Data Remanence in Flash Memory Devices

Sergei Skorobogatov

University of Cambridge, Computer Laboratory,
15 JJ Thomson Avenue, Cambridge CB3 0FD, United Kingdom
sps32@cl.cam.ac.uk

Abstract. Data remanence is the residual physical representation of data that has been erased or overwritten. In non-volatile programmable devices, such as UV EPROM, EEPROM or Flash, bits are stored as charge in the floating gate of a transistor. After each erase operation, some of this charge remains. Security protection in microcontrollers and smartcards with EEPROM/Flash memories is based on the assumption that information from the memory disappears completely after erasing. While microcontroller manufacturers successfully hardened already their designs against a range of attacks, they still have a common problem with data remanence in floating-gate transistors. Even after an erase operation, the transistor does not return fully to its initial state, thereby allowing the attacker to distinguish between previously programmed and not programmed transistors, and thus restore information from erased memory. The research in this direction is summarised here and it is shown how much information can be extracted from some microcontrollers after their memory has been 'erased'.

1 Introduction

Data remanence as a problem was first discovered in magnetic media [1,2]. Even if the information is overwritten several times on disks and tapes, it can still be possible to extract the initial data. This led to the development of special methods for reliably removing confidential information from magnetic media.

Semiconductor memory in security modules was found to have similar problems with reliable data deletion [3,4].

Data remanence affects not only SRAM, but also memory types like DRAM, UV EPROM, EEPROM and Flash [5]. As a result, there is possibility that some information still can be extracted from memory that has been erased. This could create problems with secure devices where designers assumed that all sensitive information is gone once the memory is erased.

In some smartcards and microcontrollers, a password-protected boot-loader restricts firmware updates and data access to authorised users only. Usually, the on-chip operating system erases both code and data memory before uploading new code, thus preventing any new application from accessing previously stored secrets. If the passwords or secret keys can be extracted afterwards, it could create serious problems for confidentiality of the previously encrypted information.

Chip manufacturers do not publish data about remanence effects for their memory chips. The only parameter they specify is data retention – the time during which the memory content is preserved. This is almost the opposite of data remanence. Data retention time can be used roughly to estimate the data remanence effect, but this works only for devices within the same family [4].

Therefore, a series of experiments was performed to check whether it is feasible to extract information from erased EPROM, EEPROM and Flash memory devices using low-cost methods. The results should be of considerable concern to designers of embedded security applications.

2 Background

Unlike SRAM, which has only two stable logic states, EPROM, EEPROM and Flash cells store analog values in the form of a charge on the floating gate of a MOS transistor. The floating-gate charge shifts the threshold voltage of the cell transistor and this is detected with a sense amplifier when the cell is read. The maximum charge the floating gate can accumulate varies from one technology to another and normally is between 10^3 and 10^5 electrons. For standard 5 V EEPROM cells, programming causes about a 3.5 V shift in the threshold level. Some modern Flash memory devices employ multiple level detection, thus increasing the capacity of the memory [6]. There are also memory devices with a fully analog design, which store charges proportional to the input voltage [7].

	Static RAM	Mask ROM	OTP EPROM	UV EPROM	EEPROM	Flash EEPROM	NVRAM
Read time	FAST $\approx$ 10 ns	FAST $\approx$ 5 ns	MED $\approx$ 50 ns	MED $\approx$ 50 ns	MED $\approx$ 50 ns	FAST $\approx$ 20 ns	MED $\approx$ 50 ns
Write time	FAST $\approx$ 10 ns	N/A	SLOW $\approx$ 10 ms	SLOW $\approx$ 10 ms	SLOW $\approx$ 1 ms	MED $\approx 10\ \mu$s	FAST $\approx$ 50 ns
Data retention	$>$ 5 years (battery)	N/A	$>$ 10 years	$>$ 10 years	$>$ 40 years	$>$ 100 years	$>$ 40 years
Cell size	6T	1T	1T	1T	2T	1T	10T
Low voltage	Yes	Yes	No	No	No	No	No
Endurance (cycles)	N/A	N/A	1	100	10^3–10^6	10^4–10^6	N/A
Cost	HIGH	LOW	MED	HIGH	MED	LOW	HIGH

Table 1. Characteristics of different memory types used in microcontrollers

There are two basic processes that allow placing electrons on the floating gate – Fowler-Nordheim tunnelling and channel hot electron (CHE) injection [8]. Both processes are destructive to the very thin dielectric insulation layer between the floating gate and the channel of a transistor. This oxide layer is responsible for preserving the charge on the floating gate. As a result, the number of possible write/erase cycles is limited, because the floating gate slowly accumulates electrons, causing a gradual increase in the storage transistor's threshold voltage and programming time. After a certain amount of program/erase cycles (typical values are represented in Table 1) it is no longer possible to erase or program the cell. Another negative effect (which is the main failure mode for Flash memory) is negative charge trapping in the gate oxide. It inhibits CHE injection and tunnelling, changes the write and erase times of the cell, and shifts its threshold voltage.

