Evaluation of the Masked Logic Style MDPL on a Prototype Chip

Thomas Popp[1], Mario Kirschbaum[1], Thomas Zefferer[1], and Stefan Mangard[2,⋆]

[1] Institute for Applied Information Processing and Communications (IAIK)
Graz University of Technology
Inffeldgasse 16a, 8010 Graz, Austria
{Thomas.Popp, Mario.Kirschbaum, Thomas.Zefferer}@iaik.tugraz.at
[2] Infineon Technologies AG
Security Innovation
Am Campeon 1-12, 85579 Neubiberg, Germany
Stefan.Mangard@infineon.com

Abstract. MDPL has been proposed as a masked logic style that counteracts DPA attacks. Recently, it has been shown that the so-called "early propagation effect" might reduce the security of this logic style significantly. In the light of these findings, a $0.13\,\mu m$ prototype chip that includes the implementation of an 8051-compatible microcontroller in MDPL has been analyzed. Attacks on the measured power traces of this implementation show a severe DPA leakage. In this paper, the results of a detailed analysis of the reasons for this leakage are presented. Furthermore, a proposal is made on how to improve MDPL with respect to the identified problems.

Keywords: DPA-Resistant Logic Styles, Masked Logic, Dual-Rail Precharge Logic, Early Propagation Effect, Improved MDPL, Prototype Chip

1 Introduction

One of the biggest challenges of designers of cryptographic devices is to provide resistance against side-channel attacks [1]. These attacks pose a serious threat to the security of implementations of cryptographic algorithms in practice. In particular, differential power analysis (DPA) attacks [7] are known to be very powerful.

During the last years, several proposals to counteract DPA attacks at the logic level have been published. The basic idea of these proposals is to design logic cells with a power consumption that is independent of the data they process. Essentially, there exist two approaches to build such cells. The first approach is to design these cells from scratch. This implies that a completely new cell library needs to be designed for every process technology. Examples of such logic styles are SABL [14], RSL [13], DRSL [4], and TDPL [3].

⋆ This work was done while the author was with Graz University of Technology.

The alternative to this approach is to build secure logic cells based on existing standard cells. In this case, the design effort for new cell libraries is minimal. This is the motivation for logic styles like WDDL [14], MDPL [11], and FGL [5].

Of course, each of the proposed logic styles also has other pros and cons besides the design effort for the cells. Dual-rail precharge (DRP) logic styles (e.g. SABL, TDPL, WDDL), which belong to the group of hiding logic styles, are for example smaller than masked logic styles (e.g. MDPL, RSL, DRSL, FGL). However, the security of DRP logic styles strongly depends on the balancing of complementary wires in the circuit, while this is not the case for masked logic styles. Design methods to balance complementary wires can be found in [6], [15] and [16] .

Another property that leads to a side-channel leakage of certain logic styles has been identified in [8] and [12]. In these articles, the so-called "early propagation effect" is described. The main observation is that logic cells are insecure if the cells switch at data-dependent moments in time. In [8], this effect is discussed for SABL, and in [12], it is discussed for WDDL and MDPL. Furthermore, results of experiments on an FPGA are presented that confirm the early propagation effect in practice. In [4], a proposal to prevent early propagation in case of RSL has been published.

The current article also focuses on the early propagation effect. In fact, we confirm the results of [12] for ASIC implementations. For this purpose, we use an 8051 microcontroller core that has been implemented in three different logic styles (CMOS, MDPL, and a DRP variant based on custom cells). The comparison of the different implementations shows that the MDPL core can almost be attacked as easily as the CMOS core due to the early propagation effect. The DRP core is more robust against DPA attacks and it can only be attacked with a significantly larger number of measurements.

The remainder of this article is organized as follows. Section 2 gives an overview of the prototype chip that has been used in the experiments. The respective DPA-resistant logic styles in which the 8051 microcontroller core has been implemented are introduced shortly. Results of the DPA attacks on the measured power consumption are presented in Section 3. These results confirm that MDPL has significant problems in terms of DPA resistance. In Section 4, these problems are analyzed in detail with the help of transistor-level simulations and logic simulations. In Section 5, improvements for MDPL are proposed that avoid the DPA leakage caused by early propagation. Finally, Section 6 provides conclusions.

2 The Prototype Chip

This section introduces the prototype chip that has been used to analyze the effectiveness of the DPA-resistant logic styles in practice. The general architecture of the prototype chip is shown in Figure 1. The system that has been implemented consists of the following main parts: an Intel 8051-compatible microcontroller and an AES cryptographic module that is used as a coprocessor of the 8051 microcontroller. The microcontroller features 128 bytes of internal random-access memory (IRAM), a serial interface (RS-232), and an 8-bit parallel input/output port. The program that is executed resides in an external program memory (PROM) chip. Additionally, an external RAM (XRAM) chip can also be attached.

