Security Evaluation of Asynchronous Circuits

Jacques J.A. Fournier[1],
Simon Moore[2], Huiyun Li[2], Robert Mullins[2], and George Taylor[2]

[1] Security Technologies Department, Gemplus
Ave des Jujubiers, ZI Athelia IV,
13705 La Ciotat, CEDEX, France.
jacques.fournier@gemplus.com
[2] Computer Laboratory, University of Cambridge.
Simon.Moore@cl.cam.ac.uk

Abstract. Balanced asynchronous circuits have been touted as a superior replacement for conventional synchronous circuits. To assess these claims, we have designed, manufactured and tested an experimental asynchronous smart-card style device. In this paper we describe the tests performed and show that asynchronous circuits can provide better tamper-resistance. However, we have also discovered weaknesses with our test chip, some of which have resulted in new designs, and others which are more fundamental to the asynchronous design approach. This has led us to investigate the novel approach of design-time security analysis rather than rely on post manufacture analysis.

Keywords. Asynchronous circuits, Dual-Rail encoding, Power Analysis, EMA, Fault Analysis, Design-time security evaluation

1 Introduction

The wide-spreading use of processors in security applications, for e.g. in smart-cards or Hardware Security Modules (HSM) has increased both the financial and social benefits that hackers would gain in tampering with such systems. During the past seven years, there has been extensive research to enhance the security of such systems. Most of the counter-measures developed were software-based, protecting mainly against side-channel information leakage. The performance and cost penalties resulting from such counter-measures were affordable. However, software protection against more recently-publicised classes of attacks, like those involving fault injection, consume considerable memory.

There is an urgent need to put more focus on the hardware side of the system. One attractive path is the use of *self-timed* or *asynchronous* circuits. In this respect, we have designed, manufactured and tested an experimental asynchronous smart-card style device. In this paper, we present the principal results of the security analysis of a secure asynchronous processor. We highlight the advantages brought by the *self-timed* nature of the circuit. We also analyse some of the weaknesses that we spotted. Hence, we not only present one of the

industry's first thorough stress-testing of a clockless circuit but also propose an evaluation procedure for post manufacture analysis. Finally we introduce a concept whereby those flaws could have been identified at design level, through thorough simulation leading to what we call *design-time security analysis*.

We finish this introduction by providing motivations for using asynchronous circuits in this field, and give a brief overview of the Springbank test chip (see also [1]) and the experimental set-up used. In Section 2 we provide results for DPA and EMA before describing, in Section 3, the chip's resistance to optical probing and power glitches. Finally, in Section 4, we give an insight of our first results on Design-time analysis.

1.1 Motivation for using asynchronous circuits

Speed independent (SI) asynchronous circuits are expected to offer a number of advantages over their synchronous counterparts when designing secure systems [1, 2]:

Environment tolerance — SI circuits adapt to their environment which means that they should tolerate many forms of fault injection (power glitches, thermal gradients, etc). This makes fault sensing easier since only major faults need to be detected and reacted to. This is desirable since minor fluctuations in environment conditions are normal during real-world operation.

Redundant data encoding — SI circuits typically use a redundant encoding scheme (e.g. dual-rail). In the latter, each bit is encoded onto two wires **A0** & **A1** as shown in the table below. This mechanism also provides a means to encode an *alarm* signal (e.g. use 11 = *alarm* in a dual-rail scheme [1]).

A1	**A0**	**meaning**
0	0	clear
0	1	logical 0
1	0	logical 1
1	1	alarm

Balanced power consumption — Circuits comprising dual-rail (or multi-rail) codes can be balanced to reduce data dependent emissions. In the above illustration whether we have a *logical-0* or a *logical-1*, the encoding of the bit ensures that the data is transmitted and computations are performed with constant Hamming weight. This is important since side-channel analysis is based on the leakage of the Hamming weight of the sensitive data.

Fine-grained random timing variation — may be used to make correlation of repeated runs more difficult, thereby making signal averaging problematic.

Absence of a clock signal — no clock means that clock glitch attacks are removed.

Whilst dual-rail coding might be used in a clocked environment one would have to ensure that combinational circuits were balanced and glitch free. Return-to-zero (RTZ) signalling is also required to ensure data independent power emissions. Once you have gone to these lengths, it is just a small step to an SI

asynchronous implementation which offers the additional benefit of better environment tolerance, i.e. tolerance to fault injection.