The amount of trapped charge can be detected by measuring the gate-induced drain leakage current of the cell, or its effect can be observed indirectly by measuring the threshold voltage of the cell. In older devices, which had the reference voltage for the sense amplifier tied to the device supply voltage, it was often possible to do this by varying the device supply voltage. In newer devices, it is necessary to change the parameters of the reference cell used in the read process, either by re-wiring portions of the cell circuitry or by using undocumented test modes built into the device by manufacturers.

Another relevant phenomenon is overerasing. If the erase cycle is applied to an already-erased cell, it leaves the floating gate positively charged, thus turning the memory transistor into a depletion-mode transistor. To avoid this problem, some devices, for example Intel's original ETOX [9], first program all cells to 0 before erasing them to 1. In later devices, this problem was solved by redesigning the cell to avoid excessive overerasing. However, even with this protection, there is still a noticeable threshold shift when a virgin cell is programmed and erased.

The changes in the cell threshold voltage caused by write/erase cycles are particularly apparent in virgin and freshly-programmed cells. It is possible to differentiate between programmed-and-erased and never-programmed cells, especially if the cells have only been programmed and erased once, since virgin cell characteristics will differ from the erased cell characteristics. The changes become less noticeable after ten program/erase cycles.

Programmed floating-gate memories cannot store information forever. Various processes (such as field-assisted electron emission and ionic contamination) cause the floating gate to lose the charge, and these go faster at higher temperatures. Another failure mode in the very thin tunnel oxides used in Flash memories is programming disturb, where unselected erased cells adjacent to selected cells gain charge when the selected cell is written. This is not enough to change the cell threshold sufficiently to upset a normal read operation, but could cause problems to the data retention time and should be considered during measurement of the threshold voltage of the cells for data analysis and information recovery. Typical guaranteed data retention time for EPROM, EEPROM and Flash memories are 10, 40 and 100 years, respectively.

3 Experimental Method

Obviously, in a floating gate memory cell, the floating gate itself cannot be accessed. Its voltage is controlled through capacitive coupling with the external nodes of the device. Often, the floating-gate transistor is modelled by a capacitor equivalent circuit called the capacitor model [10]. In practice, write/erase characteristics for many EEPROM/Flash memories are close to that of a charge/ discharge of a capacitor. Meanwhile there are some differences in how the charge/ discharge process takes place in real memory cells. There is an initial delay between the time the voltages are applied to the cell, and the charge starting to be removed or injected. This delay is caused by the need for very high electric fields to be created inside the floating-gate transistor to start the injection or tunnelling process. Some EEPROM cells have been reported to have nonuniformity during the erase operation [11]. As a result, it might take longer to erase a half-charged cell than a fully-charged cell. In addition, an ideal capacitor discharges exponentially: $q = q_0 \cdot \mathrm{e}^{-t/\tau}$. Applied to the floating gate, that would mean that after $t = 10\tau$ the charge is totally removed from the cell. In practice this does not happen, because the parameters of the cell's transistor change as the charge is removed from its floating gate. All the above-mentioned problems could seriously affect data remanence in floating-gate memories.

The main difficulty with analysis of the floating-gate memory devices, especially EEPROM and Flash, is the variety of different designs and implementations from many semiconductor manufacturers. There are hundreds of different types of floating-gate transistor, each with its own characteristics and peculiarities. It means that for security applications where data remanence could cause problems, careful testing should be applied to the specific non-volatile memory device used in the system.

