# Bakkalaureatsarbeit

**Sicherheitsanalysetechniken von Mikrocontrollern**

Markus Kammerstetter,
mne (at) seclab tuwien ac at

11. März 2009

## Zusammenfassung

*In der folgenden Arbeit wird eine Einleitung zu den bestehenden Hardware-Angriffmethoden auf Mikrocontroller sowie einigen verfügbaren Gegenmaßnahmen gegeben.*
*Die Angriffsmethoden werden dabei in die Klassen der nicht-invasiven sowie invasiven Methoden eingeteilt, wobei die Klasse der semi-invasiven Methoden einen Spezialfall darstellt.*
*Es werden rein passive Techniken wie power- oder timing-Analyse behandelt, der Großteil der Arbeit beschreibt allerdings aktive Techniken. Die vorgestellten aktiven Techniken beginnen bei low-cost Methoden wie Glitching oder Datenremanenz und reichen bis zu neueren, invasiven oder semi-invasiven Techniken, die eine teils kostenintensive Laborausstattung wie Microprobing- oder FIB Workstations benötigen. Software-Angriffe oder -techniken werden nicht behandelt, da der Fokus der vorliegenden Arbeit im Bereich der Hardware Sicherheit liegt.*

**Schlüsselwörter:** Mikrocontroller, Sicherheit, Angriffe, Angriffsmethoden, Sicherheitsanalyse

## 1 Einleitung

Mikrocontroller finden heute breite Verwendung und übernehmen nicht nur einfache Steueraufgaben, sondern immer häufiger auch sicherheitskritische Funktionen. Um den Anforderungen gerecht zu werden implementieren Chiphersteller Sicherheitsfunktionen, die etwa das Auslesen des Programmcodes und somit des geistigen Eigentums eines Unternehmens verhindern sollen. Ist im konkreten Fall dennoch ein Auslesen möglich, so kann dies unter Umständen gravierende geschäftsschädigende Konsequenzen mit sich führen, da ein Konkurrent nun leicht einen Chip mit derselben Funktion herstellen und verkaufen könnte (*"chip cloning"*).
In den letzten 10 Jahren hat sich der Sicherheitsbedarf in diesem Bereich gesteigert. Von militärischer Verwendung oder Verwendung im Bankensektor hat die Technologie nun auch Einzug in den alltäglichen Bereich gefunden[2]: etwa um zu vermeiden, daß Batterien von Fremdherstellern in Mobiltelefonen oder Laptops verwendet werden können, dem Blockieren von Nicht-Original- oder Nachfüll-Patronen bei Druckern oder um zu gewährleisten, daß ein gekauftes Produkt nur vom Service-Center des Herstellers repariert wer-

den kann. Daneben existieren auch Applikation mit klarem Sicherheitsbedürftnis wie etwa portable Verschlüsselungsprodukte (DECT, GSM, etc.) oder Zutrittssysteme.

Chiphersteller von Mikrocontrollern befinden sich daher in einem andauernden Kampf gegen neue aber auch alte Angriffsmethoden. Die Bürde liegt bei den Herstellern, da sie ihre Produkte nicht nur gewissenhaft entwickeln und testen, sondern auch gegen (möglichst) alle Angriffe absichern müssen. Findet ein Angreifer nur eine Sicherheitslücke in der Implementation, so kann die Gesamt-Sicherheit des Produkts bereits kompromittiert sein. Eine ähnliche Problematik findet man auch in Softwareprodukten: So werden täglich sogar in vermeintlich sicheren Softwareprodukten neue Schwachstellen gefunden[1]. Während jedoch Softwarehersteller ihre Produkte mit Updates und Sicherheitspatches "nachbessern" können, haben Chiphersteller nur die Möglichkeit in darauf folgenden Chiprevisionen Fehler zu beheben. Die bereits in Produkten verbauten Bausteine bleiben jedoch verwundbar. Hinzu kommt, daß der richtigen Auswahl eines Mikrocontrollers für Sicherheitsapplikationen teilweise nur wenig Aufmerksamkeit geschenkt wird. Dies ist insbesondere der Fall weil nur wenig über die Absicherungsmaßnahmen von den Chipherstellern verfügbar ist[2]. Desweiteren kommt es auch vor, daß Chiphersteller nicht die notwendige Sorgfalt bei der Absicherung ihrer Controller treffen, sodass selbst bei Hochsicherheitsprodukten keine Garantien gegeben oder Verantwortung im Falle von Sicherheitsmängeln übernommen wird[2].

Smartcards sind üblicherweise besser geschützt, jedoch basieren auch sie auf demselben Coredesign mit zusätzlich hinzugefügten Sicherheitsmerkmalen. Manche Angriffe, überlicherweise invasiv, können auch noch erfolgreich auf Smartcards angewendet werden, jedoch sind sie meist aufwändiger und damit kostenintensiver[2].

Es ist anzumerken, daß so etwas wie absolute Sicherheit nicht existiert. Stehen einem Angreifer ausreichend Ressourcen (Zeit und Geld) zur Verfügung, wird jede Absicherung früher oder später brechen. Die Frage ist die Praktikabilität: Benötigt es 10 Jahre um die Sicherheit eines Geräts zu brechen, daß jedoch bereits in 3 Jahren durch ein noch besser abgesichertes Gerät ausgetauscht wird, so haben die Verteidigungsmaßnahmen den Kampf um die Sicherheit gewonnen. Das Ziel sollte es sein ein der Anwendung entsprechendes Sicherheitslevel zu erreichen, damit ein Brechen der Schutzmaßnahmen unverhältnismäßig teuer und zeitaufwändig wird.

Wie auch im Softwarebereich ist es aus der Sicht des Autors bei sicherheitskritischen Applikationen nötig nicht nur auf die vom Hersteller suggerierte Sicherheit zu vertrauen, sondern insbesondere auch unabhängige Sicherheitsanalysen durchzuführen.

In der folgenden Arbeit sollen nun einige Angriffsmethoden sowie mögliche Abwehrmaßnahmen auf Mikrocontroller vorgestellt werden. Die Aufteilung ist wie folgend: In Kapitel 2 werden Angriffs- und Analysemethoden behandelt, wobei zwischen nicht-invasiven, invasiven und semi-invasiven Methoden differenziert wird. In Kapitel 3 finden sich schließlich Schlußwort und Ausblick sowie anschließend noch das Literaturverzeichnis.

# 2 Analysetechniken und Angriffsmethoden

In diesem Kapitel sollen einige Angriffsmethoden sowie mögliche Abwehrmaßnahmen vorgestellt werden. Der Schwerpunkt liegt jedoch auf der Seite der Angriffstechniken, da diese zur Sicherheitsanalyse von bestehenden Controllern genutzt werden können. Einige der Techniken (wie beispielsweise die Verwendung von UV Licht oder sogenanntes "glitching") sind schon lange bekannt, sodass mittlerweile immer weniger

Bausteine existieren, die dafür anfällig sind. Dennoch sollten jene Techniken nicht unter den Tisch gekehrt werden, da sie teilweise in modifizierter Form auch in neueren Angriffsmethoden wieder erfolgreich zur Anwendung kommen und als "Basistechniken" verstanden werden können.
Im Folgenden findet eine Unterteilung in nicht-invasive sowie invasive Angriffe statt, wobei ebenfalls die noch relativ neue Klasse der semi-invasiven Methoden erläutert werden soll.

## 2.1 Nicht-invasive Angriffe

Bei den nicht-invasiven Angriffsmethoden ist es nicht nötig das Gehäuse des Bausteins zu öffnen. Stattdessen wird bei aktiven Attacken die Betriebsumgebung (Spannung, Clock-Signal, Temperatur, Kommunikationsprotokoll, ...) des ICs in einer Weise verändert, sodass Nebeneffekte auftreten, welche die Sicherheit kompromitieren können. Neben den aktiven Angriffen existieren auch rein passive Methoden, die durch Messungen und anschliessender Signalverarbeitung kritische Einblicke in die Implementation liefern können.

### 2.1.1 Zeitmessung (passiv)

Benötigen sicherheitsrelevante (z.B. kryptografische) Operationen auf einem Prozessor je nach Eingabe-Daten und geheimen Schlüssel unterschiedlich lange, so kann unter Umständen durch eine Analyse der Ausführungszeiten der geheime Schlüssel eruiert werden[3]. Der Grund für die Laufzeitunterschiede sind etwa Performance-Optimierungen um unnötige Iterationen zu überspringen, datenabhängige Sprünge oder Programm-Statements, RAM cache hits oder Prozessor Instruktionen (wie Multiplikation oder Division), die in nicht konstanter Zeit ausgeführt werden[3].

Diese sogenannten *timing attacks* zählen dabei zu den *sidechannel*[1] Angriffen, da sie auf Informationen (wie Zeit, Stromverbrauch, elektromagnetische Abstrahlung, Geräusch, ...) beruhen, die durch die physische Implementation eines Sicherheits- bzw. Kryptosystems entstehen.