1.2 Overview of the Springbank test chip

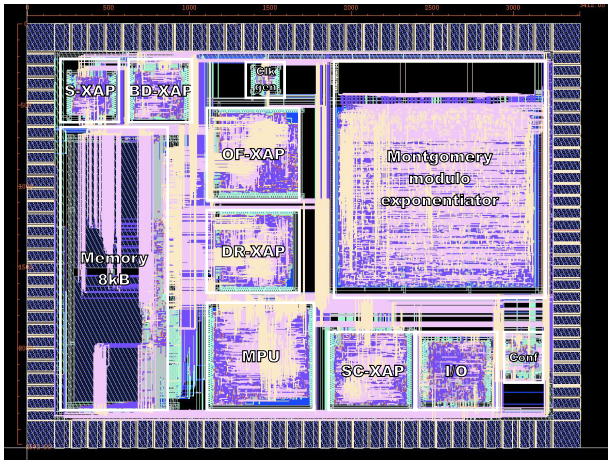

Fig. 1. Springbank Test chip

The Springbank chip was fabricated in the UMC 0.18μm six metal CMOS process. It contains five 16-bit microcontroller processors, various I/O interfaces and other units as part of other projects. All five processors are based on the same 16-bit XAP architecture but with different implementations. The processors are: one synchronous XAP (S-XAP), a bundled data XAP (BD-XAP), 1-of-4 XAP (OF-XAP), 1-of-2 (dual-rail) XAP (DR-XAP) and a secure variant of the dual-rail XAP (SC-XAP). Given that all processors lie on the same chip and that we used the same standard cell library, comparisons do not need to take into account technology or foundry variations. A 1-of-4 distributed interconnect interfaces these processors to a standard single-rail SRAM holding program and data. In addition, communication between the SC-XAP and SRAM is done via a memory protection unit (MPU) with bus encryption; these were disabled in the experiments described in this paper.

Figure 1 shows a picture of the test chip. The SC-XAP is approximately twice the area of the synchronous XAP. However, the commercial standard cell library used was optimised for synchronous design and not asynchronous design. An optimised library might reduce this area penalty to 1.5 times large. Furthermore, one must remember that the clocked system requires a clock generator. Clock multipliers (PLLs), for example, can take up considerable space.

1.3 Experimental set-up

The aim behind these tests is to tally the gain in security while moving from a conventional clocked design (as implemented on the S-XAP) to an asynchronous

dual-rail environment (an example of which is the SC-XAP which bears all the features described in section 1.1). For this reason, tests were mainly carried on the S-XAP and the SC-XAP. Since we were in a 'characterisation' phase, our aim was not to break any cryptographic algorithm. We targeted simple instructions which gave a good indication of how the hardware reacts to the several tests performed. The latter were made on the execution of a simple XOR execution whereby we:

- load the memory address at which data are found,
- load a first operand (Op1) into a register,
- perform an XOR between Op1 and the second operand (Op2),
- store result (Res) back to memory.

To monitor the above execution, after each execution of the above sequence, the three data, i.e. Op1, Op2 and Res, were retrieved via the UART port and displayed on a monitor (or stored into a file). The tests were carried out in a *white box* configuration, without any encryption mechanism activated. This allowed us to thoroughly analyse the benefits and weaknesses of asynchronism.

In the next sections, we describe the results obtained for each family of tests. The interpretation and explanation for those observations are then detailed accordingly.

2 Side Channel Analysis

In this section, we look at the information leakage through two forms of side-channels: one is by studying the power consumed by the processor (Differential Power Analysis - DPA [7]) and the other is by observing the Electro-Magnetic (EM) waves emitted by the processor (Electro-Magnetic Analysis - EMA [8, 9]).

2.1 Differential Power Analysis

Power dissipation in static CMOS circuits is dominated by switching activity. As a result, the power dissipated is highly dependent on the switching activity produced by a change of input data. In the simple case of a bus, activity is observed as the Hamming weight of the state changes. Data-dependent power leakage may be exploited to reveal useful information, either by analysing single power traces (Simple Power Analysis) or by collecting many power traces and performing a statistical analysis of the power variation with respect to changes in data values (Differential Power Analysis [7]).