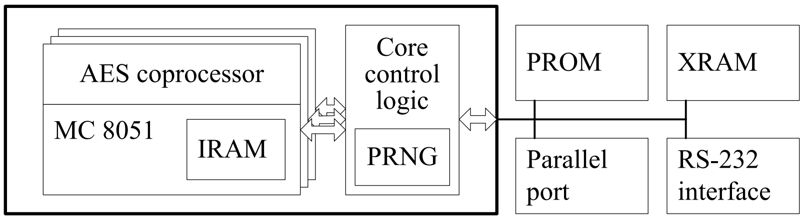

Fig. 1. General architecture of the prototype chip.

The system has been implemented in different cores using DPA-resistant logic styles (MDPL, DRP) and standard CMOS logic. The cell netlist of all cores is practically identical, only the implementations of the cells are done in the respective logic style. The complementary wires in the DRP core have been balanced by routing them in parallel [15]. The CMOS core acts as a reference implementation.

The core control logic is used to activate the currently selected core, *i.e.* supplying it with the clock signal and connecting its input and output signals to the corresponding chip pins. Part of the core control logic is a pseudo-random number generator (PRNG), which produces the mask values for MDPL. The PRNG is controlled by the currently selected 8051 microcontroller via additional parallel ports that are connected on-chip to the PRNG. The main operations of the PRNG are: load a seed value, generate one random bit per clock cycle, provide a constant mask value, and stop operating.

In a masked logic style like MDPL, the power consumption is made independent of the processed data by concealing this data with a random mask and by operating only on the masked data. MDPL uses boolean masking, *i.e.* every signal d in the circuit is represented by the masked signal $d_m = d \oplus m$, where m is the random mask. MDPL also works in a DRP-like manner in order to avoid glitches, which have negative effects on the DPA resistance of masking [10].

A DRP logic style achieves independence between the power consumption and the processed data by making the power consumption constant. Every signal d in the circuit is represented by two complementary signals d and $\overline{d}$. Furthermore, both signals are precharged to a constant value in every clock cycle. Thus, exactly one signal of every signal pair switches in each clock cycle. If the complementary wires carrying a signal pair are balanced (*i.e.* have the same capacitive load) the power consumption is constant.

3 DPA Attacks Based on Measured Power Traces

The effectiveness of the DPA-resistant logic styles has been analyzed by attacking the 8051 microcontroller of the respective core while it performs an internal MOV operation, *i.e.* one byte of data is moved from one IRAM register to another one. The value in the destination register has been set to 0 before this operation. In the DPA attack, the Hamming weight (HW) of the moved byte has been used as the predicted power consumption. In the given scenario, the HW of the moved byte equals the number of bit transitions at the destination register. Besides this leakage model, the correlation coefficient has been used in the DPA attack to quantify the relationship between the predicted and the measured power consumption [2].

The measurement setup that has been used to record the power consumption of the prototype chip while it executes the MOV operation consists of three main parts: a board that holds the prototype chip and necessary external devices like power regulators and the PROM, a digital oscilloscope, and a host PC that controls both the oscilloscope and the prototype chip on the board. The bandwidth of the oscilloscope has been $1\,GHz$. A suitable differential probe has been used to measure the power consumption via a $10\,\Omega$ measurement resistor in the VDD line of the prototype chip. The voltage levels required by the prototype chip are $1.5\,V$ for the core cells and $3.3\,V$ for the I/O cells.

An investigation of the measured power traces has revealed the presence of significant disturbances within some traces, which have a negative effect on the DPA attack. Highly disturbed traces have been identified by calculating the "sum of squared differences" of each trace and the mean trace of a set of measurements: first, the difference between a trace and the mean trace was calculated pointwise; these difference values were then squared and summed up. Traces for which this sum exceeded some threshold were considered as highly disturbed and were filtered out.

The clock frequency provided to the prototype chip has been the same in all three attacks: $3.686\,MHz$. The relevant settings of the digital oscilloscope have also been the same in the measurement runs for the three different cores:

- **Vertical resolution:** $39\,mV/Div$
- **Input coupling:** $1\,M\Omega - AC$
- **Horizontal resolution:** $0.2\,\mu s/Div$
- **Sampling rate:** $4\,GS/s$
- **Points per power trace:** 8000 (follows from horizontal resolution and sampling rate)

Figure 2 shows the result of the DPA attack for the MOV operation on the CMOS core. The correlation trace when using the correct data bytes to generate the power hypothesis is plotted in black. Additionally, 10 correlation traces are plotted in gray for which random data values have been used to generate the power hypotheses in the DPA attack. As expected, a rather high maximum correlation coefficient of 0.3068 occurs for the correct power hypothesis in the