Um den Angriff durchzuführen ist es zuerst nötig Informationen über das Design (wie Architektur und Implementation) zu haben. Anschliessend werden Nachrichtenpaare zusammen mit ihrer genauen Ausführungszeit gesammelt (wie beispielsweise die genaue Zeit zwischen einem "Request" und der darauf folgenden "Response"). Liegen genug Nachrichtenpaare vor, so können unter genauer Analyse des Designs sensible Daten wie geheimes Schlüsselmaterial gewonnen werden.

Nachdem die Methode 1996 publiziert wurde, konnte sie erfolgreich auf einigen Smartcard-Implementationen des RSA Signaturalgorithmus durchgeführt werden, womit der geheime RSA Schlüssel auslesbar war.

Derartige Angriffe können nicht nur kryptografisch relevant sein. So existieren Mikrocontroller oder Zutrittssysteme (wie etwa Maxim iButton[5]) die fixe Daten wie Passwörter, Schlüssel oder Seriennummern als Schutzmaßnahmen einsetzen. Ist die Implementation anfällig für derartige timing Angriffe, so kann der jeweilige Schutz umgangen werden.

Die häufigste Schwachstelle tritt bei diesen Systemen auf wenn etwa die Seriennummer mit einer Datenbank Byte-weise verglichen wird, jedoch ein unkorrektes Byte sofort zum Abbruch des Authentifikationsvorganges führt[2]. Anstatt eines zeitaufwendigen Brute-Force Angriffs kann ein gültiger Schlüssel nun mit einer relativ kleinen Anzahl von Versuchen eruiert werden.

Um derartige Schwachstellen zu vermeiden ist beim Design sehr genau darauf zu achten, daß die Ausführungszeit von sicherheitsrelevanten Operationen unabhängig von den Daten ist.

[1] "Seitenkanal"

Dies wurde etwa bei der Motorola 68HC08 Mikrocontroller Familie wie folgt erreicht: Der ROM Bootloader erlaubt lediglich Zugriff auf den Flash Speicher, wenn zuvor das richtige 8-Byte lange Passwort eingegeben wurde[4].
Um die Ausführungszeit der Passwort-Überprüfung datenunabhängig zu gestalten, wurden zusätzliche NOP Instruktionen eingefügt, sodass die Authentifizierung immer gleich lange dauert.

Desweiteren verwenden einige Mikrocontroller einen internen RC Oszillator, dessen Frequenz abhängig von der Betriebsspannung, sowie der Die Temperatur ist. Für eine Zeitanalyse müsste ein Angreifer daher nicht nur eine sehr stabile und rauschfreie Spannungsversorgung verwenden, sodern auch die Die-Temerpatur stabilisieren.

Einige Smartcards verwenden als Schutz gegen timing-Angriffe eine ähnliche Methode, indem sie ein intern randomisiertes Clock-Signal verwenden.

### 2.1.2 Strommessung (passiv)

Der Stromverbrauch eines Prozessors ist abhängig von seiner derzeitigen Aktivität, wobei nicht die Zustände selbst, sondern vielmehr die Zustandsübergänge dafür ausschlaggebend sind. Dies liegt in der Natur der verwendeten CMOS Transistoren. So muss etwa ein NOT-Gatter beim Anlegen einer Eingangs-Spannung zuerst einige (beispielsweise kapazitive) Hürden überwinden, bis schließlich der Schaltvorgang beendet und das Ausgangssignal ausgegeben werden kann. Durch die Verzögerungen, die beim Schaltvorgang entstehen, wird daher auch die maximale Arbeitsfrequenz eines Gatters beschränkt.

Schaltet man einen kleinen Widerstand (z.B. 10 Ω) zwischen Gatter und Spannungsversorgung, so wird der Stromverbrauch über den Spannungsabfall am Widerstand messbar. Je kleiner der Widerstand, desto mehr Strom kann über den Widerstand fließen. Allerdings wird dadurch auch der Spannungsabfall (und somit die Messpräzision) kleiner. Zieht der Verbraucher viel Strom, so kann der Spannungseinbruch unter Umständen so groß werden, daß beispielsweise eine angeschlossener Mikrocontroller nicht mehr normal arbeiten kann. (So wäre entsprechend dem ohmschen Gesetz[2] bei einem Stromverbrauch von 100 mA an einem 10 Ω Widerstand bereits ein Spannungsabfall von 1 V zu erwarten.)

Statt einem Widerstand kann auch ein kleiner Ferrit-Transformator (Induktionsprinzip) verwendet werden. Der Stromfluß ist dadurch kaum limitiert, jedoch geht die Gleichspannungs-Komponente in der Messung verloren.

Handelt es sich um einen Prozessor mit bekannter Architektur und rein serieller Programmabarbeitung (dh. beispielsweise kein Pipelining), so kann anhand der Strommessung bestimmt werden, welche Instruktion gerade abgearbeitet wird. Konkret könnte man etwa bei einem 1 MHz Prozessor ein Referenz-Programm mit shift-left Instruktionen ablaufen lassen und würde dabei den Stromverbrauch pro Taktzyklus mit einer wesentlich höheren Samplingfrequenz messen (z.B. 50 MHz bei mindestens 12 bit ADC Auflösung). Unter der Annahme von möglichst rauschfreien Messungen (die etwa durch Mittelung von vielen identen shift-left Instruktionen erreicht werden kann), so bekommt man eine *power trace* von der Instruktion (eine Art Fingerabdruck). Führt man nun eine Messung an demselben Prozessortyp mit unbekanntem Programm durch, so werden die ausgeführten Instruktionen durch den Vergleich mit den Referenz-Messwerten identifizierbar. Da jedoch der Stromverbrauch der Instruktionen auch von den Daten abhängig ist, können mit etwas Aufwand sogar die Operanden-Daten rekonstruiert werden.

Diese Technik wird dabei als *simple power analysis (SPA)* bezeichnet.

Im Gegensatz dazu steht die mächtigere *differentielle* Analyse (*differential power analysis (DPA)*),

[2] $R = \frac{U}{I}$

die zusätzlich mit statistischen Methoden arbeitet. Diese Methoden identifizieren dabei kleine Unterschiede im Stromverbrauch, sodass etwa einzelne Bits eines geheimes Schlüssels identifiziert werden können. Dadurch sind auch mit weniger detailiertem Wissen über die Implementation Angriffe möglich[2].

Neben der Messung des Stromverbrauchs sind derartige Angriffe auch mit anderen Messungen wie etwa der elektromagnetischen Abstrahlung möglich bzw. kombinierbar.

Schutzmaßnahmen sind etwa die Verwendung von internen Konsensatoren mit hoher Kapazität oder von Schaltungen die den Stromverbrauch künstlich randomisieren. Da jedoch Kondensatoren mit hoher Kapazität auch entsprechend groß sind, ist der Einsatz dieser Technik bei zunehmender Miniaturisierung jedoch schwierig.

### 2.1.3 Brute force Angriff (aktiv)

Im kryptografischen Sinn versteht man unter einem Brute force Angriff das "Durchprobieren" aller möglichen Schlüssel. (In der Literatur wird dies auch als *exhaustive key search* bezeichnet.) Da ein derartiger Angriff nicht effizient (und damit sehr zeitaufwändig ist), werden üblicherweise mehrere Computer oder spezielle Hardware (wie FPGA Arrays) verwendet.

Bei der Analyse von Halbleiter Bausteinen ist die Bedeutung jedoch anders. Anstatt viele mögliche Schlüssel zu testen versucht man durch "intelligentes Probieren" mehr über den Baustein herauszufinden um diese Erkenntnisse eventuell in einem späteren Angriff nutzen zu können.

Hierfür gibt es mehrere Beispiele, die das Vorgehen anschaulicher machen sollten:

Bei kleinen Designs, die in einem ASIC oder CPLD[3] untergebracht sind, könnten etwa alle möglichen Kombinationen von Input-Signalen an den Eingängen angelegt werden. Um mehr über die Funktionen des Bausteins herauszufinden werden zur selben Zeit die Output-Pins beobachtet. Da der Aufwand exponentiell ansteigt, ist diese Methode bei größeren Designs mit vielen I/O Pins nicht mehr praktikabel.

Eine andere Methode wäre an die Pins eines Chips eine im Vergleich zur Betriebsspannung hohe Spannung (üblicherweise die doppelte Betriebsspannung) anzulegen. So lässt sich evt. herauszufinden, ob einer dieser Pins den Chip in den Herstellertest- oder Programmiermodus versetzt. Derartige Pins lassen sich auch mit einem Multimeter finden, da die Chip-Eingänge im Vergleich zu normalen Eingängen intern keine schützende Diode zur Stromversorgung geschaltet haben[2]. Ist der Test- bzw. Programmiermodus gefunden, kann ein systematisches Anlegen von Eingangssignal-Kombinationen (wie oben beschrieben) verwendet werden um deren Funktion im jeweiligen Modus herauszufinden.
Ist ein derartiger Angriff erfolgreich, könnten damit unter Umständen sensible Speicherbereiche ausgelesen oder modifiziert werden.

Eine ähnliche Methode wäre es alle möglichen Zustände im Kommunikationsprotokoll mit dem Chip zu testen. Auf diese Weise könnten versteckte Funktionen, die von den Entwicklern zu Test- oder Upgrade-Zwecken implementiert wurden, aufgedeckt werden.