DPA Attacks on the Springbank Chip Power analysis of the secure dual-rail processor revealed that small imbalances in the design of the dual-rail gates allowed some data-dependent power leakage to be observed. The XOR operation provides one example of where data-dependent power consumption may be observed. Power traces were collected for two different XOR operations, through

experimentation. The operands for the first XOR instruction were 0x11 and 0x22, these were changed to 0x33 and 0x55 for the second.

Figure 2 shows the results of collecting power traces for each operation, averaging the traces over 4000 runs, and then subtracting one averaged trace from the other. The centre curve represents this difference. The small disturbance, left of centre, is the result of data-dependent differences in the power requirements for the two XOR operations.

The same kind of analysis was carried out on the S-XAP and similar data-dependant information leakages were observed. However, the extent of the leakage was more significant in the case of the clocked XAP compared with the asynchronous one. More detailed measurements showed that the data dependant information leakage of the SC-XAP was lower than that of the S-XAP by about 22 dB. This reduction is not sufficient to completely protect against DPA. However, in other cases, we have seen that a reduction in information leakage by 20-24 dB could neutralize leakage with respect to SPA.

Further reducing the data dependant power leakage This example is a good illustration of the difficulty of designing secure processors. On paper, the SC-XAP seemed breachless thanks to its dual-rail with RTZ implementation. However, when it came down to implementing this scheme, conventional place & route tools were used. Those tools tend to optimise space which means that if a bit is encoded onto two 'wires', one wire might end up being longer than the other creating an imbalance which could produce power leakage. So

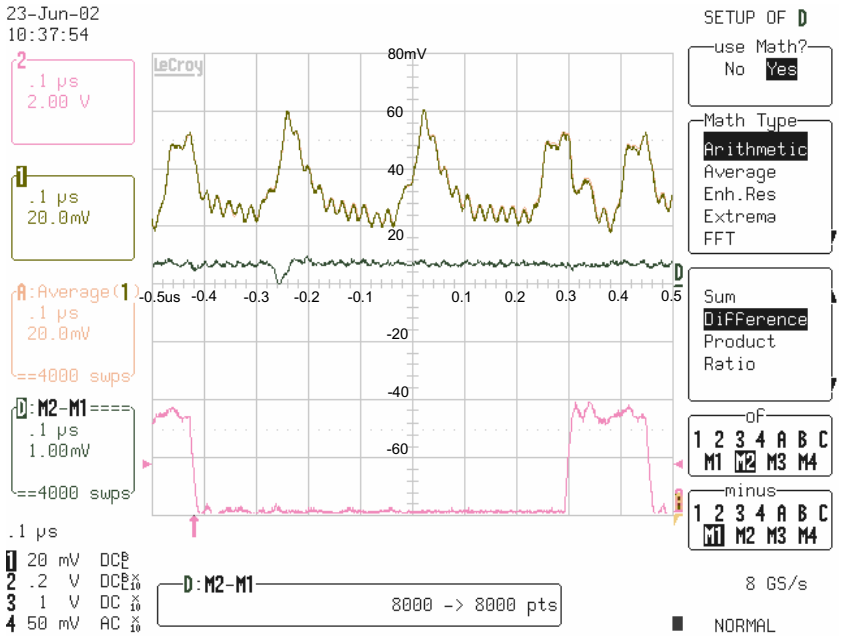

Fig. 2. Differential Power Analysis on Secure XAP (experimental graph)

a first improvement would be to either have a full-custom design or develop a place and route tool which understood how to balance signal paths. Further improvements could be made at a transistor level: current standard cell libraries typically optimise the transistor sizing of gates to minimise the delay through the gate rather than ensuring that the capacitance across all inputs is identical.

2.2 Electro-Magnetic Analysis

In this case, the tests performed were similar to the ones for the DPA, but this time, for each XOR execution, we measured the Electro-Magnetic (EM) waves emitted by the active processor (asynchronous or clocked one) [8]. For the SC-XAP, the EM signals collected were of exploitable magnitudes, which allowed successful DPA-like treatments to be carried out on the collected reference curves. Both for the SC-XAP and S-XAP, data dependant 'signatures' were obtained at three places: at the load of Op1, at the XOR execution and at the write-back of the XOR result into memory. This is illustrated in Figure 3.