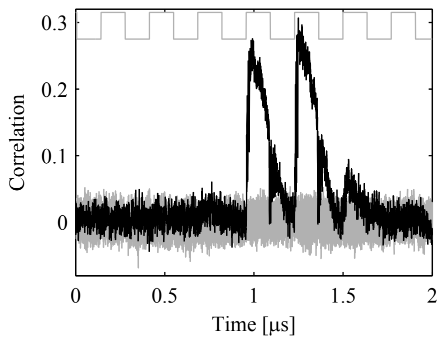

Fig. 2. Result of the DPA attack on the CMOS core: internal MOV operation in the IRAM, 5000 samples, correlation trace for correct power hypothesis is plotted in black.

clock cycles where the MOV operation is executed. The first correlation peak occurs when the moved byte is fetched from the source register via the internal bus to the destination register. The second peak occurs when the moved byte is stored in the destination register and removed from the internal bus. In the 10 correlation traces for random data values, no significant correlation values occur.

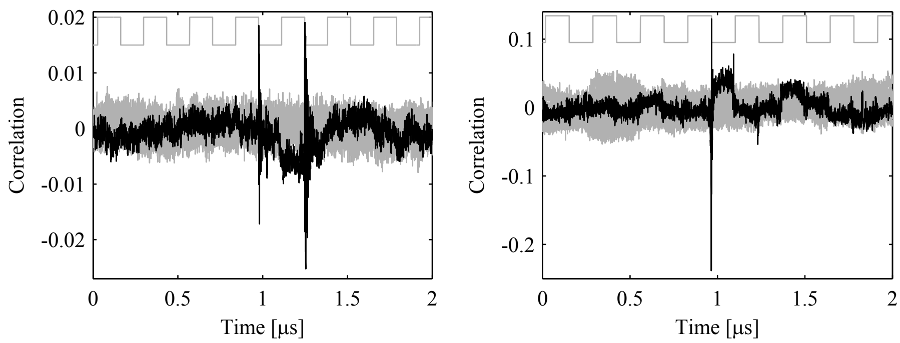

Fig. 3. Results of the DPA attacks on the DRP core (left, 300000 samples) and the MDPL core (right, 5000 samples): internal MOV operation in the IRAM, correlation trace for correct power hypothesis is plotted in black.

As expected, using the DRP logic style reduces the correlation significantly. This is shown in Figure 3 (left). The highest absolute correlation peak here is only 0.025. This leakage in the DRP core is most likely caused by imperfect balanced dual-rail wire pairs. Note that the DRP core precharges when the clock signal is 1 and evaluates when the clock signal is 0.

The correlation trace for the MDPL core depicted in Figure 3 (right) shows a significant leakage in the second clock cycle of the MOV operation. As we will show in the next section, this leakage is mainly caused by the early propagation effect. The highest correlation peak of 0.2385 lies in the range of that one of the CMOS core. Note that the MDPL core has been operated with activated PRNG. As for the DRP core, the MDPL core precharges when the clock signal is 1 and evaluates when the clock signal is 0.

In Table 1, the results of the DPA attacks on the measured power traces of the prototype chip are summarized. The formula to calculate the required power traces for a successful attack from the highest correlation value is given in [9].

Table 1. Results of the DPA attacks on the measured power traces of the prototype chip, internal MOV operation

	Used power traces	Highest absolute correlation peak	Required power traces
CMOS	5000	0.3068	279
DRP	300000	0.0253	43201
MDPL	5000	0.2385	471

Interestingly, attacks on the AES coprocessor did not show any significant DPA leakage neither for the MDPL nor for the DRP core (we considered up to 1 million power traces so far). No significant peaks occurred in the correlation traces for the correct key hypothesis. It seems that the early propagation effect does not affect the MDPL AES implementation in such a way as the 8051 microcontroller implementation. We suspect that the reason lies in the rather different design of both circuits. While the microcontroller is synthesized from a very complex high-level description, the high-level description of the AES module has already been done in a very regular way. This issue needs further investigation, which is not the scope of this paper.

4 Problem Analysis

In this section, the origin of the leakage of the IRAM MOV operation on the MDPL core is analyzed in detail. As shown by Suzuki and Saeki [12], MDPL cells may leak information due to timing differences in the input signals and the early propagation effect, which is not prevented in such cells. Suzuki and Saeki verified their theoretical results by measurements on an FPGA. In the following, we show that these effects are most probably also the cause for the DPA leakage in the MDPL core of the prototype chip.

As already mentioned, the DRP logic style used on the prototype chip is based on custom cells. These cells are implemented in a way that early propagation is avoided, *i.e.* the combinational cells only evaluate after all input signals have reached a differential state. This explains why the peaks in the correlation traces of the DRP core are much smaller than the peaks of the MDPL core.