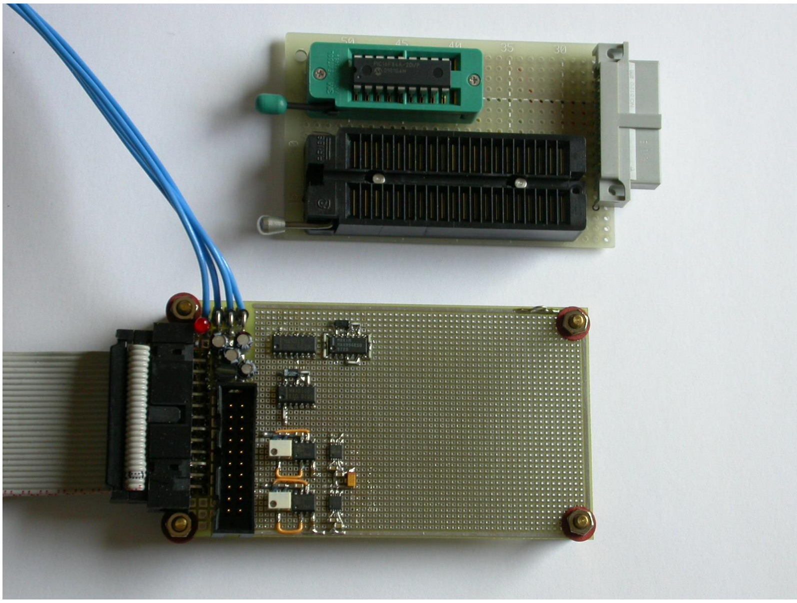

Fig. 1. The test board for data remanence evaluation

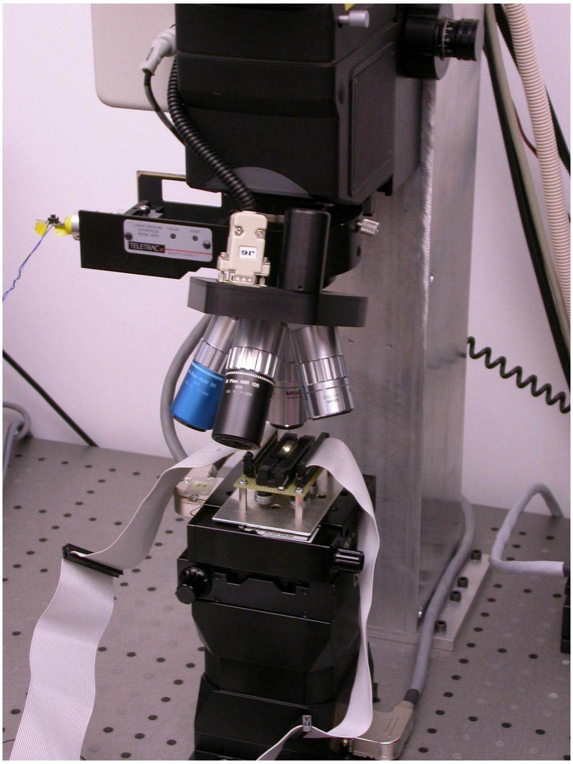

Fig. 2. Test setup for semi-invasive analysis

Some microcontrollers with different memory types to investigate the possible influence of data remanence on EPROM, EEPROM and Flash memories were tested. For that purpose I built a special test board controlled by a PC via a parallel interface (Figure 1). The board has two programmable power supplies for generating V_{DD} and V_{PP} voltages, a programming interface with bidirectional voltage level converters, and sockets for microcontroller chips. That allowed me to control the voltages applied to the chip under test with 100 μV precision and apply any signals within a 1 μs time frame.

Recently introduced semi-invasive attack methods [15] might also be helpful for testing data remanence effect in floating-gate memory devices. These methods are more effective in some applications as they do not require physical access to the internal wires inside the chip thus reducing the preparation time. However, partial or full decapsulation of the sample is required [16]. For such analysis, a low cost laser diode pointer with external power control was mounted on the autofocus module optical port of a Mitutoyo FS60Y microscope. Computer controlled Newport PM500-XYZ motorised stage was used for moving the sample under test (Figure 2). Using 100× objective on the microscope it was possible to focus the red laser beam (650 nm) down to 0.5 μm (Figure 3). Although the

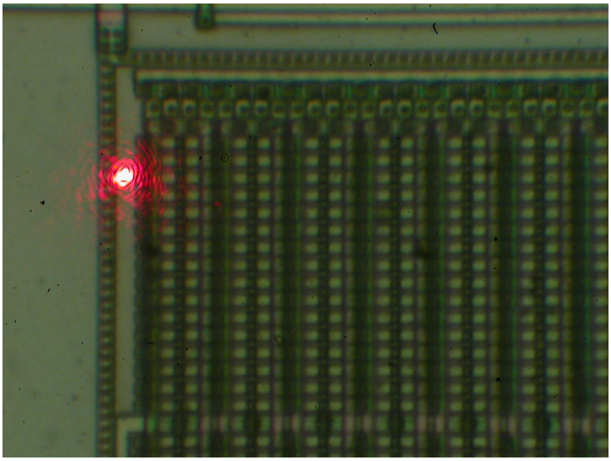

Fig. 3. Focusing the laser with a 100× objective

laser used was classified as a class 2M laser device, an ordinary digital camera mounted on the microscope was used for navigation to avoid necessity of looking at the laser beam with unprotected eyes.

4 Non-Invasive Results

The first experiment was performed on the Microchip PIC12C509 microcontroller [12] with UV EPROM. The chip was programmed with all 0's (charged cell state) and exposed to UV light for different periods of time. Then it was read in the test board at different power supply voltages to estimate the threshold level for each EPROM cell in the memory array. The reference voltage was assumed to be tied to the power supply line and therefore the threshold level of the transistor is proportional to the power supply voltage $V_{\mathrm{TH}} = K \cdot V_{\mathrm{DD}}$. The fact that the exact threshold voltage of the transistor is not measured does not affect the results because an attacker is normally interested in the relative erase timing between the memory and the security protection. Once the security fuse is erased, the memory can be easily read. The same test was applied to a chip with a programmed security fuse. The results are presented in Figure 4. As can be seen from the graph, the memory gets fully erased before the security fuse is erased. However some security flaws still could exist. Although nothing could be extracted directly by reading the memory when the fuse is erased, power glitch tricks could work. For example, after seven minutes of exposure to the UV light (253 nm peak, 12 mW/cm^2) the memory content can be read non-corrupted at V_{DD} below 2.2 V, but the security fuse remains active up to 4.8 V. If the attacker works out the exact time when the data from memory is latched into the output shift register and the time when the state of the security fuse is checked, he might be able to extract the memory contents by reducing the power supply down to 2 V for the data latching and increasing it to 5 V to make the security fuse inactive.