In Halbleiter Chips sind derartige Testinterfaces häufig zu finden, da sie vom Hersteller für das abschließende Testen des Bauteils in der Produktion genutzt werden.
Als Schutzmaßnahme werden diese Interfaces bei Smartcards meist nach der Produktion physisch abgeschnitten.

[3] Complex programmable logic device

### 2.1.4 Clock Glitching (aktiv)

Jeder Transistor in einem Prozessor wirkt ähnlich einem RC Element und hat damit eine charakteristische Verzögerungszeit. Die maximale Taktfrequenz eines Prozessors ergibt sich unter anderem aus den Verzögerungen der einzelnen Elemente. Aus diesem Grund hat auch jedes Flip-Flop ein individuelles, charakteristisches Zeitfenster in welchem sein Eingang abgetastet (setup time SU) und der Ausgang entsprechend verändert wird. Während sich dieses Fenster bei allen Flip-Flops innerhalb der spezifizierten Setup-Zeit bewegt, ist der genaue Abtast-Zeitpunkt dennoch für jedes Flip-Flop in einem Halbleiterchip unterschiedlich und von Spannung sowie Temperatur abhängig[2].

Bei einem Clock Glitch verwendet man einen Clock-Puls der wesentlich kürzer als normal ist. Dadurch bedingt werden einige der Flip-Flops (je nach Zeitpunkt des Glock Glitches) einen falschen Zustand übernehmen. Die Glitches können somit verwendet werden um im Programmcode eines Controllers gezielt bedingte Sprünge oder Tests zu überspringen bzw. sogar zu überschreiben.

Ein Beispiel für eine derartige Schwachstelle ist der Mask ROM Bootloader des Motorola MC68HC05B6 Mikrocontrollers, der verhindert, daß user-code geladen wird, falls das Sicherheitsbit im EEPROM gesetzt ist. In diesem Fall geht der Bootloader in eine Endlosschleife. Diese Sicherheitsmaßnahme ist nun mit einem Clock Glitch vergleichsweise einfach zu umgehen, da selbst bei zufälligen Glitch Zeitpunkten die Chancen sehr hoch sind, daß es zu einem Fehler im CPU kommt und die Ausführung des Programms nach der Endlosschleife fortgeführt wird[2].
Ein Beispiel wäre etwa, daß der Program Counter (PC) bereits inkrementiert wird, bevor die Sprunginstruktion der Schleife ausgeführt und den PC Wert mit der Sprungaddresse überschreiben wird. Das Resultat wäre die Fortführung des Programms direkt nach der Schleife.

Derartige Clock Glitch Signale können auf einfachstem Weg erzeugt werden, indem der übliche Quartz Kristallresonantor über kurze Zeit kurzgeschlossen wird. Der Grund dafür ist, daß ein derartiger Resonator zu Beginn auch auf Harmonischen seiner Eigenfrequenz schwingt und dadurch häufig Glitches produziert. Erst nach seiner Einschwingzeit gibt er oberwellenfreie Schwingungen ab.
Bessere Erfolge lassen sich allerdings mit einem *Pattern Generator* oder einem entsprechend programmierten FPGA Evaluierungsboard erzielen.

Gegenmaßnahmen sind beispielsweise die Verwendung von Clock Monitor Schaltungen, die den Prozessor rücksetzen, sobald hohe Frequenzen erkannt werden.
Viele Hersteller behaupten entsprechende Hochfrequenz Detektoren in ihren Chips eingebaut zu haben, bei näherem Hinblick sind dies allerdings oft nur einfache low-pass Filterschaltungen. Diese Filterschaltungen können durch eine Anpassung des Duty-Cycles des Clock-Signals während des Glitch Angriffs umgangen werden[6].

### 2.1.5 Power Glitching (aktiv)

Ändert man die Versorgungsspannung, so kann sich auch die Threshold-Spannung der Transistoren ändern. Aus diesem Grund ist es möglich, dass einzelne Flip-Flops entweder ihren Eingang zu unterschiedlichen Zeiten abtasten oder sogar einen falschen Logikzustand übernehmen[6]. Bei einem Angriff durch Power Glitching nutzt man genau dieses Verhalten, indem für einen kurzen Zeitpunkt (meist 1-10 Taktzyklen) die Versorgungsspannung entweder erhöht oder vermindert wird[2].

Ein Beispiel für einen derartigen Angriff wäre das Umgehen des Security Bits im MaskROM Bootloader code des bereits erwähnten Motorola MC68HC05B6 Mikrocontrollers. In diesem Fall wird das Security Bit aus dem EEPROM gelesen. Senkt man jedoch die Versorgungsspannung um 50-70% ab, so wird anstatt eines Null-Bytes ein Byte mit einem Wert von FFh aus dem EEPROM ausgelesen, da mit der Versorgungsspannung auch

die Threshold Spannung der jeweiligen Flip-Flops abgesenkt wird. Aufgrund des falsch ausgelesenen Security-Bytes ist nun in Folge die eingesetzte Absicherung wirkungslos.

In älteren Versionen des PIC16F84 Mikrocontrollers kann hingegen die Security Fuse zurückgesetzt werden. Dabei wird normalerweise bei der *chip erase* Operation zuerst der gesamte Speicher gelöscht und anschließend die Security-Fuse zurückgesetzt. Eine für einige Millisekunden erhöhte Versorgungsspannung von etwa 10 V führt jedoch zum frühzeitigen Abbruch der Lösch-Operation, wobei die Security-Fuse dennoch zurückgesetzt wird. Aus diesem Grund kann im Weiteren der Inhalt des Speichers ausgelesen werden. In der späteren Variante PIC16F84A wurde eine Schutzmaßnahme eingebaut, sodass alle Speicher-Modifikationen sofort abgebrochen werden, wenn die Versorgungsspannung außerhalb des Bereiches von 3 V bis 6 V gelangt. Interessanterweise ist bei dieser Revision derselbe Angriff immer noch möglich, wobei die nötige Glitch Spannung nun lediglich 50 mV über der spezifizierten Versorgungsspannung und damit sogar innerhalb der Spezifikation liegt[2].

Ebenfalls interessant ist die Tatsache, daß es in einigen Designs Clock-Monitor Schaltungen gibt um sich vor Clock-Glitches zu schützen, diese jedoch manchmal durch eine sorgfältig festgelegte Änderung in der Versorgungsspannung umgangen werden können[6].

Insbesondere durch die Kombination von Clock- und Power-Glitching Angriffen kann möglicherweise ein entsprechender Glitching Schutz übergangen werden[6].

Wie bereits erwähnt ist eine mögliche Schutzmaßnahme vor Power Glitching Angriffen die Verwendung eines Spannungsversorgungs-Sensors, welcher etwa den Prozessor zurücksetzt sobald eine Spannung ausserhalb des Spezifikationsbereiches detektiert wird. Eine andere hilfreiche Methode ist die Verwendung von Kondensatoren um die Spannungsversorgung zu glätten. Wie bereits im Kapitel zur Strommessung erwähnt, kann die Technik aufgrund der nötigen Kondensatorgröße jedoch oft nicht effizient oder gar nicht in Halbleiter Chips angewendet werden.

### 2.1.6 alternative Formen von Glitching (aktiv)

Neben den vorgestellten Methoden des *clock-* und *power-* glitching existieren auch weniger verbreitete Variationen dieser Angriffstechniken. Darunter fallen auch die sogennanten *electrical field glitching* Angriffe, bei welchen zwei nur wenige Mikrometer entfernte Metallnadeln über sensible Bereiche eines Chips positioniert werden. Legt man für weniger als eine Mikrosekunde eine hohe Spannung im Bereich von einigen hundert Volt an, so wird von den Nadeln ein elektrisches Feld erzeugt, welches stark genug ist, um die Threshold-Spannung von naheliegenden Transistoren zu verändern[6]. Derartige Angriffe setzen jedoch möglicherweise bereits das Öffnen des IC Gehäuses voraus, womit sie nicht mehr zur Klasse der nichtinvasiven Angriffsmethoden zählen würden.

Neben einem elektrischen Feld ist auch der Einsatz eines elektromagnetischer Impulses[4] denkbar, der aufgrund der Induktion in den Leiterbahnen im Halbleiter ähnliche Effekte verursachen kann[2]. Dieser wurde auch als weitere Verbesserung der oben vorgestellten Methode verwendet[7]. Anstatt von zwei Nadeln wird Strom in eine kleine Spule mit einigen hundert Wicklungen geleitet, wobei die Spule selbst um eine Wolfram *microprobing* Nadel gewickelt ist, die die Feldlinien auf einen Punkt konzentriert.