The EMA results were taken *without* signal averaging since a signal was clearly visible above the noise. In Figure 3, the uppermost curve is an example of the EM signals measured and the lower three ones correspond to DEMA curves obtained by performing, on the EM curves, differential analysis [8] with respect to *Op1*, *Op2* and *Res*. The 'peaks' shown must be interpreted as leakage of the data's Hamming. We also clearly see at what instances the three different data are 'manipulated'. For the results presented here, a coil covering the processor was used, which means that we were capturing the 'global' EM waves of the entire SC-XAP. However, one could envisage using smaller coils (e.g. 40μm) to measure emissions from a smaller area [8] and target the exact region where the data is being manipulated (the data bus or the ALU for example) .

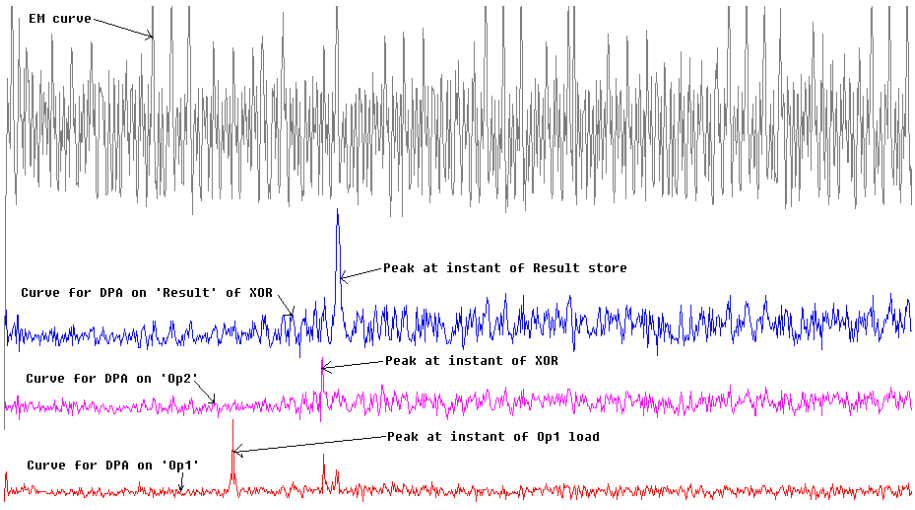

Fig. 3. DEMA Results

EMA leakage on the SC-XAP As for DPA, balanced logic was be used as a countermeasure for EMA. And like in Section 2.1, the imbalance introduced by the design tools used has been lethal. Moreover, with EMA, we did not observe the same 22 dB reduction in the amount of information leaked because EMA is able to isolate much finer circuit areas and hence the placement and routing of components becomes far more critical in achieving a balanced design. In addition to this, the absence of the clock in asynchronous circuits eases EMA. In conventional clocked circuits, the clock usually adds noisy components to the EMA signals captured whereas in asynchronous circuits, we no longer face that inconvenience.

To make the EMA measurements more difficult, a top level metal defence grid may be used. These are seen on modern smart cards and if suitable signals are injected into them they can help mask the underlying activity.

Where operations may be performed in more than one place (e.g. if using a dual execution pipeline), non-determinism may be used to make data collection more difficult. Security evaluation of this approach is most tractable when the attacker is known to have limited resources, for example one EMA trace taken from one sensor. However, when multiple runs and multiple sensors are used the evaluation is far more complex and is dependent on the algorithm being executed. We are also investigating geometrically regular structures (e.g. PLAs) to determine if this approach to design is more secure than a conventional ASIC design flow.

3 Fault Injection Analysis

A second class of stress-testing techniques consists of injecting faults into the device in order to obtain exploitable 'abnormal' behaviours. Injecting faults into working processors can change the nature of data being treated or corrupt cryptographic computations in such a way as to unveil secret information [6].

Early forms of these so called *active* attacks were focused on the device's external interface and often involved introducing glitches on power or clock input pins [10]. Changes in temperature, either by cooling or heating the whole device or the introduction of a temperature gradient, may also be used to induce faulty behaviours. Defences against such attacks are simplified by the restricted nature of the channel by which faults are injected and can easily be detected by incorporating a suitable tamper sensor. Far greater control over the nature of the faults injected has been demonstrated recently.

Two approaches were taken to inject faults into the Springbank: the first one was by optical probing and the second one was by injecting power glitches. As for the previous side-channel analysis, we targeted an XOR operation. This time we worked only on the SC-XAP.