There is another trick that makes recovery of memory contents possible, even when there is no overlap between the erased security fuse and non-corrupted memory content at the time of erasure. For example, I found that newer samples of the same chip will start to corrupt the memory before the security fuse is erased (Figure 4). In this case a power glitch cannot be used to recover information from the memory. What can be done instead is a careful adjustment of the threshold voltage in the cell's transistor. It is possible to inject a certain portion of charge into the floating gate by carefully controlling the memory programming time. Normally, the programming of an EPROM memory is controlled by external signals and all the timings should be supplied by a programmer unit. This gives an opportunity for the attacker to inject charge into the floating gate thus shifting the threshold level enough to read the memory contents when the security fuse is inactive. Such a trick is virtually impossible to apply to modern EEPROM and Flash memory devices for several reasons. Firstly, the programming is fully controlled by the on-chip hardware circuit. Secondly, the programming of EEPROM and Flash cells is normally performed by using much faster Fowler-Nordheim tunnelling rather than CHE injection. As a result it is very hard to control the exact amount of charge being placed into the cell. Also, the temperature and the supply voltage affect this process making it even harder to control.

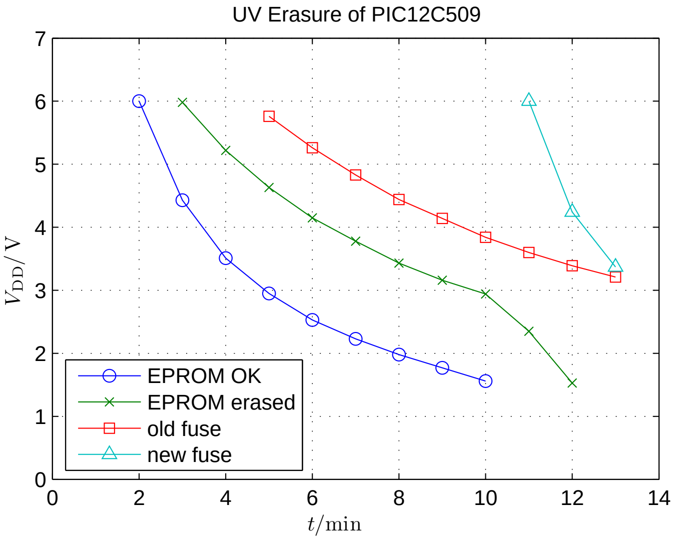

Fig. 4. Memory contents of PIC12C509 tested at different power-supply voltages after UV erasure

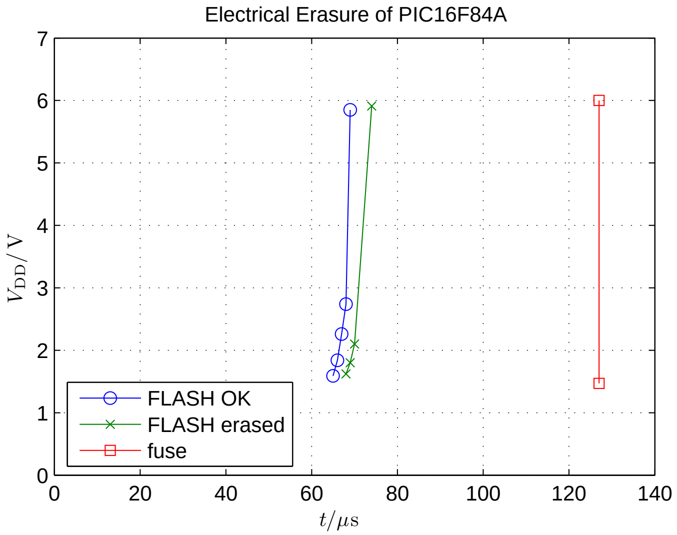

Fig. 5. Memory contents of PIC16F84A tested at different power-supply voltages after electrical erasure

The next experiment was done to the Microchip PIC16F84A microcontroller [13] which has Flash program memory and EEPROM data memory. A similar test sequence was applied with the only difference that electrical erasing was used (Figure 5). A huge difference in the memory behaviour can be observed. The memory erase starts 65 μs after the 'chip erase' command was received and by 75 μs the memory is erased. However, this time changes if the temperature or the supply voltage is changed. For example, if the chip is heated to 35 °C the memory erase starts at 60 μs and is finished by 70 μs. The security fuse requires at least 125 μs to be erased giving at least five times excess for reliable memory erase. Reducing the power supply voltage increases the erase time for both the memory and the fuse erase, so that the ratio remains practically the same. It should be mentioned that unless terminated by the hardware reset, the chip erase operation lasts for at least 1 ms. Both this fact and the fast erase time give an impression that EEPROM and Flash memories have fewer problems with data remanence and therefore should offer better security protection. I decided to investigate whether this is true or not.