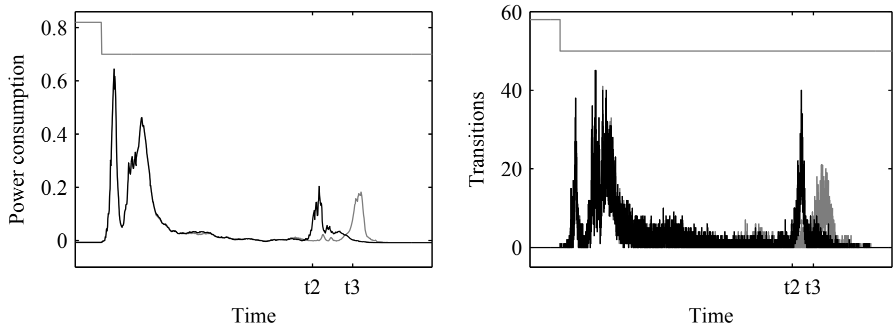

Fig. 4. Power consumption of the MDPL core in a clock cycle of the MOV operation when moving the value $0x00$ (black) and $0xFF$ (gray), the mask is kept 0. Left: transistor-level simulation without interconnect parasitics. Right: transition count at each point in time based on logic simulations including extracted delay information.

4.1 Problem Analysis Based on Transistor-Level Simulations

In a first step of the problem analysis, the cells that are directly involved in the MOV operation have been analyzed with the help of transistor-level simulations. These simulations have been carried out with Nanosim from Synopsys. The transistor netlist of the MDPL core (excluding interconnect parasitics) has been simulated for two cases: moving the value $0x00$ and moving the value $0xFF$ in the IRAM for different mask values. The power consumption in the clock cycle of the MOV operation where the first correlation peak (according to Figure 3 - right) occurs is shown in Figure 4 (left) for mask 0. The first two peaks of the power consumption, which are identical for the values $0x00$ and $0xFF$, occur right after the negative clock edge (start of evaluation phase of MDPL). For the third peak of the power consumption, the time offset $t_3 - t_2$ for the two data values is clearly visible. The time offset is in the range of $1\,ns$. The Nanosim simulations for random mask values have shown that this timing difference is independent of the actual value of the mask. Thus, a correlation occurs in the DPA attack on the MDPL core with activated PRNG.

Next, the reason for this mask-independent time offset has been analyzed. In the simulation results, an MDPL-AND cell has been identified, which switches at the beginning of the time period where the correlation peak occurs. Furthermore, the outputs of this MDPL-AND cell switch with a time difference of approximately $1\,ns$ for the two moved values in the transition from precharge phase to evaluation phase. The transistor-level simulations have also shown that the difference between the arrival times of the input signals A, B, and M of this cell is significantly larger than the propagation delay of the MDPL-AND cell, which consists of two Majority (MAJ) cells (see Figure 7). The input signal A depends on the moved value and signal B is constantly 0. The situation is depicted in Figure 5.

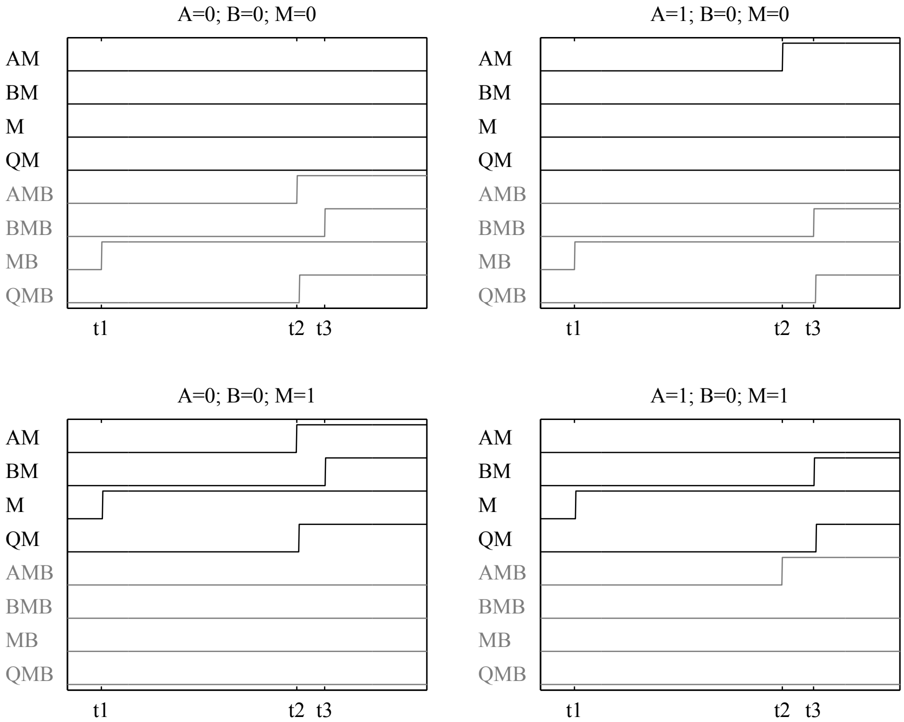

Fig. 5. Signals for the MDPL-AND Majority cells for which early propagation occurs (transistor-level simulation, black: signals of first MAJ cell, gray: signals of second MAJ cell). Signal A depends on the moved value. Signal B is constantly 0.