3.1 Optical Probing Techniques

Laser radiation with a sufficiently short wavelength (photon energy) and intensity may be used to ionise semiconductor materials. When ionisation occurs in

a depletion region the production of additional carriers and the presence of an electric field (built-in field and any reverse bias) causes a current to flow. This photocurrent is capable of switching other transistors whose gates are connected to the illuminated junction. This process is a transient one where normal circuit activity resumes once the light source is removed.

In addition to what may be considered a useful attack mechanism, negative effects are also possible. These include the possibility that latch-up may be induced by the generation of photocurrents in the bulk. Of less concern, when using readily available infra-red and visible laser light sources, is the ionisation of gate- and field-oxides due to the large band gap energy of silicon dioxide (which would require a laser with a wavelength in the UV-C range). Ionisation of this type is common when higher energy forms of radiation are absorbed. The subsequent accumulation of positive charges results in a long term shift in transistor characteristics. The following sections explore the weaknesses of the dual-rail technology employed in the Springbank test chip. We then introduce a number of improvements that could secure the design against such attacks.

Optical Probing Attacks on the Springbank Chip If the dual-rail implementation had provided a completely fault-secure design all attempts at inducing faults would have resulted in deadlock. In many cases the processor did propagate an error signal resulting in deadlock. Unfortunately, two weaknesses in the current design were revealed by the experiments.

The first involved the injection of faults into the ALU design. By targeting two different regions within the ALU two different fault behaviours were possible. The first was to disrupt the ALU operation to produce an incorrect result, the second forced the ALU to always return the result `0x0001`. These results are possible as some of the dual-rail gates within the ALU do not guarantee that the presence of the error state on their inputs (in this case a *logic-1* on both dual-rail wires) is propagated. This was a known and unfortunate concession made at the design stage.

Perhaps more interesting is the second failure behaviour. In this case it was possible to set the contents of the processor's registers. The exposure of a single register cell to laser light reliably resulted in setting its value to a *logic-1*. The dual-rail register design that was used in the Springbank chip is illustrated in Figure 4. Setting the cell to '1' was made possible by its inability to store an error state (both states of the single flip-flop are valid). The precise mechanism by which the register was set first involved both the outputs of the `NOR` gates being pulled-low. This happened as a result of the laser producing photocurrents in the junctions of the N-type transistors in both gates. When the laser was removed, the flip-flop resolved into the *logic-1* state due to differences in the threshold values for each gate. N-type transistors in general produce much larger photocurrents due in part to the superior mobility of electrons when compared to holes (*The electrons and holes in this case are the minority carriers on the larger side of the depletion region. The depletion region extends mostly into the*

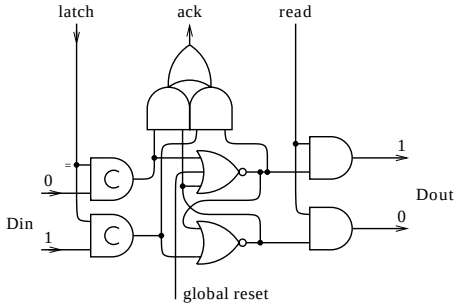

Fig. 4. Single Flip-Flop Dual-Rail Register

region of least doping.). It is important to note that the attack was successful even with a large spot size exposing many transistors (we estimate around 100).

Optical Probing Countermeasures The vulnerability of the ALU design may be countered by ensuring that an error state on any gate input is always propagated to its output. An approach to providing a secure dual-rail register design is shown in Figure 5. Here the number of flip-flops has been doubled. The four possible states are now split into two valid states (representing *zero-symbol* and *one-symbol*) and two error states (null encoded as 00 and error as 11). When the register is reset it is forced into the error state, this prevents the possibility that the reset signal may be used as a simple way to reset the contents of a register to a valid state (perhaps by targeting a reset signal buffer). The error state will only be propagated on the register's output if the register is read. For correct operation, the register must be written with a valid data value prior to reading. The register is also designed to produce an error signal if a 'null' state is ever stored. The ability to store a null value may assist an attacker by allowing them to inject an actual data value from another source.

We will now consider a number of different attack scenarios and how the design is able to detect the injected faults.