In my early experiments with the security protection in PIC microcontrollers, I noticed that the same PIC16F84 chip behaves differently if it is tested right after the erase operation was completed. As this microcontroller is no longer in use and has been replaced by the PIC16F84A, the testing was applied to the new chip.

As can be seen from Figure 5, the memory is completely erased and read as all 1's well before the end of the standard 10 ms erase cycle. The threshold of the cell's transistors becomes very low after the erase and cannot be measured the same way as with UV EPROM because the chip stops functioning if the power supply drops below 1.5 V. With the power glitch technique, it is possible to reduce the supply voltage down to 1 V for a short period of time – enough for the information from memory to be read and latched into the internal buffer. But this is still not enough to shift the reference voltage of the sense amplifier low enough to detect the threshold of the erased cells. To achieve the result another trick was used in addition to the power glitch. The threshold voltage of all the floating gate transistors inside the memory array was shifted temporarily by $V_\Delta = 0.6$–0.9 V, so that $V_{\mathrm{TH}} = K \cdot V_{\mathrm{DD}} - V_\Delta$. As a result it became possible to measure the threshold voltage of an erased cell which is close to 0 V. This was achieved by precisely controlling the memory erase operation, thus allowing the substrate and control gates to be precharged and terminating the process before the tunnelling is started. As a result, the excess charge is trapped in the substrate below the floating gate, and shifts the threshold of the transistor. The process of recombination of the trapped excess charge could take up to one second, which is enough to read the whole memory from the device. This can be repeated for different supply voltages combined with power glitches, in order to estimate the threshold of all the transistors in the memory array.

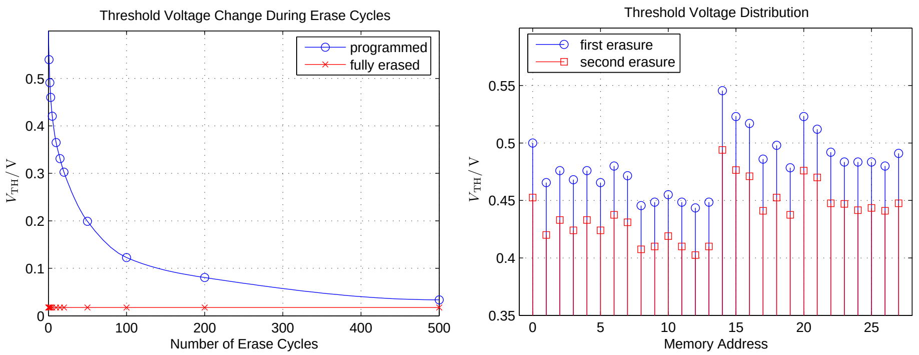

Fig. 6. Change of the threshold voltage during erasure for programmed and previously erased cells (left) and for previously programmed cells after the second erase cycle (right) in PIC16F84A

Applying the above test to differently programmed and erased chips, the diagrams for threshold voltage dependence in the Flash program memory from different factors such as the number of erased cycles (Figure 6, left) and memory address (Figure 6, right) were built. As can be seen, the charge is not entirely removed from the floating gate even after one hundred erase cycles thus making it possible for the information to be extracted from the memory. This was measured on a sample after 100 program/erase cycles to eliminate the effect of the threshold shift taking place in a virgin cell. At the same time the memory analysis and extraction is complicated by the fact that the difference in threshold voltages between the memory cells is larger than within the same cell after single erase cycle. The practical way to avoid this problem is to use the same cell as a reference and compare the measured threshold level with itself after the extra erase operation is applied to the chip. Very similar results were received for the EEPROM data memory inside the same PIC16F84A chip. The only difference was that the threshold voltage after ten erase cycles was very close to that of the fully erased cell, thus making it almost impossible to recover the information if the erase operation was applied more than ten times.

In the next test, the chip was programmed with all 0's before applying the erase operation. As a result it was practically impossible to distinguish between previously programmed and non-programmed cells. That means that pre-programming the cells before the erase operation could be a reasonably good solution to increase the security of the on-chip memory.

One more thing should be mentioned in connection with hardware security. Some microcontrollers have an incorrectly designed security protection fuse, which gets erased earlier than the memory. As a result, if the 'chip erase' operation is terminated prematurely, information could be read from the on-chip memory in a normal way. That was the case, for example, for the Atmel AT89C51 microcontroller. When this became known in the late nineties, Atmel redesigned the chip layout and improved security to prevent this attack, so that chips manufactured since 1999 do not have this problem. Nowadays, most microcontroller manufacturers design their products so that the security fuses cannot be erased before the main memory is entirely cleared, thus preventing this low cost attack on their devices.