The timing conditions for the inputs of the MDPL-AND cell are as follows: signals M, $\overline{M}$ arrive first (time t_1), then A_M, $\overline{A_M}$ arrive (time t_2), and at last B_M, $\overline{B_M}$ arrive (time t_3). The mask signals arrive first because they are provided by a so-called mask unit right at the beginning of the evaluation phase and they do not need to go through combinational logic. The delay of the signals B_M, $\overline{B_M}$ is longer than that of the signals A_M, $\overline{A_M}$ because of a higher number of cells in the respective combinational paths.

In the given situation, it turns out that for $A = 0$, always one Majority cell switches at time t_2 (neglecting the propagation delay of the Majority cell). A different mask value only switches the affected Majority cell. For $A = 1$, the Majority cells always switch at time t_3 (again neglecting the propagation delay).

These results clearly show that early propagation causes the dependency between the unmasked data values and the evaluation moment of the MDPL-AND cell. In [12], the authors show the occurrence of leakage due to early propagation for a more general case, *i.e.* the value of B is also variable. Only one cell that shows this behavior would most probably not cause such a significant correlation peak in the DPA attack on the entire chip. However, further investigations have shown that the discussed early propagation effect also occurs for the other seven bits of the moved data value and there are several other MDPL-AND cells which behave in the same way. Furthermore, the outputs of the affected cells are fed into many other MDPL cells before the data values are eventually stored in registers. Thus, also these cells are affected by the data-dependent moment of evaluation. Altogether, there are hundreds of MDPL which evaluate in a data-dependent manner.

Preventing early propagation would mean that the MDPL-AND cell only evaluates when all input signals have arrived, *i.e.* all input signals have been set to differential values. Thus, in both cases ($A = 0$ and $A = 1$), such an improved MDPL cell would always evaluate at time t_3. The DPA leakage caused by the data-dependent evaluation moments of the MDPL-AND cell would be prevented. A proposal on how to avoid early propagation is presented in Section 5.

4.2 Problem Analysis Based on Logic Simulations and Transition Counts

In a last step of the problem analysis, the correlation results based on measured power traces presented in Section 3 have been reproduced by attacking simulated power traces. Transistor-level simulations have not been suitable for this purpose because it would have taken too long to simulate an appropriate amount of power traces for such a big circuit as the analyzed one. Therefore, logic simulations including extracted delay information have been performed. From these results, a basic power trace has been generated by counting the number of transitions at each moment in time. Figure 4 shows that the result of such a simulation (right) looks quite similar to the transistor-level simulation result (left).

Logic simulations of the MOV operation on the MDPL core have then been performed for the 256 different values of the moved byte and random mask values. A subsequent DPA attack on the simulated power traces derived from the logic simulations has led to the results shown in Figure 6. Correlation traces for wrong power hypotheses are plotted in gray while the correlation trace for the correct power hypothesis is plotted in black. The correlation peak in the third clock cycle corresponds to the highest correlation peak shown in Figure 3 (right). It is also the point in time that is shown in detail in Figure 4. The correlation peaks in the first and second clock cycle do not appear in the DPA attack based on the measured power traces. A detailed analysis has shown that these correlations are caused by very small data-dependent variations in the power consumption, which can only be exploited in the attacks based on simulations. These small data-dependent variations most probably occur because the data value that is moved is already stored in the source register before the actual MOV operation takes place. The improved version of MDPL that is presented in the next section is capable of removing all these correlation peaks in a DPA attack based on logic simulations.

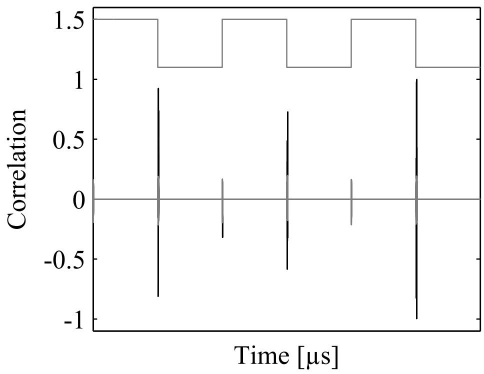

Fig. 6. Result of the DPA attack on the MDPL core: transition count based on logic simulation of internal MOV operation in the IRAM, 256 samples, correlation trace for correct power hypothesis is plotted in black.

