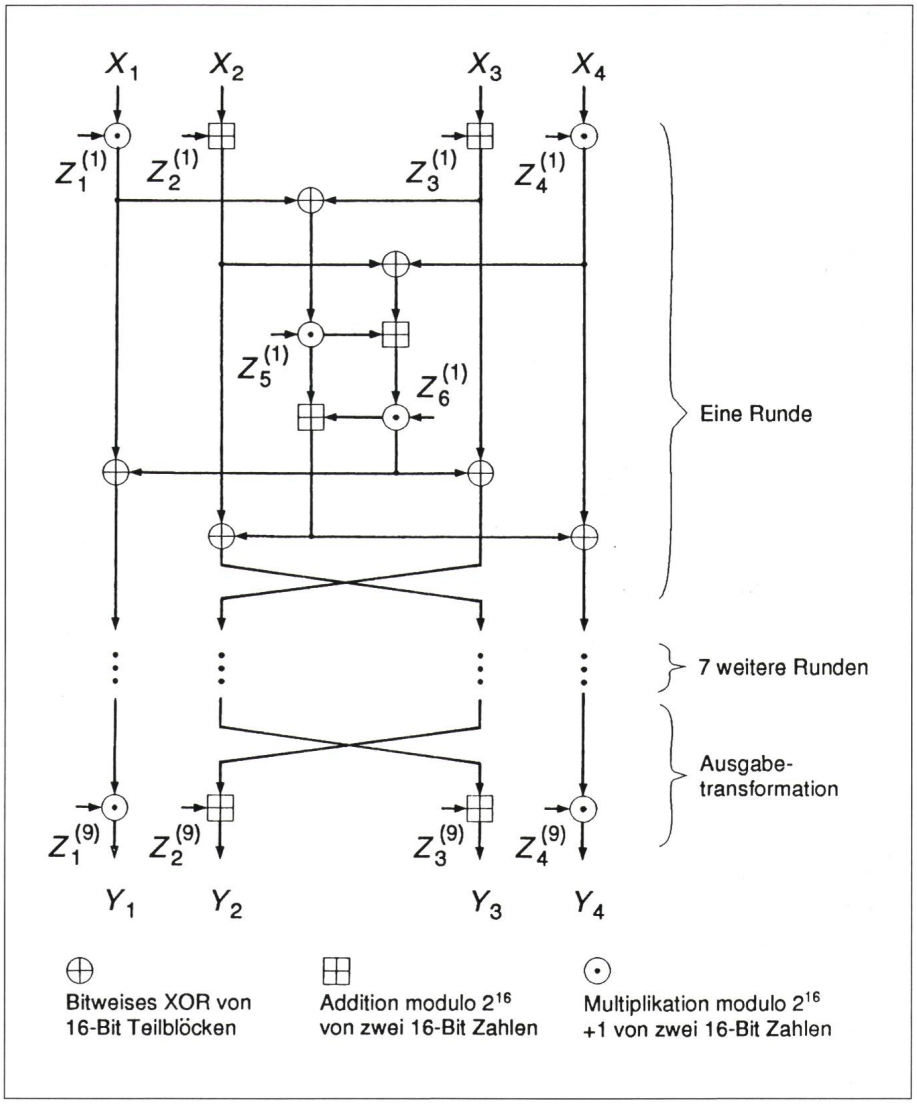



**Bild 1 Datenfluss des Idea-Algorithmus**
$X_1$ bis $X_4$ 16-Bit-Klartext-Teilblöcke
$Z_1^{(1)}$ bis $Z_4^{(9)}$ 16-Bit-Schlüssel-Teilblöcke

16-Bit-Schlüsselteilblöcke gebildet. Da für die abschliessende Ausgabetransformation nochmals vier Schlüsselteilblöcke benötigt werden, müssen aus dem Schlüssel für Ver- und Entschlüsselung je 52 verschiedene 16-Bit-Schlüsselteilblöcke generiert werden. Der Idea-Algorithmus ist heute fertig entwickelt und getestet. Als Software-Implementierung kann der Algorithmus ohne besondere Codeoptimierung auf einer Sun Sparc-Station 2 mit etwa 400 kBit/s verschlüsseln.