5 Semi-Invasive Results

The first experiment was performed on the PIC16F84A microcontroller to check whether it would be possible to extract any information from previously erased memory using semi-invasive methods with the setup mentioned in Section 3.

The location of the memory was initially found under a normal optical microscope. Then, using a proprietary laser scanning setup [16], areas sensitive to the ionisation with laser radiation (bright areas) were found (Figure 7).

A standard Flash memory array consists of the current source, memory cells, row and column selectors and a sense amplifier consisting of an amplifier and a comparator to the reference cell signal which will distinguish between 0 and 1 [8]. Obviously, if we are interested in restoring the state of previously erased or discharged cell we have to either reduce the current flowing through the cell, or increase the reference voltage of the read sense amplifier, or reduce the coefficient of the amplification itself.

Because the laser can only generate the current in p-n junctions, it is not possible to manipulate the transistor in all of the desired ways. However, for most memories built with NMOS technology this will work quite well as the laser will inject current with the opposite polarity to the current sent through the memory cells.

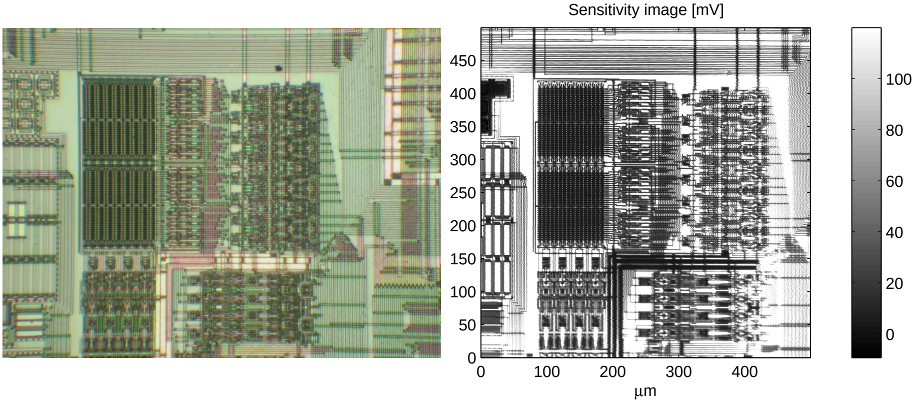

Fig. 7. Optical and laser-scanned images of the PIC16F84A EEPROM area

In my experiments I erased the data EEPROM memory for the time necessary for the memory to be read back fully erased at minimum and maximum power supply voltages. Then the sample was placed under a microscope and several areas were tested with a laser pointer beam with powers ranging from 10 μW to 5 mW. Better results were received when either the area close to the column selector or the area close to the input of the sense amplifier was exposed to the laser beam. For each memory bit the value of the laser power corresponding to the change of its value from 1 to 0 was stored in the file. Due to the reason mentioned in the previous chapter it was not possible to extract the memory contents directly by adjusting the reference voltage of the sense amplifier. Therefore, after the first measurement an extra memory erase operation was performed and the next measurement was done. Comparing the results for each memory cell revealed its content because a previously programmed cell had changed its threshold value while a non-programmed cell had not.

Going back to Figure 5 it can be noticed that when more than 75 μs has elapsed since the erase command the contents of the memory cannot be read directly. Using the above technique I was able to reliably extract the information from the memory after a 150 μs erase pulse. This is still well below the standard 10 ms erase operation but is sufficient to erase the security fuse so that the attacker can perform a ‘chip erase’ operation and then extract the information from the memory.

The most important advantage of the semi-invasive technique is that it is independent of the power supply voltage and uses only laser power alteration to measure the threshold voltage of the memory transistors. This overcomes certain protections used in modern secure chips where either voltage monitors or voltage stabilisers are used.

The next step in my research was to test whether such a semi-invasive technique would work for modern submicron chips. As a target for my next experiments I chose the Atmel ATmega8 microcontroller [17] which employs 0.35 μm

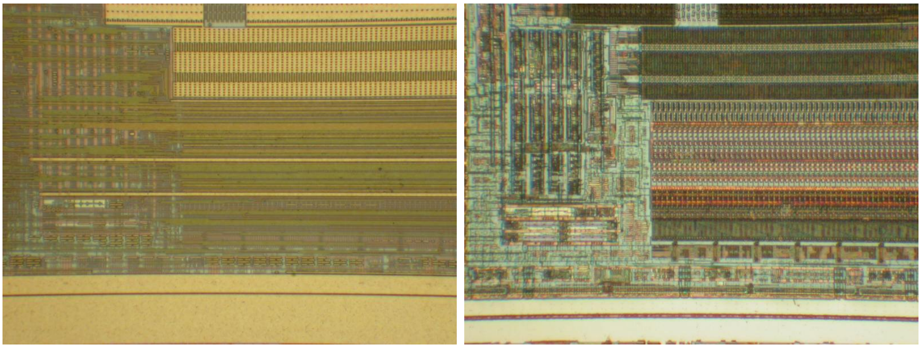

Fig. 8. Optical image of EEPROM area in the ATmega8 microcontroller before and after wet chemical etching

technology (Figure 8). It has three metal layers and as a result there is very little information that can be gained from direct optical observation of the chip under a microscope. To solve this problem and find the memory components on the die it was deprocessed using a wet chemical etching technique. The same die with the top metal layer removed is shown in Figure 8. As a result of this operation all of the memory arrays located on the chip die were recognised.