### 2.1.7 Daten Remanenz (aktiv)

**Sicherheits-Controller** speichern geheimes Schlüsselmaterial überlichweise im statischen RAM (SRAM) ab, welcher von einem mit

[4] electro-magnetic pulse (EMP)

Tamper-Sensoren geschützten Gehäuse umgeben ist. Detektieren diese Tamper-Sensoren einen Eingriff, so wird die Versorgungsspannung gekappt und die Daten im SRAM gehen verloren. Es ist jedoch allgemein bekannt, daß ein Absenken der Temperatur unter $-20°C$ dazu führt, daß die Daten noch für einige Sekunden bis hin zu mehreren Minuten erhalten bleiben[8]. Bleiben die Daten länger erhalten wie ein Angreifer zum Öffnen des Schutzgehäuses braucht, so können eventuell entsprechend sensible Daten ausgelesen werden. Als Gegenmaßnahme wurden Temperatursensoren zum Design hinzugefügt, sodass Temperaturen unter $-20°C$ detektiert und der Speicher aktiv gelöscht werden kann.
Interessanterweise haben Messungen gezeigt, daß die Remanenz Zeit im Gegensatz zu diesem Wissen bei einigen Speicherchips auch bei Temperaturen oberhalb von $-20°C$ gefährlich lange sein kann. Insbesondere low-power Varianten haben lange Remanenz Zeiten[8].
Neben SRAM sind auch andere Speichertypen wie DRAM, EEPROM oder Flash betroffen.

Die Methode ist als aktiver Angriff zu klassifizieren, da die Temperatur in einen entsprechenden Bereich abgesenkt werden muß.

**Nicht-volatile Speicher** wie EEPROM oder Flash haben im Gegensatz zu SRAM Zellen nicht nur zwei stabile Logikzustände. Vielmehr speichern diese Zellen ihren Wert in Form einer Analogspannung mittels der Ladung am *floating-gate* eines MOS Transistors. Dabei verschiebt die floating-gate Ladung die Threshold Spannung des Transistors. Der Inhalt kann danach wiederum mit einer *sense-amplifier* Schaltung ausgelesen werden. Moderne Flash Speichertypen verwenden gar mehrere Ladungsniveaus, sodass die effektive Speicherdichte erhöht werden kann.

Wird eine Speicherzelle gelöscht, so wird die Threshold Spannung nach unten in einen niedrigen Bereich verschoben. Unter normaler Betriebsumgebung wird von der sense-amplifier Schaltung nun für jede Zelle derselbe Logikwert (etwa '1') ausgegeben. Bei vielen älteren Speichertypen (besonders UV-EPROM) wird jedoch das Threshold Level der sense-amplifier Schaltung in etwa durch die Hälfte der Versorgungsspannung festgelegt. Ist es somit möglich den gelöschten Speicher mit einer ensprechend niedrigen Versorgungsspannung zu betreiben, so kann auch das Threshold Level nach unten gezogen und die bereits gelöschten Inhalte des Speichers ausgelesen werden[2].

Selbst bei neueren Speichern kann ein ähnlicher Ansatz immer noch zum Ziel führen. Ein Beispiel ist der PIC16F84A Mikrocontroller, der erst nach dem vollständigen Löschen des EEPROM- und Flashspeichers die Security Fuse zurücksetzt. Wird etwa der Programmspeicher (Flash) nach dem Löschvorgang ausgelesen, so haben alle Bits den Wert 1. Genau wie im zuvor beschriebenen Beispiel wird beim Löschvorgang ebenfalls die Threshold Spannung der Speichertransistoren nach unten verschoben, jedoch ist die Spannung noch um einiges niedriger. Der oben beschriebene Angriff den Baustein mit niedriger Versorgungsspannung zu betreiben ist allerdings nicht möglich, da der Chip unter 1.5 V die Arbeit verweigert. Eine Lösung ist jedoch einen Power Glitch anzuwenden, sodass die Versorgungsspannung für eine kurze Zeit auf 1 V einbricht.
Diese niedrige Spannung reicht gerade noch aus, damit die Informationen vom Speicher in den internen Buffer des PIC16F84A Bausteins geraten. Allerdings ist die Spannung immer noch zu hoch um die Referenz Spannung der sense-amplifier Schaltung derart weit nach unten zu ziehen, daß die bereits gelöschten Zellen ausgelesen werden könnten.

Ein weiterer Trick ist nun erneut eine Lösch-

Operation durchzuführen. Während dieser Operation kommt nochmals genau abgestimmtes Power Glitching zum Einsatz, welches dazu führt, daß sich die Speicherzelle zwar minimal auflädt, sie jedoch durch einen rechtzeitigen Einbruch der Versorgungsspannung nicht durchtunneln kann[2]. Der Effekt ist, daß für kurze Zeit (bis zu einer Sekunde) die Ladung in der Speicherzelle gefangen ist und damit die Transistor Threshold Spannung angehoben wird.
Da durch den angewendeten Trick die Threshold Spannung temporär erhöht werden konnte, wird der bereits gelöschte Inhalt der Speicherzelle wieder auslesbar[2].
Interessanterweise funktioniert dieser Angriff auf den Flash Speicher des PIC16F84A Mikrocontrollers sogar noch nach 100 Lösch-Zyklen.
Selbiger Angriff funktioniert auch auf den EEPROM, jedoch ist die Threshold Spannung nach etwa 10 Lösch-Zyklen in etwa bei der einer vollständig gelöschten Zelle angelangt.

## 2.2 invasive Angriffe

Invasive Angriffe sind wesentlich aufwändiger als nicht-invasive Methoden, da nicht nur das Gehäuse des integrierten Schaltkreises geöffnet werden muß, sondern auch teure Laborgeräte wie Lasercutter, Elektronenmikroskope (z.B. SEM[5]) bzw. FIB-[6] oder Microprobing Workstations zur Anwendung kommen.
Während diese Techniken vielversprechend sind, benötigt ein Angreifer somit neben einem hohen Maß an Wissen sowie Geschicklichkeit auch Zugang zu einem entsprechend ausgestatteten Labor. Aufgrund der aufwändigen und teuren Geräte befinden sich allerdings entsprechende Labors meist nur bei größeren Halbleiterunternehmen oder Universitäten. Zwar existiert auch ein Gebrauchtmarkt für diese Geräte, jedoch sind die Kosten und die Aufrechterhaltung für eine Privatperson wohl kaum tragbar. (So ist beispielsweise eine gebrauchte FIB Workstation für weniger als 50.000 € zu haben[2], jedoch kommt zusätzlich zum Preis und Platzbedarf noch der hohe Aufwand um das dauerhaft erforderliche Ultrahochvakuum ($\approx 10^{-15}$ mbar) mit entsprechenden Pumpen (z.B. Cryo-Pumpen) aufrecht zu erhalten.)

[5] scanning electron microscope
[6] focused ion beam

### 2.2.1 Proben Vorbereitung

Invasive Angriffe setzen entweder einen teilweise oder vollständig aus dem Gehäuse ausgelösten Chip voraus. Für Mikrocontroller reicht es normalerweise aus, das Gehäuse nur teilweise zu entfernen. Wichtig dabei ist, daß das Bauteil nach dieser Prozedur noch funktionsfähig ist und etwa in einem Programmiergerät getestet werden kann. Bei manchen Controllern ist dies jedoch nicht möglich, da die Verbindungen (*bonding*) vom Halbleiter zu den Gehäusekontakten (Pins) beschädigt oder gänzlich zerstört werden. Entsprechend ist es dann notwendig den Die in einem Testgehäuse mit Aluminium- oder Goldverbindungen neu zu bonden. Eine Alternative ist die Verwendung von Microprobing Nadeln in einer Probe Station. Klarerweise wird dies jedoch mit zunehmender Anzahl der Pins mühseliger.

Soll der Chip unter einer FIB Workstation oder einem SEM Mikroskop analysiert werden, so ist es außerdem notwendig die Chip Oberfläche etwa mit einem Sputter mit einer dünnen Gold-Schicht zu überziehen[2]. Steht eine neuere FIB Workstation mit optischer Navigation zur Verfügung, so ist zumindest die Gold-Beschichtung nicht mehr notwendig.

### 2.2.2 Öffnen des Chip Gehäuses

Während es mehrere Methoden zum Öffnen des meist aus Epoxidharz bestehenden Gehäuses gibt, ist eine Kombination aus mechanischen und chemischen Techniken wohl die einfachste.

Besonders die chemischen Methoden erfordern Vorsicht und ausreichende Schutzmaßnahmen. Zum Öffnen des Chip Gehäuses wird zuerst an der Stelle, an welcher sich der Chip Die befindet, eine Mulde hineingefräst (etwa mit einem Drehmel und einem passenden Fräßkopf).
In diese Mulde wird nun langsam rauchende Salptersäure ($HNO_3 > 95\%$) getropft.
In Folge kohlt das Epoxid-Gehäuse schichtweise ab.
Sollten im Gehäuse relevante Teile aus Kupfer oder Silber sein, so werden diese allerdings ebenfalls angegriffen. Um das Problem zu vermeiden, kann eine Mixtur aus rauchender $HNO_3$ und konzentrierter $H_2SO_4$ verwendet werden. Dadurch wird die Reaktion bei einigen Gehäusematerialien nicht nur schneller, sondern auch das Silber oder Kupfer der Bondingpads kaum angegriffen. Um ein gutes Ätzergebnis zu erhalten, ist es auch empfehlenswert den IC vorher auf $50°C$ - $70°C$ aufzuheizen. Der Die selbst ist von einer Passivierungsschicht (z.B. aus $SiO_2$) umgeben, die nicht angegriffen wird. (Hierzu wäre etwa $HF$ nötig).
Die Ätzrückstände können schließlich in einem Acetonbad mit einem Ultraschallreiniger entfernt werden.
Nach weiterem Waschen in Aceton und dem Trocknen (etwa mit einem Haushaltsfön) sollte ein sauberer und voll funktionsfähiger Chip vorliegen[2].