## Ein VLSI-Chip für Idea

In einer frühen Projektphase haben Ascom Tech, ISI und das Institut für Integrierte Systeme der ETH Zürich (IIS) vereinbart, einen VLSI-Chip zu entwickeln, welcher die neue Blockchiffriermethode implementiert. Diese Vereinbarung kam in einer frühen Projektphase zustande, als viele Optionen, welche die Chiffrierung betrafen, noch nicht definitiv festgelegt waren. Der Algorithmus konnte deshalb derart angepasst werden, dass nun ein optimales Leistungs-Kosten-Verhältnis für eine Hardware-Realisierung erreichbar ist.

Die Anforderungen an eine hohe Datendurchsatzrate liessen nur eine Chiplösung in Frage kommen. Eine VLSI-Implementierung zeigt zudem weitere erwünschte Eigenschaften wie geringer Platzbedarf oder erschwerte Nachahmungs-Bedingungen [9]. Kryptographische Applikationen erfordern gleichzeitig hohe Verarbeitungsgeschwindigkeit und hohe Si-

cherheit. Ein VLSI-Chip, welcher den Algorithmus implementieren soll, muss somit eine hohe Datenrate gewährleisten, ohne dass dabei der Algorithmus oder die Sicherheit beeinträchtigt werden.

Der Prozess, von den ersten Architektur-Überlegungen bis zum serienreifen Produkt, erfolgte in zwei Etappen:

- Am IIS wurde ein erster Chip (Leonardo) entwickelt [10], welcher sowohl als Funktionsmuster für die arithmetischen Einheiten als auch als Hardware-Accelerator zur Untersuchung offener kryptographischer Fragen diente. Es wurde Wert auf vollständige Funktionalität und einen hohen Datendurchsatz gelegt. Sicherheitsanforderungen bezüglich Teststrukturen und Selbsttest wurden ausgeklammert. Auch die Generierung der Schlüsselteilblöcke und deren Verwaltung fanden für den Prototypen ausserhalb des Chips statt.
- Das Endprodukt, der Idea Chip, basiert auf den Erfahrungen und Resultaten des ersten Chips. Zusätzlich implementiert er jedoch die gesamte Schlüsselgenerierung und deren Verwaltung sowie vier weitere standardisierte Modi der Blockverschlüsselung [3]. Als serienreifes Produkt erfüllt er Anforderungen an die Hardware-Sicherheit gemäss ANSI-Standard X9.17 [11]. Gegenüber dem ersten Chip wird ein vierfach höherer Datendurchsatz erwartet.

## Prototyp-Entwurf

Für eine VLSI-Chip-Implementierung mussten zwei kontroverse Aufgaben gelöst werden:

1. Um kryptographische Sicherheit zu gewährleisten, muss die Struktur einer Chiffre so irregulär wie möglich sein.
2. Eine Struktur, die auf einem Chip integriert wird, sollte so regulär wie möglich sein.

Des weiteren gilt, dass jeder Hardwareblock, der während der Dauer einer Berechnung nicht verwendet wird und in einem Wartezustand verharrt, vom Gesichtspunkt der Chip-Implementierung sehr ineffizient ist. Es ist daher sehr wichtig, denselben Datenpfad sowohl für Verschlüsselung als auch für Entschlüsselung zu verwenden. Dank der involutorischen Eigenschaft der neuen Chiffre konnte diese Forderung einfach realisiert werden. Die Anforderung nach hohem Datendurchsatz stellte im wesentlichen zwei Probleme:

1. Entwurf eines schnellen Datenpfads,
2. Entwurf einer Schnittstelle (Interface), welche fähig ist, den resultierenden Datenfluss zur Aussenwelt zu bewältigen.

Diese Probleme wurden gelöst, indem ein Pipelining-Schema kombiniert mit schnellen Hardware-Rechenheiten entworfen und ein schnelles Standard-Interface implementiert wurde.