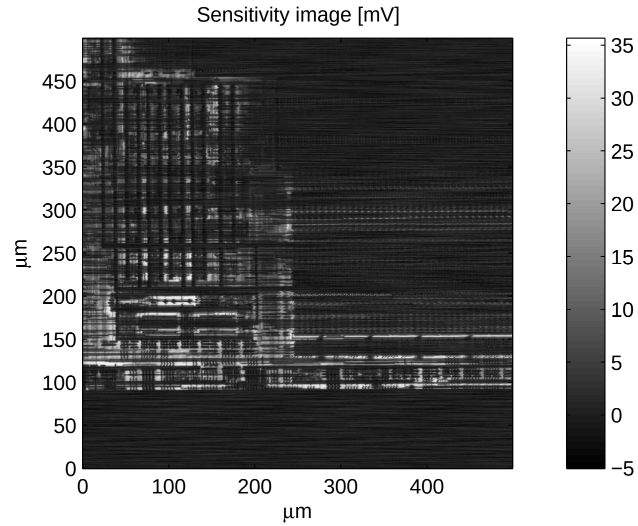

Fig. 9. Laser-scanned image of the ATmega8 EEPROM area

To find the active areas for the laser injections, the previously mentioned laser scanning technique was used. However, as the chip was built with smaller

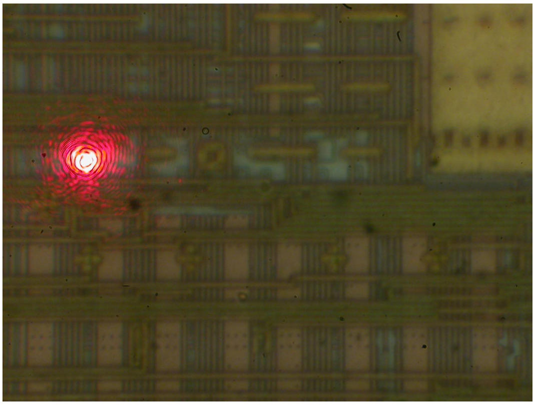

Fig. 10. Focusing the laser on the ATmega8 die using a 100× objective

technology and a large part of its surface is covered with metal wires, only a small part of the die was sensitive to the laser beam (Figure 9) and the injected current was significantly smaller than in case of PIC16F84A chip which has 0.9 μm technology. In addition, the chemical-mechanical polishing used in the production of ATmega8 die reduces the transparency of the layers and only a small fraction of light reaches the active area on the chip (Figure 10). All these facts made the analysis and further testing of this chip more difficult.

The ATmega8 microcontroller employs a very reliable security protection feature which ensures that the memory is erased well before the security fuse that prevents external access to the memory. In my experiments, I was able to extract information from the erased memory only if the erase pulse was less than 100 μs long, whereas the standard 'chip erase' operation takes 10 ms. It was still impossible to read the memory contents even after a 70 μs long erase pulse at both minimum and maximum power supply voltages, but this is still not enough to overcome the security protection. However, semi-invasive methods again showed their advantages, especially because I was not able to find any non-invasive approach for extracting the information from an erased ATmega8 microcontroller.

6 Countermeasures

To avoid data remanence attacks in secure applications, the developer should follow some general design rules that help to make data recovery from semiconductor memories harder [5]:

- Cycle EEPROM/Flash cells 10–100 times with random data before writing anything sensitive to them, to eliminate any noticeable effects arising from the use of fresh cells.
- Program all EEPROM/Flash cells before erasing them to eliminate detectable effects of residual charge.

- Remember that some non-volatile memories are too intelligent, and may leave copies of sensitive data in mapped-out memory blocks after the active copy has been erased. That also applies to file systems, which normally remove the pointer to the file rather than erasing the file itself.
- Use the latest highest-density storage devices, as the newest technologies generally make data recovery more difficult.
- Using memories covered with top metal layer or built with modern deep submicron technologies helps against semi-invasive attacks because such attacks require the laser beam to reach the transistor active areas.

Using encryption, where applicable, also helps to make data recovery from erased memory more difficult. Ideally, for secure applications, each semiconductor memory device should be evaluated for data remanence.

7 Conclusions

Floating-gate memory devices, such as UV EPROM, EEPROM and Flash, have data remanence problems. From some samples, information can still be recovered after 100 erase cycles. Even if the residual charge cannot be detected with existing methods, this might be possible in the future with new technologies. Hardware designers should pay attention to the evaluation of components planned to be used in systems sensitive to data remanence.