5 Improving MDPL

As it clearly turned out in the last section, logic styles that are secure against DPA attacks must avoid early propagation. Otherwise, a power consumption occurs that depends on the unmasked data values due to data-dependent evaluation moments.

The differential encoding of the signals in MDPL circuits allows to detect the point in time in the evaluation phase where all input signals of a cell are in a valid differential state. A cell that avoids early propagation must delay the evaluation moment until this point in time. In [4], the logic style DRSL is presented, which implements such a behavior in the evaluation phase.

As it has also been shown in [12], it is necessary to avoid an early propagation effect in the precharge phase as well. Our DPA-attack results on the measurements of the MDPL core shown in Figure 3 (right) confirm this practically. After the high correlation peak at the beginning of the evaluation phase, there occurs a smaller but still clearly recognizable correlation peak at the beginning of the subsequent precharge phase (around 1.1 μs).

According to our analysis, DRSL does not completely avoid an early propagation effect in the precharge phase. The reason is that the input signals, which arrive at different moments, can still directly precharge the DRSL cell. The propagation delay of the evaluation-precharge detection unit (EPDU) leads to a time frame in which this can happen. Only after that time frame, the EPDU unconditionally precharges the DRSL cell. Our simulations with an intermediate version of an improved MDPL cell confirmed this - there still occurred correlation peaks in the precharge phase. Thus, the input signals of a cell must be maintained until the EPDU generates the signal to precharge the cell.

Figure 7 shows the schematic of an improved MDPL (iMDPL) cell with respect to the early propagation effect. The three OR and the NAND cell on

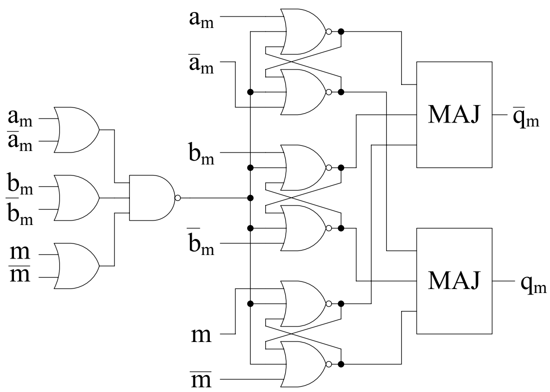

Fig. 7. An iMDPL-AND cell. The original MDPL-AND cell only consists of the two Majority cells MAJ.

the left side implement the EPDU, which generates 0 at its output only if all input signals a_m, b_m, and m are in a differential state. The following three set-reset latches, each consisting of two cross-coupled 3-input NORs, work as gate elements. As long as the EPDU provides a 1, each NOR produces a 0 at its output. Thus, the outputs of both MAJ cells are 0 and the iMDPL cell is in the precharge state.

When the EPDU provides a 0 because all input signals have been set to a differential state, the set-reset latches evaluate accordingly and the MAJ cells produce the intended output according to the masked AND function. Note that this evaluation only happens after all input signals have arrived differentially, *i.e.* no early propagation occurs. However, this is only true if the input signals reach the inputs of the three latches before the EPDU sets its output to 0. Fortunately, this timing constraint is usually fulfilled because of the propagation delay of the EPDU.

Finally, if the first input signal is set back to the precharge value, the EPDU again produces a 1 and all six outputs of the set-reset latches switch to 0. Note that the set-reset latches are only set to this state by the EPDU and not by an input signal that switches back to the precharge value. Thus, also an early propagation effect at the onset of the precharge phase is prevented. An iMDPL-OR cell can be derived from an iMDPL-AND cell by simply swapping (*i.e.* inverting) the mask signals m and $\overline{m}$.

Figure 8 shows the cell schematic of an improved MDPL-DFF. In principle, the functionality is the same as the one of the original MDPL-DFF [11]. The additional cells just control the start of the evaluation and the precharge moments as described for the iMDPL-AND cell. Note that the iMDPL-AND cell used in the iMDPL-DFF is actually used as an iMDPL-NAND cell. The unnecessary MAJ cell in the iMDPL-AND cell, which produces the output signal q_m, can be removed.