### Entwurf des Datenpfads

Das Hauptproblem des Algorithmus hinsichtlich einer VLSI-Implementierung bildet die Multiplikation modulo $2^{16}+1$ zwischen einem 16-Bit-Daten- und einem 16-Bit-Schlüsselwort (Bild 1). Die Dauer dieser Rechenoperation und die Fläche, die ein solches Multiplizierwerk auf dem Chip beansprucht, haben einen bedeutenden Einfluss auf die Architektur und somit auch auf die erreichbare Datenrate. Verschiedene Techniken zur Implementierung eines solchen speziellen Multiplizierers wurden daher zuerst untersucht [12;13]. Zusammengefasst ergaben sich folgende Varianten:

- Multiplikation mittels direktem Tabellen-Lookup: Multiplikand und Multiplikator bilden die Adresse einer Speichereinheit, welche das gewünschte Resultat enthält.
- Entwurf eines Multiplizierers, welcher die Operation in der multiplikativen Gruppe selbst ausführt.
- Ausnutzung algebraischer Eigenschaften (Isomorphie der additiven und der multiplikativen Gruppe): Die Multiplikation modulo $p$ wird durch Addition modulo $p$-1 gelöst, wobei Transformation und Rücktransformation nötig werden. Die Technik entspricht etwa dem Verfahren der Multiplikation ganzer Zahlen, wo die Logarithmen der Operanden gebildet und aufaddiert werden und die Summe anschliessend durch Potenzieren rücktransformiert wird.

Aufgrund der Evaluationsresultate wurde ein Schema gewählt, bei dem die Multiplikation in der multiplikativen Gruppe selbst ausgeführt wird. Es gilt:

$$ab \bmod (2^n+1) = (ab \bmod 2^n - ab \operatorname{div} 2^n) \bmod (2^n+1) \qquad (2)$$

Somit wird zuerst eine Multiplikation der beiden 16-Bit-Operanden vorgenommen. Das 32-Bit-Resultat wird in ein nieder- und ein höherwertiges 16-Bit-Wort aufgeteilt. Anschliessend wird das höherwertige Wort vom niederwertigen subtrahiert und die Modulokorrektur durchgeführt.

Weil der Multiplizierer die grösste Hardwareeinheit auf den Chip darstellt, wurde die Mehrfachverwendung innerhalb einer Runde geprüft. Grundsätzlich könnte ein einzelner Multiplizierer alle vier Multiplikationen ausführen und vierfach gemultiplext werden. Bei Einsatz von zwei Multiplizierern kann der Datendurchsatz verdoppelt werden. Auch wird die Steuerungslogik auf dem Chip einfacher und die gesamte Verdrahtungslänge kürzer, weil die Multiplikationen im Innern der Runde physikalisch von denen am Eingang getrennt werden können. Eine Architektur mit vier Multiplizierern würde den Durchsatz noch einmal verdoppeln und die Steuerung weiter vereinfachen, da nun die Daten nicht mehr speziell verteilt werden müssten. Leider überstieg der Flächenbedarf für vier Multiplizierer die Siliziumfläche, die zur Verfügung stand. Deshalb wurde die Wahl getroffen, zwei Multiplizierer, diese jedoch mit zweistufiger Pipeline, zu implementieren. Mehr Pipelinestufen hätten keine weitere Verbesserung gebracht, da infolge der grossen Anzahl Äste im Datenflussgraph, welche neben einer Multiplikation vorbeiführen, jede zusätzliche Pipelinestufe fünf weitere 16-Bit-Register zur Datenverzögerung nötig gemacht hätten.

Abweichend vom Schema in Gleichung 2 wurde ein mit einer zweistufigen Pipeline versehener $17 \times 17$-Bit-Multiplizierer mit anschliessender Modulokorrektur gewählt. Das zusätzliche Bit des Multiplizierers wird benötigt, um den Fall, dass eine Null als Operand erscheint, richtig zu behandeln: Eine Null muss als $2^{16}$ interpretiert werden, da die multiplikative Gruppe keine Null kennt. Mit diesem Multiplizierer konnte eine Berechnungsdauer von 60 ns für eine 16-Bit Modulo-Multiplikation erreicht werden.