Als Alternative existieren kommerzielle Anbieter wie etwa FIB International Inc.[7], die derartige Services anbieten.

### 2.2.3 Modifikation von Chip Schichten

Ein üblicher CMOS chip besteht aus mehreren Schichten. Die tiefsten Schichten im Substrat sind die dotierten Schichten, die beispielsweise Transistoren bilden. Um das Gate eines Transistors von der darunter liegenden dotierten Schicht zu trennen, wird eine dünne Gate-Oxidschicht verwendet. Das Gate selbst sowie die Zwischenverbindungen bestehen aus einer Poly-Siliziumschicht. Zur Isolierung kommt auch hier wieder eine Oxidschicht zu Anwendung. Darauf befinden sich nun die eigentlichen metallischen Leiterbahnen des Chips, die meist aus Aluminium bestehen. Durchkontaktierungen zu anderen Layern (sogenannte *via plugs*) bestehen meist aus Aluminium, Wolfram oder Titan. Zum Schutz des Dies wird der Chip Aufbau abschließend mit einem Passivierungslayer aus Siliziumdioxid $SiO_2$ oder -nitrid $Si_3N_4$ versehen. (Die Passivierungsschicht wurde bereits im vorigen Kapitel erwähnt.)

Im weiteren Vorgehen kann man nun entweder die schützende Passivierungsschicht entfernen um an die metallischen Leiterbahnen mit microprobing Prüfnadeln heranzukommen oder noch invasivere Techniken einsetzen, um in tiefere Schichten vorzudringen (etwa um die interne Struktur des Chips zu analysieren).

Die allgemeinen Techniken zum Abtragen von dünnen Schichten sind:

- **nasschemisches Ätzen**

  Hier werden je nach Material verschiedene Säuren oder Säure-Mischungen verwendet. Der Nachteil liegt in der isotropischen Ätzwirkung, die unerwünschte Nebeneffekte verursachen kann (etwa metallische Leiterbahnen die sich von der Oberfläche des Chips ablösen).
  Zusätzlich bieten derartige Methoden leider oft kein gleichmäßiges Ätzergebnis, da die Ätzgeschwindigkeit von der Angriffsfläche abhängig ist[2].
  Einige genauere "Ätzrezepte" für Halbleiter können in der Literatur gefunden werden[2][9].

[7] `http://www.fibinternational.com`

- **Plasma- bzw. Trockenätzen**

  Bei dieser Ätztechnik (auch reaktives Ionenätzen genannt[8]) wird eine Niederdruck-Kammer mit einem vom Material abhängigen Prozessgas (Ätzgas) gefüllt.
  Dieses Gas wird durch Ionisierung (etwa mit einem hochfrequenten elektrischen Feld[9]) zu Plasma, sodass Radikale im Gas entstehen. Diese reagieren in Folge mit der Oberfläche des Chips und tragen diese langsam ab. Die dabei entstehenden Ätzprodukte können von der Kammer abgepumpt werden. Der große Vorteil dieser Technik liegt in der durch die Richtung der Ionen vorgegebenen, stark anisotropischen Ätzwirkung, die nur dort auftritt, wo auch Ionen auf die Chip Oberfläche prallen.

- **mechanisches bzw. chemisch-mechanisches Polieren (CMP)**

  Beim rein mechanischen Polieren verwendet man spezielle Poliermittel[10] in welchen feine Feststoffe mit einer genau festgelegter "Körnung" enthalten sind.
  Es sind spezielle Poliermaschinen nötig, damit die Oberfläche des Chips eben bleibt. Durch den Einsatz dieser Maschinen ist es möglich in einem relativ zeitaufwändigen Prozess einzelne Lagen des Chips wegzuschleifen.

  Beim chemisch-mechanischem Polieren wird zusätzlich noch eine nasschemische Ätzlösung verwendet um den Prozess zu beschleunigen. Der Vorteil des Polierens liegt im gleichmäßigen Abtragen von Schichten, womit diese Methode besonders gut für optische Analysetechniken geeignet ist. Im Gegensatz zum nasschemischen Ätzen, bei welchem Materialien in Abhängigkeit der verwendeten Atzlösung angegriffen werden, ist es durch Polieren möglich alle auf einer Ebene liegenden Materialien gleichmäßig abzuschleifen.

### 2.2.4 Reverse Engineering (aktiv)

Die Aufgabe von Reverse Engineering besteht darin mehr Erkenntnis über den internen Aufbau eines Halbleiters sowie dessen Funktion zu erlangen. Beim einem ASIC[11] könnte dies etwa die Lage der Transistoren und deren Verbindungen sein. In Folge könnten alle Lagen des Chips schichtweise abgetragen und fotografiert werden, sodass die interne Struktur des Chips ersichtlich wird. Ist die entsprechende Vielzahl von hochauflösenden Bildern vorhanden, so können diese automatisiert durch Mustererkennung analysiert werden. Das Ziel dieses mühseligen und zeitaufwändigen Prozesses ist es schließlich eine standardisierte Netlist zu erstellen, mit welcher der Chip vollständig simuliert werden kann. Dies geht sogar so weit, daß daraus das gesamte Chipdesign in einer Hardware Beschreibungssprache wie VHDL[12] oder Verilog wiederhergestellt werden kann.
Unternehmen wie Chipworks[10] haben sich auf derartige Aufgaben spezialisiert und bieten diese im Rahmen ihrer Standard Services an.

Bei Mikrocontrollern oder Smartcards bedarf es meist nur einer kleineren strukturellen Analyse. Viel bedeutsamer ist hingegen das Erlangen und Reverse Engineering des Programmcodes (meist im Flash) sowie der gespeicherten Daten (oft im EEPROM). Die strukturelle Analyse dient dabei vorallem, um herauszufinden, in welcher Weise die Security Fuse implementiert wurde.

[8] RIE - reactive ion etching
[9] typischerweise mit einer Frequenz von 13.56 MHz
[10] auch "Slurry" genannt
[11] application specific integrated circuit
[12] Very High Speed Integrated Circuits hardware description language

Falls zusätzlich Bus Encryption verwendet wird, ist es notwendig die dafür zuständige Logik zu analysieren. Mit entsprechendem Aufwand sollte es danach möglich die internen Speicherinhalte auszulesen und den Programmcode zu disassemblieren[2].

**Zur Analyse** bis hin zu einer feature size von etwa $0.18\mu m$ kann man ein optisches Mikroskop mit einer hochauflösenden CCD Kamera verwenden. Dabei werden (etwa mit Hilfe eines motorisierten Präzisions X-Y Tisches) alle relevanten Bereiche eines Chip Layers fotografiert. Die entstehenden Bilder (vgl. Mosaike) werden anschließend zu einem Gesamtbild zusammengefügt. Da normales Licht nicht durch den Chip dringen kann, ist es nötig eine Auflichtbeleuchtung zu wählen. Desweiteren ist es wichtig hochauflösende Bilder ohne geometrische Verzerrungen oder Farbveränderungen zu gewinnen, da sonst die Bilder nicht (optimal) zusammensetzbar sind. Die höchstmögliche Auflösung (in der Größenordnung von $0.3\mu m$) kann bei sichtbarem Licht mit einem üblichen 100x Objektiv erzielt werden[2]. (Bei höherer Vergrösserung wird das Bild zunehmend unscharf und verzerrt.)
Verwendet man kürzere Wellenlängen (etwa im UV Nahbereich[13]), so lässt sich die Auflösung noch weiter steigern (etwa bis hin zu $0.18\mu m$) , jedoch wird damit auch der Einsatz von speziellen UV CCD Kameras notwendig.

Mikroskopie mit IR Licht ist dabei besonders, da Silizium für Licht in dieser Wellenlänge durchsichtig wird. Ein polierter Chip Die könnte daher etwa von der Rückseite untersucht[14] und in einem weiteren Arbeitsprozess evt. einzelne Transistoren durch die Substratschicht hindurch kontaktiert werden[2].

Leider sinkt mit der hohen Vergrößerung auch schnell die Arbeitsdistanz zwischen Objektiv und Probe. Als Konsequenz kann es unter Umständen nötig sein, daß der Chip Die zur Gänze aus dem Gehäuse entfernt werden muß.

Es gibt eine Vielzahl von unterschiedlichen optischen Mikroskoptypen, wie etwa konfokale Mikroskope, die mit der Höhe der Probe ihre Farbe verändern, sodass effektiv zwischen höher- und niederliegenden Schichten unterschieden werden kann.

Für Halbleiter Chips mit einer feature size von weniger als $0.13\mu m$ wird die Verwendung eines Elektronenmikroskopes (z.B. SEM) notwendig. So wäre die Auflösung eines SEM üblicherweise besser als 10 nm[2]. Auch mit modernen FIB Workstations können Bilder gemacht werden, sodass eventuell auf die Verwendung des SEM verzichtet werden kann.