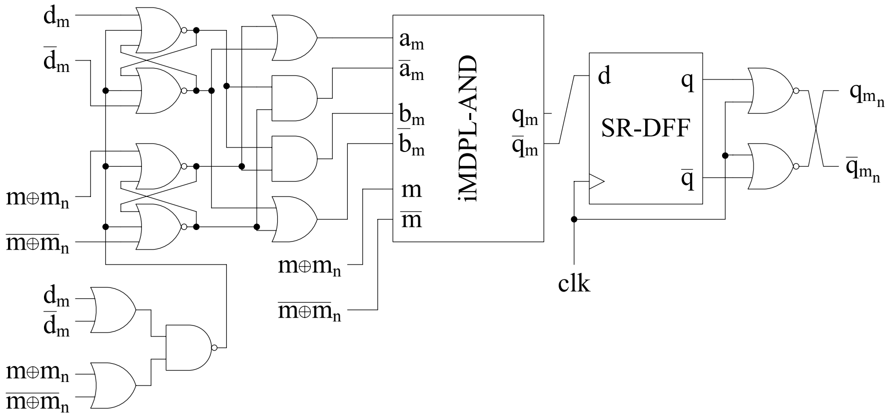

Fig. 8. An iMDPL-DFF. The original MDPL-DFF does not have the two input latches and the EPDU.

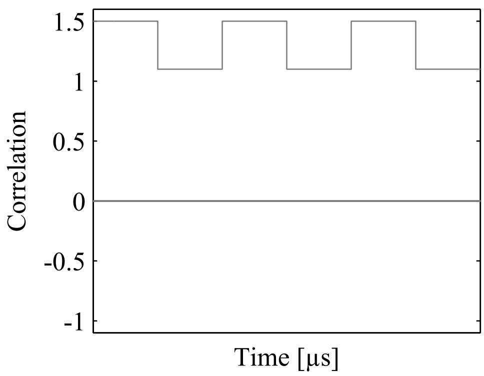

Fig. 9. Result of the DPA attack on the iMDPL core: transition count based on logic simulation of internal MOV operation in the IRAM, 256 samples, correlation trace for correct power hypothesis is plotted in black.

In Figure 9, the correlation traces when attacking simulated power traces of the core implemented in iMDPL are shown. In order to perform the necessary logic simulations, the MDPL cells in the circuit netlist of the microcontroller core have been replaced by the corresponding iMDPL cells. The correlation traces for both the correct and the wrong power hypotheses show an ideal flat line for the attacked MOV operation. This indicates that the DPA leakage due to the early propagation effect is removed successfully.

Obviously, the price that has to be paid for the improvements in terms of early propagation is a further significant increase of the area requirements of iMDPL cells compared to MDPL. Since the iMDPL cells are already quite complex, exact figures for the area increase can not be given in general because it depends significantly on the particular standard cell library that is used to implement an iMDPL circuit. For example, there might be a standard cell available that implements the complete EPDU - such a cell is usually called OAI222. However, one can expect an increase of the area by a factor of up to 3 compared to original MDPL. This makes it clear that carefully finding out which parts of a design really need to be implemented in DPA-resistant logic is essential to save chip area.

A significant reduction of the cell size can be achieved by designing new standard cells that implement the functionality of iMDPL. Of course, that has the well known disadvantages of a greatly increased design and verification effort. Furthermore, a change of the process technology would then mean spending all the effort to design an iMDPL standard cell library again.

6 Conclusions

In this paper, we have presented the results of DPA attacks on a prototype chip that implements an 8051-compatible microcontroller in different DPA-resistant logic styles. Our analysis focused on the core that is implemented in the masked logic style MDPL. For this core, the DPA attacks on measured power traces show a significant leakage when attacking a MOV operation of one byte in the internal memory. Further analysis based on simulations on the transistor level and on the logic level showed that the early propagation effect is the major cause for this leakage. Furthermore, a proposal for improving MDPL to avoid the early propagation effect is made in this paper. These cells can still be implemented based on commonly available standard cells. The main drawback is a further increase of the area requirements of the improved version of MDPL compared to the original version by a factor of 3.

Acknowledgements. This work has been supported by the Austrian Government through the research program FIT-IT Trust in IT Systems (Project GRANDESCA, Project Number 813434).

References

1. R. J. Anderson, M. Bond, J. Clulow, and S. P. Skorobogatov. Cryptographic Processors—A Survey. *Proceedings of the IEEE*, 94(2):357–369, February 2006. ISSN 0018-9219.
2. E. Brier, C. Clavier, and F. Olivier. Correlation Power Analysis with a Leakage Model. In M. Joye and J.-J. Quisquater, editors, *Cryptographic Hardware and Embedded Systems – CHES 2004, 6th International Workshop, Cambridge, MA, USA, August 11-13, 2004, Proceedings*, volume 3156 of *Lecture Notes in Computer Science*, pages 16–29. Springer, 2004.
3. M. Bucci, L. Giancane, R. Luzzi, and A. Trifiletti. Three-Phase Dual-Rail Pre-Charge Logic. In L. Goubin and M. Matsui, editors, *Cryptographic Hardware and Embedded Systems – CHES 2006, 8th International Workshop, Yokohama, Japan,*