### Entwurf der Schnittstelle

Aufgrund der hohen Verschlüsselungsrate, die der Datenpfad zu leisten in der Lage ist, muss eine grosse Menge Daten zum Chip hin und von ihm weg transportiert werden. Damit

die Schnittstelle nicht zum Flaschenhals des Chips wird, wurde entschieden, den Datenpfad mit der halben, die Schnittstelle jedoch mit der vollen Frequenz zu takten. Das implementierte Protokoll wurde auf den synchronen Systembus der Sun Sparc-Station ausgelegt und erlaubt Datentransferraten bis maximal 30 MByte/s bei 25 MHz Systemtakt, während der Datenpfad mit einer Rate von 5,5 MByte/s (44,1 Mbit/s) chiffriert.

Der Chip ist in Bild 2 abgebildet. Deutlich zu erkennen sind die beiden $17 \times 17$-Bit Makrozellen-Multiplizierer und sechs RAMs, welche als Daten- und Schlüsselspeicher Verwendung finden. Die technischen Daten des Chips sind in Tabelle I aufgeführt.

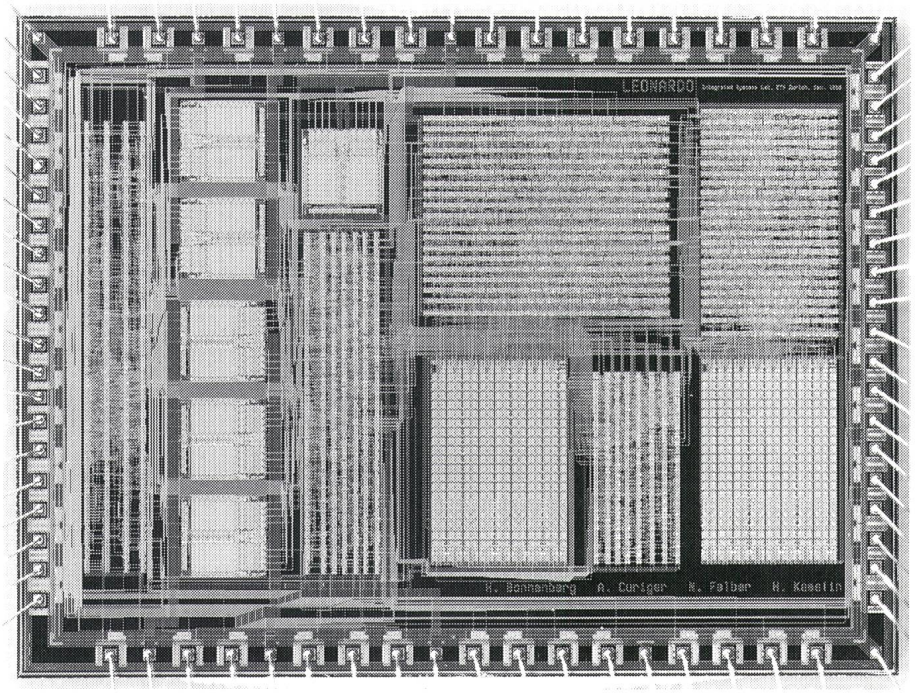



**Bild 2 Foto des Prototypenchips Leonardo**

## Test und Versuchsbetrieb

Der Chip wurde nach der Fabrikation am CSEM in Neuchâtel verpackt und anschliessend am IIS funktional getestet [10]. Messungen mit dem Asic-Tester haben das korrekte Funktionieren des Chips bis zu einer Frequenz von 33 MHz gezeigt. Anschliessend wurde der Chip in eine Sun Sparc-Station als Hostrechner eingebettet [14]. Am ISI wurden schliesslich verschiedene kryptographische Untersuchungen vorgenommen.

## Entwurf der definitiven VLSI-Schaltung

In der letzten Phase wurde der Optimierung der Chiparchitektur in Hinblick auf maximalen Datendurchsatz durch weitergehende Parallelverarbeitung und unter Einbezug von Blöcken in Full-Custom-Technik besondere Beachtung geschenkt. Neue Lösungen, um die Bedürfnisse der Testbarkeit einerseits und der Sicherheitsanforderungen der ANSI Standards bezüglich Funktions- und Abhörsicherheit [11] bei kryptographischen Systemen andererseits unter einen Hut zu bringen, wurden untersucht und implementiert.