Fortunately, the presented techniques for extracting erased memory can be applied only to a limited number of chips with EEPROM or Flash memory. Firstly, some microcontrollers, such as the Texas Instruments MSP430 family [14], have an internally stabilised supply voltage for the on-chip memory. Changing the power supply from 1.8 V to 3.6 V does not affect a memory read operation from partially erased cells. Secondly, most microcontrollers fully reset and discharge the memory control circuit if the chip is reset or the programming mode is re-entered. But still, if the memory contents do not disappear completely, this can represent a serious threat to any security based on an assumption that the information is irrecoverable after one memory erase cycle. Where non-invasive methods fail, invasive methods could still succeed. For example, the memory control circuit can be modified using a focused ion-beam workstation to directly access the reference voltage, the current source or the control gate voltage. Finally, some chips program all the memory locations before applying the erase operation. This makes it almost impossible to extract any useful information from the erased memory.

Semi-invasive methods have once again shown their use in hardware security analysis. However, they have some limitations, especially for modern deep submicron technologies, where multiple metal layers and small transistor size prevent easy and precise analysis. Further improvements to these methods might involve approaching the die from its reverse side but this requires the use of more expensive equipment.

References

1. A Guide to Understanding Data Remanence in Automated Information Systems. Version 2, September 1991, NSA/NCSC Rainbow Series
2. Peter Gutmann: Secure Deletion of Data from Magnetic and Solid-State Memory. 6th USENIX Security Symposium Proceedings, San Jose, California, July 22–25, 1996, pp. 77–89
3. Ross J. Anderson, Markus G. Kuhn: Tamper Resistance – a Cautionary Note. The Second USENIX Workshop on Electronic Commerce, Oakland, California, November 18–21, 1996
4. Sergei Skorobogatov: Low Temperature Data Remanence in Static RAM. Technical Report UCAM-CL-TR-536, University of Cambridge, Computer Laboratory, June 2002
5. Peter Gutmann: Data Remanence in Semiconductor Devices. 10th USENIX Security Symposium, Washington, D.C., August 13–17, 2001
6. Intel StrataFlash Memory (J3), 28F256J3, 28F128J3, 28F640J3, 28F320J3. `ftp://download.intel.com/design/flcomp/datashts/29066719.pdf`
7. P.L. Rolandi, R. Canegallo, E. Chioffi, D. Gerna, G. Guaitini, C. Issartel, A. Kramer, F. Lhermet, M. Pasotti: 1M-Cell 6b/Cell Analog Flash Memory for Digital Storage. SGS-Thomson Microelectronics, IEEE International Solid-State Circuits Conference (ISSCC), Agrate Brianza, Italy, 1998
8. William D. Brown, Joe E. Brewer: Nonvolatile Semiconductor Memory Technology: A Comprehensive Guide to Understanding and Using NVSM Devices. IEEE Press, 1997
9. Intel 28F010 and 28F020, 5 Volt Bulk Erase Flash Memory. `http://www.sunmark.com/datasheets/28f010.pdf`
10. Paolo Pavan, Luca Larcher, Massimiliano Cuozzo, Paola Zuliani, Antonino Conte: A Complete Model of E2PROM Memory Cells for Circuit Simulations. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 22(8), August 2003
11. H. Kume, H. Yamamoto, T. Adachi, T. Hagiwara, K. Komori, T. Nishimoto, A. Koike, S. Meguro, T. Hayashida, T. Tsukada: A Flash-erase EEPROM cell with an asymmetric source and drain structure. IEEE IEDM Technical Digest, 1987, pp. 560–563
12. Microchip PIC12C5XX Data Sheet, 8-Pin, 8-Bit CMOS Microcontrollers. `http://ww1.microchip.com/downloads/en/DeviceDoc/40139e.pdf`
13. Microchip PIC16F84A Data Sheet, 18-pin Enhanced Flash/EEPROM 8-bit Microcontroller. `http://ww1.microchip.com/downloads/en/DeviceDoc/35007b.pdf`
14. Texas Instruments, MSP430x1xx Family, User's Guide. `http://focus.ti.com/lit/ug/slau049e/slau049e.pdf`
15. Sergei Skorobogatov, Ross Anderson: Optical Fault Induction Attacks. Cryptographic Hardware and Embedded Systems Workshop (CHES-2002), LNCS, Vol. 2523, Springer-Verlag, 2002, pp. 2–12
16. Sergei Skorobogatov: Semi-invasive attacks – A new approach to hardware security analysis. Technical Report UCAM-CL-TR-630, University of Cambridge, Computer Laboratory, April 2005
17. Atmel ATmega8 Data Sheet, 8-bit, 8K Bytes In-System Programmable Flash Microcontroller. `http://www.atmel.com/dyn/resources/prod_documents/doc2486.pdf`