Nähere Information zur Chip Layout Rekonstruktion sind in der Literatur[2][11] zu finden.

**Desweiteren** ist es möglich verschiedene Arten von Speicher (etwa Mask oder Laser ROM) optisch auszulesen. Dabei sind die Bits als Verbindungen in der Rom Matrix zu erkennen.

**Schutzmaßnahmen** vor derartigen Angriffen sind etwa die Planarisierung der Metallschichten, bei welcher freie Stellen in den Metallschichten mit Metall-Pads aufgefüllt werden und keine Sichtmöglichkeit mehr auf die unteren Schichten gegeben ist. Somit wird das Abtragen von entsprechend planarisierten Schichten unumgänglich. Auch die Abnahme der feature size erfordert aufwändigere Analysemethoden und erschwert damit das Reverse Engineering[2].
Vielversprechend ist auch die Verwendung von sog. *glue logic design*: Anstatt klar sichtbare Blöcke wie etwa EEPROM, SRAM, oder ALU zu verwenden, wird die gesamte Logik in einem einzigen Block zusammengefasst. Dies erlaubt oft nicht

[13] NUV - near ultravilot
[14] *backside imaging*

nur höhere Effizienz (etwa in Hinblick auf Stromverbrauch oder maximale Taktfrequenz), sondern erschwert auch deutlich den Reverse Engineering Prozess, da die Funktionalität eines jeweiligen Chip Bereiches nicht mehr unmittelbar ersichtlich ist[2].

### 2.2.5 Microprobing (aktiv)

Microprobing ist eine besonders wichtige Technik in der Klasse der invasiven Angriffe, die mit einer sog. *Microprobing Station* umgesetzt werden kann. Sie besteht üblicherweise aus einem optischen Mikroskop, einer präzise beweglichen Arbeitsplatte, einer Testfassung für den Chip, Mikromanipulatoren und aus den Prüfnadeln.

Um einen Halbleiter Baustein mit Microprobing analysieren zu können, muß zu Beginn die isolierende Passivierungsschicht entfernt werden, sodass auf die metallischen Leiterbahnen zugegriffen werden kann. Der Chip wird nun in die Testfassung gesteckt und beispielsweise von einem Computer angesteuert. Die Fassung befindet sich dabei auf dem beweglichen Arbeitstisch unter dem Mikroskop. Schließlich werden die Prüfnadeln mit Hilfe der Mikromanipulatoren auf "interessante" Bereiche des Chips gesetzt, damit die Signale auf den Leiterbahnen etwa mit einem Oszilloskop oder einem Logikanalysator abgegriffen werden können. Die Mikromanipulatoren haben eine Präzision im unteren $\mu m$ Bereich.[2]

Da die Nadeln so dünn sein müssen, sind sie üblicherweise aus Wolfram hergestellt. Ihre Spitze ist dünner als $0.1\mu m$ damit auch die kleinsten Strukturen kontaktierbar sind.

Soll auf tiefere Schichten zugegriffen werden, so ist es unter Umständen notwendig den Chip mit einer FIB Workstation oder einem Lasercutter zu manipulieren. Mit derartigen Techniken ist es manchmal sogar möglich, Schutzmaßnahmen wie ein Sensor-Gitter auf der Oberseite des Chips zu überwinden. So konnte etwa in der Smartcard ST16SF48A vorsichtig mit einem FIB[15] ein kleines Loch in das Sensor-Gitter gebohrt werden. Das Loch wurde mit Metall aufgefüllt und oberhalb des Sensor-Gitters ein Testpad geformt, welches wiederum mit einer Microprobe Nadel erreichbar ist[6].

Auch das vorsichtige Kratzen mit der Prüfnadel kann eventuell angewendet werden um beispielsweise Leiterbahnen auf der Oberseite des Chips zu durchtrennen[2].

### 2.2.6 Lasercutter (aktiv)

Ein präziser Laser zum Schneiden auf Chip Layern kann beispielsweise sehr praktisch zum stellenweisen Entfernen der Passivierungsschicht sein. Wird sie nur stellenweise entfernt (im Gegensatz zum chemischen Entfernen), so entstehen an den entsprechenden Stellen Vertiefungen in der Passivierungsschicht, die das Justieren von Microprobing Nadeln erleichtern (die Nadelspitze "rastet" mehr oder weniger in die Vertiefung ein).

Um auf derart kleinen Strukturen arbeiten zu können, wird der Laserkopf an der Kameraöffnung eines Mikroskops angebracht, wobei die Optik des Mikroskops für den Laserstrahl geeignet sein muss (etwa metallurgische Mikroskope mit Objektiven für den NUV oder NIR[16] Bereich). Mit einem UV Laser lassen sich besonders gut Polyimid[17] oder andere organische Materialien, die häufig auf der Passivierungsschicht vorhanden sind, entfernen.

Um Metallschichten zu schneiden, ist der Einsatz von grünen oder IR Lasern möglich. Da diese Systeme nicht weit verbreitet und mit Preisen von mehr als 50.000 € sehr teuer sind, muß ein An-

[15] *FIB editing*
[16] near infrared
[17] Hochleistungskunststoff

greifer evt. auf die bereits vorgestellten nasschemischen Methoden zurückgreifen.

### 2.2.7 FIB Workstation (aktiv)

Eine FIB Workstation ist im Aufbau dem Elektronenmikroskop (z.B. SEM) ähnlich, jedoch werden mit einstellbarer Beschleunigungsspannung nicht Elektronen, sondern Ionen (üblicherweise Gallium-Ionen) auf die Probe geschossen. Bei geringer Beschleunigungsspannung verursachen die Gallium Ionen noch keine Veränderung auf die Chipoberfläche, jedoch können die Sekundärpartikel von der FIB Workstation verwendet werden, um ähnlich einem SEM ein Bild von der Oberfläche bis hin zu einer Auflösung von etwa 10 nm zu produzieren. Steigert man die Intensität des Ionenstrahls, so wird Chipmaterial von der Oberfläche des Chips mit derselben Präsision herausgeschlagen[2]. Leitet man zusätzlich etwa noch ein Platin hältiges Gas in die Vakuumkammer, so kann gezielt Platin (oder bei einem anderem Gas auch andere Metalle) am Chip abgeschieden werden. Sogar das Abscheiden von Isolatoren ist möglich.

Zusammenfassend ist eine FIB Workstation eines der hilfreichsten Werkzeuge für invasive Angriffe auf Halbleiter, da nicht nur mit hoher Präzision geschnitten und Materialien abgeschieden werden können (sogenanntes *FIB editing*), sondern auch die Erzeugung von Bildern vergleichbar mit einem SEM möglich ist.

Mit derartigen Möglichkeiten is es manchmal bei Mikrocontrollern möglich, Schutzmaßnahmen lediglich durch Unterbrechen von wenigen Leitungen oder durch Zerstören von sicherheitsrelevanten Schaltungsteilen zu umgehen[2].

## 2.3 semi-invasive Angriffe

Zwar ist die Klassifizierung der semi-invasiven Angriffe noch relativ neu, jedoch sind einige Angriffe, die in diesen Bereich fallen, bereits seit vielen Jahren bekannt. Wie auch bei den invasiven Angriffen, muß erst das Gehäuse des ICs geöffnet werden, jedoch bleibt die Passivierungsschicht des Halbleiters unberührt. Neben der beispielsweise optischen Halbleiteranalyse kann nun auch Lichteinwirkung oder Strahlung verwendet werden um Veränderungen im Zustand des Bausteins zu verursachen. So könnten etwa die Sicherheitsbits der security-fuse(s) eines Mikrocontrollers rückgesetzt werden, sodass der zuvor geschützte Programmcode auslesbar wird.

### 2.3.1 optische Angriffe mit UV Licht (aktiv)

Die Angriffsmethode ist schon seit den Siebzigern bekannt und wird vorallem zum Löschen von Speicherzellen (wie etwa der Security Fuse) verwendet. So beruht etwa auch der heute kaum mehr eingesetzte UV EPROM Speichertyp auf der Verwendung von UV Licht zum Löschen der Speichers. (Zu diesem Zweck war in das Gehäuse des Speicherbausteins ein Quarzglas eingearbeitet, damit der Speicher unter minutenlanger Bestrahlung mit UV Licht gelöscht werden konnte.)

Heute sind Security Fuses mittlerweile derart implementiert, daß bei der UV Bestrahlung des gesamten Chips zuerst der Speicherinhalt verloren geht und erst danach die Security Fuse zurückgesetzt wird. Aus diesem Grund müssten entsprechende Speicherbereiche abgedeckt werden, damit der Speicherinhalt nicht durch das UV Licht verschwindet[2]. Auch der Einsatz eines UV Lasers ist dafür denkbar.

**Die Security Fuse** kann sich komplett getrennt beziehungsweise auch auf derselben Fläche wie

die Speicherzellen des Chips befinden. Auch eine direkt in die Speicherzellen eingebettete Security Fuse ist möglich. Damit ist es auch oft eine nicht triviale Aufgabe, die Position der Security Fuse auf dem Chip zu finden.