### Funktionalität des Idea-Chips

Der Idea-Chip implementiert die fünf standardisierten Betriebsmodi [3] ECB (Electronic Code Book), CBC (Cipher Block Chaining), CFB (Cipher Feedback), OFB (Output Feedback) und MAC (Message Authentication Code). Diejenigen Modi, bei welchen Vorwärts- oder Rückwärtskoppelungen der Datenströme vorkommen (CBC, CFB, OFB und MAC), können allerdings das Pipelining, wie es im ECB-Modus verwendet wird, nicht ausnutzen. Im CBC-Modus zum Beispiel muss ein bereits verschlüsselter Datenblock mit dem unmittelbar folgenden unverschlüsselten Datenblock verknüpft werden. Bei einer $k$-stufigen Pipeline-Struktur werden jedoch jeweils $k$ Datenblöcke quasiparallel verarbeitet. Der erste verschlüsselte Datenblock kann also erst mit dem $(k+1)$-ten, dem ersten noch unverschlüsselten Datenblock, verknüpft werden. Eine originalgetreue Implementation verlangt somit, dass der zweite Datenblock verzögert an $(k+1)$-ter Stelle, der dritte an $(2k+1)$-ter Stelle usw. folgt. Dadurch kann nur der $k$-te Teil des (im ECB-Modus möglichen) Durchsatzes erreicht werden. Im Idea-Chip beträgt $k = 8$.

Um diesem Verlust an Verarbeitungsgeschwindigkeit entgegenzutreten, wurde zusätzlich ein neues Schema vorgeschlagen und in den Chip implementiert: Ein verarbeiteter Block wird nicht mit dem direkt nächsten, noch nicht verarbeiteten, sondern mit dem (k+1)-ten Block verknüpft. Mit dieser Massnahme kann die Pipeline wieder voll ausgenutzt werden. Gegenüber den Standardmodis werden jedoch nun $k$ voneinander unabhängige «Ketten» statt einer einzigen gebildet.

Das gesamte Key Management findet auf dem Chip selbst statt. Geladen wird nur noch der 128 Bit lange Schlüssel; alle Teilschlüssel werden automatisch intern generiert. Da der Chip vor allem für High-Speed Applikationen vorgesehen ist, verfügt er über drei Ports, zwei unidirektionale Daten- und einen bidirektionalen

| Technische Daten | |
|---|---|
| Prozesstechnologie | CMOS n-Wannen 1,5 µm (VLSI cmn16) |
| Anzahl Transistoren | 110 000 |
| Chipfläche | 64,9 mm2 |
| Datendurchsatz (25 MHz) | 44,1 MBit/s |
| Maximale Taktfrequenz | 33 MHz |
| Gehäuse | PGA 84 |
| Anzahl Pads | 76 |
| Teststrukturen | 6 partielle Scan-Pfade |

**Tabelle I Technische Daten von Leonardo**

Kontrollport. Der Chip ist zurzeit in Entwicklung. Erste getestete Muster sind Mitte Oktober zu erwarten.

## Anwendungen und Marktperspektiven

Als Anwendungsgebiete des neuen Verschlüsselungsalgorithmus kommen alle Bereiche der digitalen Daten- und Übertragungstechnik in Frage, bei denen Sicherheitsanforderungen in Hinblick auf die Vertraulichkeit der Daten oder die Prüfung der Zugriffsberechtigung zu Daten oder Betriebsmittel bestehen. Die Vermarktung des Algorithmus als Chip und als Programm wird durch Ascom vorgenommen. In Zukunft wird für alle Blockverschlüsselungsaufgaben ein VLSI-Chip sowie ein Softwareprogramm zur Verfügung stehen. Idea und seine Implementierungen wurden durch Ascom patentiert. Um den Algorithmus und den Chip allen interessierten Stellen zugänglich zu machen, sind folgende Möglichkeiten vorgesehen:

- Der publizierte Algorithmus darf für nichtkommerzielle Anwendungen (selbst) als Programm implementiert werden.
- Für kommerzielle Anwendungen als Softwareprodukt (z.B. Programmpaket auf PC oder Mainfraime) oder zur Herstellung von Chips werden Lizenzen vergeben.
- Verkauf der Chips auf dem freien Markt.