October 10-13, 2006, Proceedings, volume 4249 of *Lecture Notes in Computer Science*, pages 232–241. Springer, 2006.
4. Z. Chen and Y. Zhou. Dual-Rail Random Switching Logic: A Countermeasure to Reduce Side Channel Leakage. In L. Goubin and M. Matsui, editors, *Cryptographic Hardware and Embedded Systems – CHES 2006, 8th International Workshop, Yokohama, Japan, October 10-13, 2006, Proceedings*, volume 4249 of *Lecture Notes in Computer Science*, pages 242–254. Springer, 2006.
5. W. Fischer and B. M. Gammel. Masking at Gate Level in the Presence of Glitches. In J. R. Rao and B. Sunar, editors, *Cryptographic Hardware and Embedded Systems – CHES 2005, 7th International Workshop, Edinburgh, UK, August 29 - September 1, 2005, Proceedings*, volume 3659 of *Lecture Notes in Computer Science*, pages 187–200. Springer, 2005.
6. S. Guilley, P. Hoogvorst, Y. Mathieu, and R. Pacalet. The "Backend Duplication" Method. In J. R. Rao and B. Sunar, editors, *Cryptographic Hardware and Embedded Systems – CHES 2005, 7th International Workshop, Edinburgh, UK, August 29 - September 1, 2005, Proceedings*, volume 3659 of *Lecture Notes in Computer Science*, pages 383–397. Springer, 2005.
7. P. C. Kocher, J. Jaffe, and B. Jun. Differential Power Analysis. In M. Wiener, editor, *Advances in Cryptology - CRYPTO '99, 19th Annual International Cryptology Conference, Santa Barbara, California, USA, August 15-19, 1999, Proceedings*, volume 1666 of *Lecture Notes in Computer Science*, pages 388–397. Springer, 1999.
8. K. J. Kulikowski, M. G. Karpovsky, and A. Taubin. Power Attacks on Secure Hardware Based on Early Propagation of Data. In *12th IEEE International On-Line Testing Symposium (IOLTS 2006), July 10-12, 2006*, pages 131–138. IEEE Computer Society, July 2006.
9. S. Mangard, E. Oswald, and T. Popp. *Power Analysis Attacks – Revealing the Secrets of Smart Cards*. Springer, 2007. ISBN 978-0-387-30857-9.
10. S. Mangard, T. Popp, and B. M. Gammel. Side-Channel Leakage of Masked CMOS Gates. In A. Menezes, editor, *Topics in Cryptology - CT-RSA 2005, The Cryptographers' Track at the RSA Conference 2005, San Francisco, CA, USA, February 14-18, 2005, Proceedings*, volume 3376 of *Lecture Notes in Computer Science*, pages 351–365. Springer, 2005.
11. T. Popp and S. Mangard. Masked Dual-Rail Pre-Charge Logic: DPA-Resistance without Routing Constraints. In J. R. Rao and B. Sunar, editors, *Cryptographic Hardware and Embedded Systems – CHES 2005, 7th International Workshop, Edinburgh, UK, August 29 - September 1, 2005, Proceedings*, volume 3659 of *Lecture Notes in Computer Science*, pages 172–186. Springer, 2005.
12. D. Suzuki and M. Saeki. Security Evaluation of DPA Countermeasures Using Dual-Rail Pre-charge Logic Style. In L. Goubin and M. Matsui, editors, *Cryptographic Hardware and Embedded Systems – CHES 2006, 8th International Workshop, Yokohama, Japan, October 10-13, 2006, Proceedings*, volume 4249 of *Lecture Notes in Computer Science*, pages 255–269. Springer, 2006.
13. D. Suzuki, M. Saeki, and T. Ichikawa. Random Switching Logic: A Countermeasure against DPA based on Transition Probability. Cryptology ePrint Archive (`http://eprint.iacr.org/`), Report 2004/346, 2004.
14. K. Tiri and I. Verbauwhede. A Logic Level Design Methodology for a Secure DPA Resistant ASIC or FPGA Implementation. In *2004 Design, Automation and Test in Europe Conference and Exposition (DATE 2004), 16-20 February 2004, Paris, France*, volume 1, pages 246–251. IEEE Computer Society, 2004.

15. K. Tiri and I. Verbauwhede. Place and Route for Secure Standard Cell Design. In J.-J. Quisquater, P. Paradinas, Y. Deswarte, and A. A. E. Kadam, editors, *Sixth International Conference on Smart Card Research and Advanced Applications (CARDIS '04), 23-26 August 2004, Toulouse, France*, pages 143–158. Kluwer Academic Publishers, August 2004.
16. K. Tiri and I. Verbauwhede. A Digital Design Flow for Secure Integrated Circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 25(7):1197–1208, July 2006. ISSN 0278-0070.