Die wohl aufwändigste Methode ist Reverse Engineering um den Baustein gänzlich oder zumindest zum Teil zu analysieren. Man könnte etwa die Leiterbahn zur Verfügungsstellung der hohen Programmierspannung verfolgen, da diese auch zu allen EEPROM Speicherzellen und Fuses führen muß[2].

Liegt die Security Fuse sehr nahe bzw. sogar eingebettet im Speicher, so ist das Auffinden der Position deutlich schwieriger.

Eine einfache Technik ist die binäre Suche nach der Security Fuse mit UV Licht. Dazu wird der Chip zuerst vollständig programmiert (es wird somit der gesamte Speicher vollgeschrieben) und mit der Security Fuse geschützt. Anschließend wird entsprechend der binären Suche immer eine Hälfte des Chips mit einem UV undurchlässigem Material abgedeckt (etwa Isolierband). Der Chip wird nun in ein UV Löschgerät gelegt damit die ungeschützen Speicherbereiche gelöscht werden. Wird der Chip anschließend ausgelesen, kann festgestellt werden, welche Speicherbereiche (und Fuses) von dem Löschvorgang betroffen waren. Eine weitere Wiederholung dieses Vorganges führt zur schrittweisen genaueren Lokalisation von gesuchten Speicherbereichen wie etwa der Security Fuse[2].

Als Alternative zum Klebeband ist auch ein UV beständiger Textmarker[18] verwendbar. Dazu wird zuerst der gesamte Speicherbereich mit Farbe überzogen. Anschließend können die zu analysierenden Stellen entweder sehr genau mit einem Laser Cutter oder aber auch mit der Spitze eines einfachen Zahnstochers entfernt werden[12][2].

[18] etwa ein “Edding” Stift

Schutzmaßnahmen gegen derartige UV Angriffe sind heute häufig in Mikrocontrollern zu finden. Sie reichen etwa von einer UV undurchlässigen Metallummantelung der Security Fuse bis hin zu UV Sensoren oder speziellen Speicherzellen, die unter UV Licht nicht gelöscht, sondern stattdessen gesetzt werden[2].
Die Security Fuse kann zudem im Speicher “versteckt” werden um die Zeit zum Auffinden zu maximieren. Mikocontroller wie der Philips 87C51 Controller verschlüsseln den Speicherinhalt, damit selbst nach kompromitierter Security Fuse kein Klartext ausgelesen werden kann. Da die Verschlüsselung in diesem Beispiel jedoch lediglich aus einem Encryption-Table und einer XNOR Operation besteht, könnte ein Angreifer den Encryption-Table Rekonstruieren und die Daten entschlüsseln[2].

**Auch EEPROM- und Flash-** Speicherzellen sind aufgrund des floating Gates für UV Licht sensibel. Je nach Implementation der Speicherzelle, kann UV Lichen den Zustand der Zelle entweder von “programmiert” zu “gelöscht” bzw. umgekehrt verändern. Sollte der Zustand, der mittels UV Bestrahlung erreicht werden kann, dem Zustand einer deaktivierten Security Fuse entsprechen, so wäre mit einem derartigen Angriff auch die Sicherheit des Controllers kompromitierbar. Je nach Zelle können auch noch andere Zustände (etwa Zwischenzustände) erreicht werden, die unter Umständen sicherheitsrelevant sein können[2].

### 2.3.2 Backside Imaging (aktiv)

Optische Analyse ist üblicherweise der erste Schritt bei der semi-invasiven Analyse. Durch die sinkende feature size werden die Strukturen jedoch immer kleiner, sodass der Einsatz von nichtoptischer Mikroskopie wie Elektronenmikroskopie nötig wird. Hinzu kommt die Verwendung von Planarisierungstechnologien (siehe Kapitel *Reverse Engineering*) die ein Hindurchblicken durch

obere Schichten deutlich erschweren.
Da jedoch Silizium für IR Licht mit Wellenlängen $> 1100nm$ nahezu durchlässig wird, ist es möglich den Chip von der Rückseite zu analysieren (*Backside Imaging*).
Um IR Aufnahmen anzufertigen sind klarerweise entsprechende IR sensible Kameratypen notwendig.
Mit Backside Imaging können gleichsam zur normalen optischen Analyse nicht nur Chipstrukturen analysiert, sondern auch Speicherinhalte (etwa aus dem MaskROM) ausgelesen werden[2].

### 2.3.3 Photonen Probing (aktiv)

Halbleiter Transistoren sind anfällig für ionisierende Strahlung, egal ob diese Strahlung etwa von einer nuklearen Explosion, radioaktiven Isotopen, Röntgen- oder kosmischer Strahlung stammt. Neben den genannten Strahlungstypen hat jedoch auch koheräntes Licht wie Laserstrahlung eine nahezu idente Auswirkung auf Halbleiter. Es wird aus diesem Grund in der Halbleiterindustrie auch eingesetzt um Strahlungseffekte auf Halbleiter zu simulieren[2]. Dabei muß die Energie der Photonen größer als das Band Gap des Halbleiters sein, damit die IC Regionen entsprechend ionisiert werden.
Experimente haben ergeben, daß sich Laser Strahlung mit einer Wellenlänge von 1060nm (mit 1.17 eV Photonenenergie) mit einer Eindringtiefe von etwa 700 $\mu m$ gut für diese Zwecke eignet[2]. Unglücklicherweise ist das Licht bei dieser Wellenlänge nur auf einen Bereich von mehreren Mikrometern fokusierbar und daher für aktuellere Halbleiter Bausteine mit kleiner feature size schlecht geeignet. Mit kürzeren Wellenlängen wird andrerseits die Absorption deutlich größer. Aus diesem Grund ist es zwar möglich Licht im sichtbaren Bereich zu verwenden (etwa rote oder grüne Laser) um damit die Fokusierbarkeit zu erhöhen, jedoch steigt aufgrund der höheren Absorption auch die Gefahr von permanenter Schädigung beispielsweise durch Latch-Ups[19][2].

Beim aktiven Photonen Probing verwendet man beispielsweise Laser Scanning Mikroskope, die schrittweise kleine Bereiche des Chips bestrahlen. Aufgrund der hohen Kosten kann als Alternative aber auch ein einfacher Laserpointer am Kamera Anschluß eines optischen Mikroskops in Verbindung mit einem motorisierten X-Y Tisch verwendet werden. Im Vergleich zur professionellen Umsetzung ist jedoch bei dieser Methode die Scan Geschwindigkeit bedeutend niedriger.

**Bei der OBIC** (*optical beam induced current*) Technik wird ein nicht unter Spannung stehender Chip untersucht. Dabei verwendet man eine *current amplifier* Schaltung um geringe Ströme am Stromversorgungspin messen zu können. Aufgrund des photovoltaischen Effekts entstehen beim Auftrennen von Photonen in den Transistoren geringe Ströme, die am Stromversorgungspin messbar sind. An diesem Pin wird nun für jeden bestrahlten Punkt im Scan-Raster eine Messung vorgenommen. Trifft der Strahl auf einen Schaltungsteil mit aktiven Elementen (wie etwa einem Transistor), so ist der gemessene Strom höher als in den rein passiven Bereichen. Man bekommt bei der OBIC Technik als Ergebnis somit ein *Sensivity Image* welches in Verbindung mit einer herkömmlichen optischen Aufnahme verwendet werden kann um Bereiche mit aktiver Logik zu finden (etwa die Security Fuse Logik)[2]. Entsprechend der Technik des Backside Imaging kann Laser Scanning mit Wellenlängen im IR Bereich auch auf der Chip-Rückseite verwendet werden. Dies hat insbesonders Vorteile, da die Metall Layer auf der Oberseite des Chip Dies dem Laserstrahl nicht im Weg sind und auf diese Weise ein genaueres Bild von der Logik angefertigt werden kann.

[19] fehlerhafter "Kurzschluss" im Transistor

**Die LIVA** (*light-induced voltage alteration*) Technik funktioniert ähnlich der OBIC Technik, jedoch wird der Chip unter Spannung analysiert. Trifft der Laserstrahl auf Silizium Strukturen, so werden Paare von Elektronen Löchern generiert und es ändert sich der Strom und somit auch die Spannung am Stromversorgungspin des ICs[13][2]. Das Bild wird schließlich in Abhängigkeit der Spannungs Änderungen erstellt.

Interessant ist, daß mit den vorgestellten Techniken etwa MaskROMs (z.B. im MC68HC705P6A Controller) auch semi-invasiv ausgelesen werden können. Dabei ist die Sensibilität gegenüber dem Laserstrahl im MaskROM bei einer '1' anders als bei einer '0'[2].

Ein ähnliches Verfahren ist die Verwendung von diversen *Electron beam induced current (EBIC)* Techniken, die anstatt eines Lasers einen viel genauer fokusierbaren Elektronenstrahl meist in Kombination mit Elektronenmikroskopie (SEM) verwenden um vergleichbare Bilder zu generieren[14].