### Verdankung

An dieser Stelle sei Dr. H. Kaeslin, dem Leiter des Designzentrums der ETH Zürich, für seine Unterstützung herzlich gedankt. Dank gebührt auch der Kommission zur Förderung der wissenschaftlichen Forschung (KWF) für die finanzielle Unterstützung dieses Projektes.

### Literatur

[1] *J.L. Massey*: An introduction to contemporary cryptology. Proceedings of the IEEE, 76(1988)5, pp. 533–549.

[2] *R. Forré*: Die Chipkarte als kryptographisches Werkzeug. Bulletin SEV/VSE, 81(1990)5, S. 23–31.

[3] ISO/IEC 10116: Information technology – modes of operation for an *n*-bit block cipher algorithm, 1991.

[4] Cryptech NV/SA Brussels: The CRY12C102 DES Chip, 1989.

[5] *X. Lai* und *J.L. Massey*: A proposal for a new block encryption standard. In Advances in Cryptology – Eurocrypt '90, pp. 389–404, Berlin, 1990. Springer Verlag.

[6] *X. Lai, J.L. Massey* und *S. Murphy*: Markov ciphers and differential cryptanalysis. In Advances in Cryptology – Eurocrypt '91, pp. 8–13, Berlin, 1991. Springer Verlag.

[7] *C.E. Shannon*: Communication theory of secrecy systems. The Bell system Technical Journal, 28(1949)4, pp. 656–715.

[8] *R. Lidl* und *H. Niederreiter*: Introduction to finite fields and their applications. Cambridge University Press, 1986.

[9] *H. Kaeslin*: Asic-Entwicklung, wann und wie? Bulletin SEV/VSE,80(1989)15, S. 933–940.

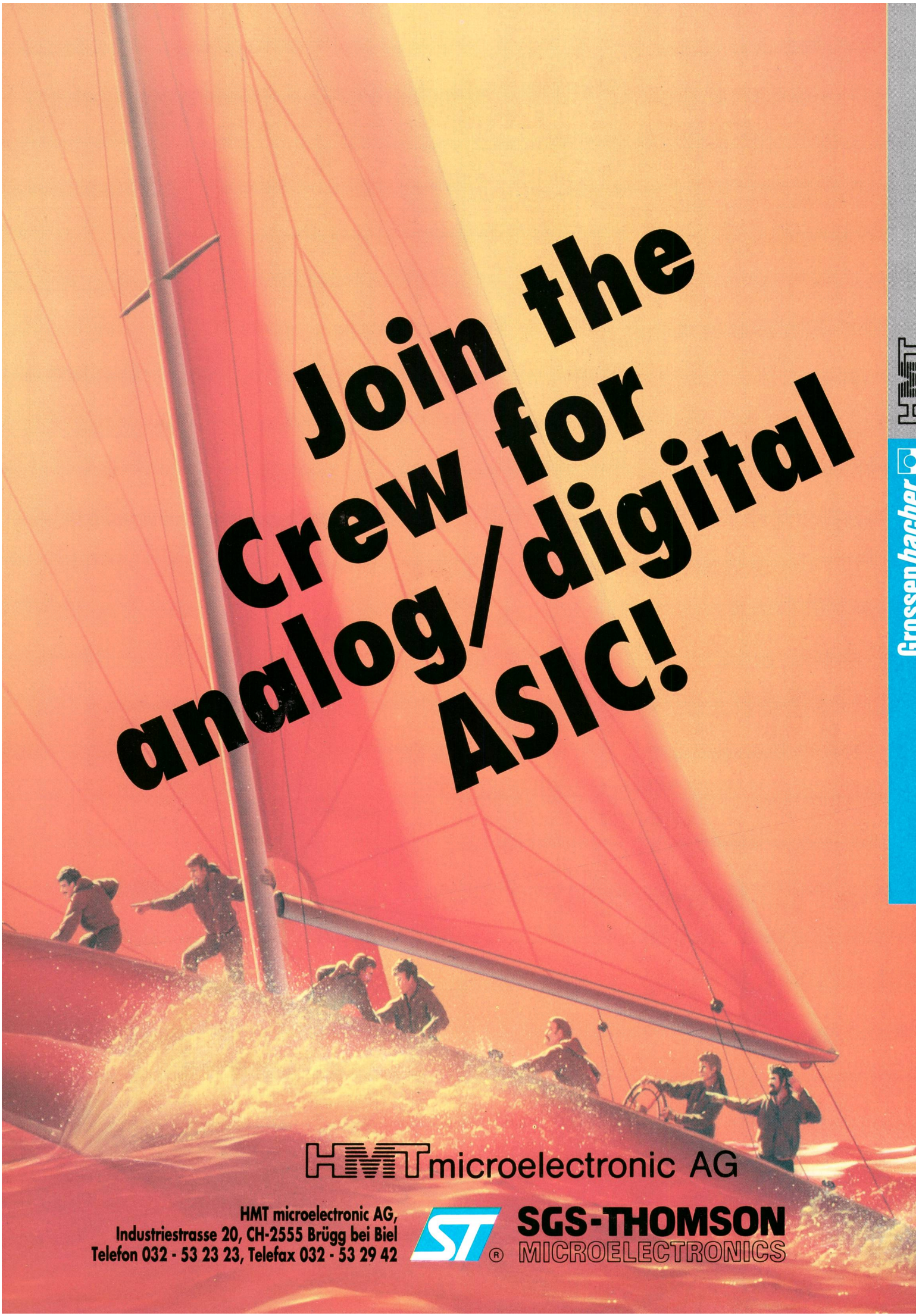
[10] *H. Bonnenberg* und *A. Curiger*: Leonardo – design and test report. Technical Report 91/6, Institut für Integrierte Systeme, ETH Zürich, Juni 1991.

[11] ANSI X9.17: Financial Institution Key Management, April 1985.

[12] *H. Bonnenberg, A. Curiger* und *H. Kaeslin*: Leonardo – design aspects of the VLSI implementation of a new secret key block cipher. Technical Report 90/5, Institut für Integrierte Systeme, ETH Zürich, April 1990.

[13] *A. Curiger, H. Bonnenberg* und *H. Kaeslin*: VLSI architectures for multiplication modulo $2^n+1$. IEEE Journal of Solid-State Circuits, 26(1991)7, pp. 990–994.

[14] *R. Zahir*: An SBus cryptography board for the sparc-station. Technical Report 91/25, Institut für Integrierte Systeme, ETH Zürich, November 1991.

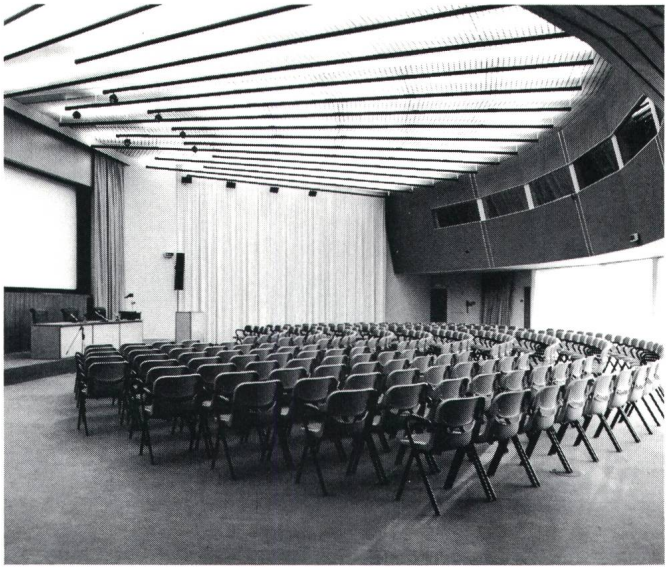

Join the Crew for analog/digital ASIC!
HMT microelectronic AG
HMT microelectronic AG,
Industriestrasse 20, CH-2555 Brügg bei Biel
Telefon 032 - 53 23 23, Telefax 032 - 53 29 42
SGS-THOMSON MICROELECTRONICS
Grossenbacher
HMT