**Das Bestimmen von Logikzuständen** von CMOS Transistoren ist traditionellerweise nur durch invasive Angriffe umsetzbar, indem etwa durch Microprobing der Bus eines Prozessors abgegriffen wird. Mit den vorgestellten Laser Scanning Techniken ist es jedoch ebenfalls möglich den Zustand von SRAM Zellen semi-invasiv auszulesen. Trifft ein Laserstrahl auf einen p-n Übergang, so wird gemäß dem photovoltaischen Effekt Spannung erzeugt. Treffen die Photonen hingegen auf einen p- oder n-Kanal Bereich, so nimmt lediglich der Widerstand des Kanals ab. Ein geschlossener Kanal produziert nun einen Stromanstieg während der Effekt bei einem offenen Kanal vernachlässigbar ist. Durch die Messung des Stromflusses kann daher der Zustand einer SRAM Zelle bestimmt werden[2].

Sind sensible Daten auch nur für einen einzigen Taktzyklus im Klartext im SRAM vorhanden und ein Angreifer kann nicht nur die physische Position der Daten finden, sondern diese auch einfrieren (etwa durch Absenken der Betriebstempeartur oder durch Stoppen des Taktsignals), so ist es auch sehr wahrscheinlich, daß der Angreifer die Daten durch optische oder elektro-magnetische Angriffe auslesen kann[2].

### 2.3.4 Fault Injection Angriffe (aktiv)

Wird ein Transistor mit ausreichender Leistung angeleuchtet (etwa durch einen Laser Pointer oder durch den Blitz einer Photokamera), so führt dies zum Durchschalten des Transistors. Aus diesem Grund können absichtlich Fehler in die laufende Schaltung eingeschleust werden (*fault injection*). Die Macht dieser Angriffsmethoden wird deutlich, indem etwa einzelne Bits im SRAM eines Controllers nach Belieben gesetzt und gelöscht werden können. Dies kann etwa verwendet werden um Sicherheitschecks zu Umgehen oder den Programmablauf eines Prozessors zu ändern, sodass beispielsweise kryptografische Berechnungen frühzeitig abgebrochen und sensible Informationen über Daten oder den geheimen Schlüssel preisgegeben werden[2][6].

**Bei SRAM Zellen** werden jeweils zwei Paare von p- und n-Kanal Transistoren verwendet[20] um ein zustandshaltendes Flip-Flop zu formen. Zwei weitere Transistoren sind notwendig um einen neuen Zustand einzulesen bzw. am Ausgang auszugeben. Der Zustand der Zelle kann nun nach belieben verändert werden, indem die externe Lichteinwirkung kurzzeitig zum Durchschalten des jeweiligen Inverters und somit zur Zustandsänderung des gesamten Flip-Flops verwendet wird[2]. Da die Inverter physisch einen Abstand zueinander haben,

[20] jeweils ein CMOS Inverter

kann punktuell auf den gewünschten Bereich gezielt werden.

**Bei nicht volatilen Speichern** wie EPROM-, EEPROM- oder Flash-Speicherzellen ist die Anfälligkeit gegenüber optischen Fault Injection Angriffen noch höher als bei SRAM Zellen, da der Zustand der Speicherzelle in der Ladung des floating Gate gespeichert wird. Diese Ladungen sind bei Weitem geringer als etwa innerhalb einer SRAM Zelle, wodurch schon eine wesentlich kleinere (optische) Störung zur Änderung des gespeicherten Zustands führt. Desweiteren können derartige Angriffe ebenfalls von der Rückseite des ICs durchgeführt werden, jedoch muss dann auch der IR Anteil des verwendeten Lichts entsprechend hoch sein[2].
Mit zunehmender Miniaturisierung wird der Angriff allerdings schwieriger weil die Wellenlänge des Lichts so groß ist, daß Probleme bei der präzisen Fokusierung auftreten bzw. kleinere Transistoren auch höhere Strahlungsdosen benötigen um ihren Zustand zu verändern bzw. durchzuschalten. Eventuell wäre hier ein Zurückgreifen auf EBIC Techniken denkbar.

Weitere Gegenmaßnahmen sind etwa die Verwendung von Licht- oder Strahlungssensoren, die den Controller durch einen Reset zurücksetzen, bevor sensible Informationen preisgegeben werden können.

# 3 Schlußwort und Ausblick

In der vorliegenden Arbeit wurden sowohl verfügbare Schutzmaßnahmen sowie ein breiter Bereich von Angriffsmethoden auf Halbleiter vorgestellt. Obwohl einige der vorgestellen Methoden schon lange bekannt sind, treten auch diese alte Methoden häufig im Rahmen von neueren Techniken wieder auf (etwa bei optischen Angriffen die besonders in der Klasse der semi-invasiven Angriffs- und Analysemethoden vielversprechend sind). Sowohl aus Kostengründen wie auch aus Unachtsamkeit können (vergleichbar mit der Entwicklung von Software) auch bei der Chipentwicklung Sicherheitslücken und -risiken entstehen. Ziel der Arbeit war es einen Einblick in die möglichen Methoden zum Auffinden bzw. zur Analyse von derartigen Schwachstellen zu ermöglichen. Leider scheint viel Forschung in diesem Bereich hinter den geschlossenen Türen von Halbleiterherstellern oder spezialisierten Analyseunternehmen wie Chipworks[10] oder Flylogic Engineering[15] statt zu finden, womit öffentliche Forschung in diesem hochinteressanten Bereich erschwert wird. Insbesondere in Bezug auf die steigende Verwendung von Smartcards und Mikrocontrollern in sicherheitsrelevanten Applikationen scheint die Sicherheitsanalyse zunehmende Relevanz zu bekommen. So wurde etwa erst unlängst die Sicherheit der weitverbreiteten Mifare Classic Karten kompromittiert, indem durch optische Analyse der zuvor geheime proprietäre Verschlüsselungsalgorithmus aufgedeckt wurde[16].
Weitere aktuelle Beispiele sind das Brechen des KeeLoq Zutrittsystems mittles differentieller Power-Analyse (DPA)[17] oder das Aufdecken der zuvor geheimen DECT Algorithmen duch Chip Reverse Engineering[18].

# Literatur


[1] BugTraq Archive,
`http://www.securityfocus.com/archive/1`

[2] Sergei P. Skorobogatov,
Semi-invasive attacks - A new approach to hardware security analysis,
Technical Report Number 630,
April 2005,
University of Cambridge, Computer Laboratory

[3] Paul C. Kocher,
Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and other Systems,
Advances in Cryptology, CRYPTO96, LNCS, Vol. 1109, Springer-Verlag, 1996, pp. 104-113,
http://www.cryptography.com/resources/whitepapers/TimingAttacks.pdf

[4] Engineering Technical Laboratory,
Security protection in Motorola Microcontrollers,
http://www.etlweb.com/articles_secprotect.html

[5] Maxim iButton products,
http://www.maxim-ic.com/products/ibutton/products/ibuttons.cfm

[6] Oliver Kömmerling, Markus G. Kuhn,
Design Principles for Tamper-Resistant Smartcard,
Processors, USENIX Workshop on Smartcard Technology, Chicago, Illinois, USA, May 10-11, 1999

[7] David Samyde, Sergei Skorobogatov, Ross Anderson, Jean-Jacques Quisquater,
On a New Way to Read Data from Memory,
SISW2002 First International IEEE Security in Storage Workshop

[8] Sergei Skorobogatov,
Low Temperature Data Remanence in Static RAM,
University of Cambridge Computer Laboratory

[9] Virginia Semiconductor, Inc.,
Wet-Chemical Etching and Cleaning of Silicon,
http://www.virginiasemi.com/pdf/siliconetchingandcleaning.pdf

[10] Chipworks Semiconductor Manufacturing:
Reverse Engineering of Semiconductor components, parts and processes for Semiconductor Distributors,
http://http://www.chipworks.com

[11] Simon Blythe, Beatrice Fraboni, Sanjay Lall, Haroon Ahmed and Ugo de Riu,
Layout Reconstruction of Complex Silicon Chip,
Cavendish Lab., Cambridge University

[12] Jesse Jenkins,
CoolRunner-II CPLDs in Secure Applications,
Xilinx White Paper WP170,
http://www.xilinx.com/bvdocs/whitepapers/wp170.pdf

[13] Wikipedia article on Light induced voltage alteration,
http://en.wikipedia.org/wiki/Light_induced_voltage_alteration

[14] Gatan, Inc.
An Introduction to EBIC,
http://www.gatan.com/files/PDF/products/Introduction_to_EBIC.pdf

[15] Flylogic Engineering, LLC.
http://www.flylogic.net

[16] Security of MIFARE Classic,
NXP Semiconductors Austria GmbH Styria,
http://mifare.net/security/
mifare_classic.asp

[17] Thomas Eisenbarth1, Timo Kasper1, Amir Moradi2, Christof Paar1 , et al.
Physical Cryptanalysis of KeeLoq Code Hopping Applications,
Horst Görtz Institute for IT Security, Ruhr University of Bochum, Germany et al.

[18] Stefan Lucks, Andreas Schuler, Erik Tews, Ralf-Philipp Weinmann, Matthias Wenzel,
Attacks on the DECT authentication mechanisms,
Bauhaus-University Weimar, Germany et al.