# varintens® Lichtsteuerungen

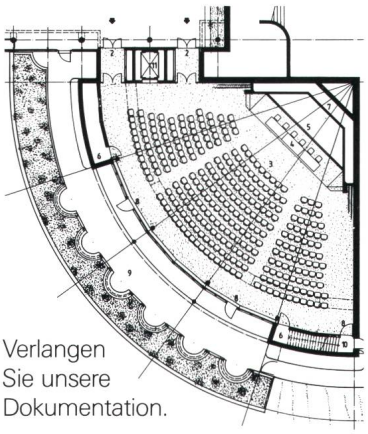
**Die Kombination von varintens-Lichtsteuerungen mit dem varintens-Intensiv-Pulser-System (VIP 90) ermöglicht ein Steuerverhältnis bei 26 mm-Leuchtstofflampen (18-36-58 W) von bis zu 1 : 10 000 mit Sofortstart in jeder Dim-Position.**

VARINTENS INTENSIVE PULSER VIP 90/S® VARINTENS INTENSIV-PULSGERÄT

Weil anspruchsvolle Steuerungen von Plenarsälen, Konferenzräumen und Aulen ein Steuerverhältnis von mindestens 1 : 1 000 erfordern, ist und bleibt das varintens-VIP-90-System die optimale Lösung bei höchster Betriebssicherheit.



Das Beispiel aus der Praxis: Licht-Helligkeitssteuerung varintens für das Auditorium im Forschungszentrum Nestlé in Vevey.



Verlangen Sie unsere Dokumentation.

## Praxiserprobte Konzepte und Anlagen für professionelle Anwender

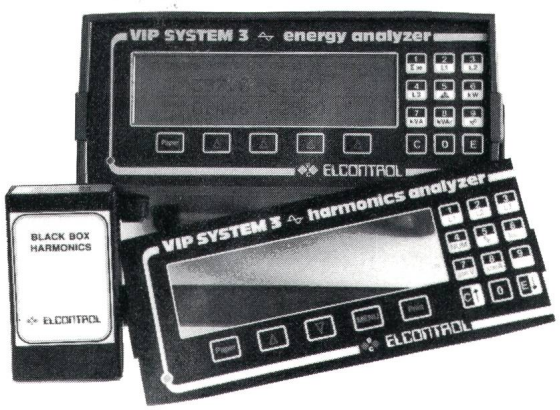
**starkstrom- elektronik ag**

se starkstrom-elektronik

Güterstrasse 11
CH-8957 Spreitenbach
Telefon: 056 / 70 13 75
Telex: 826 333 sew ch
Telefax: 056 / 71 49 86

# VIP-System 3

## Oberwellen-Analysator
und **Energie-Analysator**




Ihr Partner für die Elektroenergie-Optimierung seit 1965

**detron ag 4332 Stein**

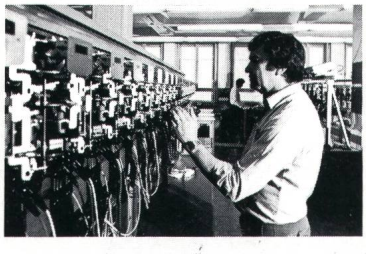
4332 Stein Tel. 064 - 63 16 73

Schweizerischer Elektrotechnischer Verein
Association Suisse des Electriciens

SEV ASE

SWISS CALIBRATION

## *Die SEV-Prüfstelle Zürich*



*Abteilung Eichstätte*
*revidiert, kalibriert und eicht*

- *Messinstrumente*
- *Elektrizitätszähler*
- *Messwandler*

**Ein Anruf genügt!**

Ihr Partner in der Elektrotechnik

*Schweizerischer Elektrotechnischer Verein,*
*Prüfstelle Zürich*
*Seefeldstrasse 301, Postfach, 8034 Zürich*
*Telefon 01 / 384 91 11 – Telex 817 431*
*Telefax 01 / 55 14 26*