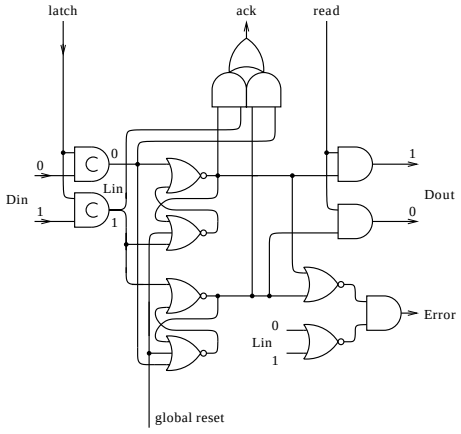

Fig. 5. Dual Flip-Flop Dual-Rail Register

Initially we consider an attack similar to the one described above. Here a large number of transistors are exposed and force all the gate outputs to a *logic-0*. By guaranteeing that both flip-flops resolve in the same direction, the resulting state of the register will always be one that represents a fault (null or error). Attempts to target and modify a single flip-flop will again always result in a fault state. A successful attack now requires greater control over the fault injection process. For example, if both flip-flops could be exposed and then the laser spot moved up away from the lowermost `NOR` gate, a valid data value could be written. Independent and simultaneous control over individual transistors also offers the possibility of setting registers to particular values.

Security may be further improved by including small optical tamper sensors within each standard cell. These sensors, constructed from one or two transistors, would normally play no part in normal circuit behaviour (only adding a small amount of capacitance). Their only function is to force the dual-rail outputs of the gate into an error state when illuminated. A similar approach is already taken in many standard cell libraries to protect against plasma-induced oxide damage during manufacture, in this case an antenna diode is added on every gate input. These ideas together with security-driven place-and-route would again increase the level of controllability required to perform a successful modification of register values.

3.2 Power Glitch Attacks

Power glitches may be used to inject faults at a coarse level. Tests on the SC-XAP revealed that it was resistant to short Vcc glitches which went down to the ground rail and back. For longer duration glitches we observed faulty processor behaviours which could constitute favourable conditions for the cryptanalysis of cryptographic algorithms like the DES or RSA [5, 6]. By injecting the power glitch at different times, we succeeded in causing specific parts of our small program to malfunction. The interesting thing to note is that if we want to target, say, the load of Op1 instruction, we synchronize our program so as to 'cut' the power just at that instant and resume it several tens of nanoseconds later in such a way that the normal program execution resumes. The effects of the glitch are monitored through the power consumed by the processor, just like for the Differential Power Analysis. This is illustrated in Figure 6 which is a superposition of the power traces: one in the normal mode and one where we introduced the power glitch.

If we synchronize the curves and zoom in as shown in Figure 7, we see that the real impact is on the LOAD of Op1 execution. In this case, the value read as Op1 was always 0xFFFF. Consequently, the result of the XOR operation was always the logical inverse of Op2. In this case, we have targeted one precise instruction and corrupted an entire data. This scenario could be lethal if we were to attack the load of a DES key for example.

In another experiment, we generated the glitch while uploading the data's address into the address register. This led to the 'writing' of an erroneous address.

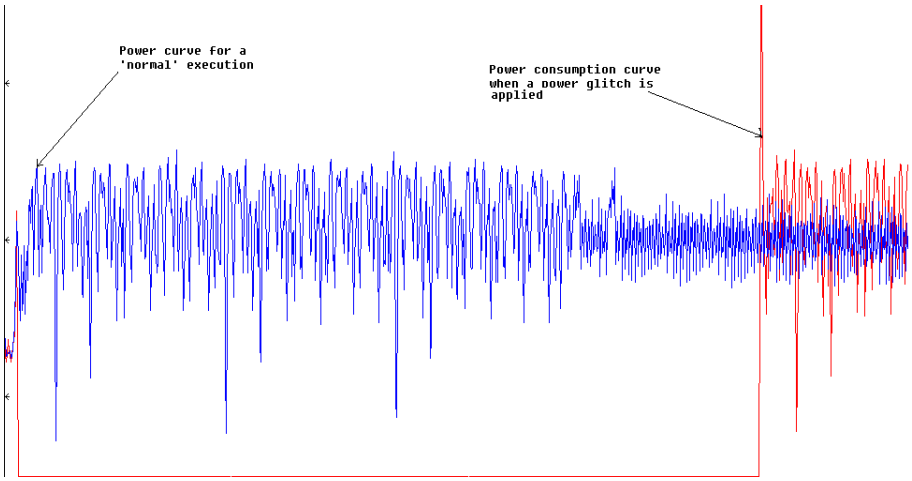

Fig. 6. Vcc Glitch on Secure XAP

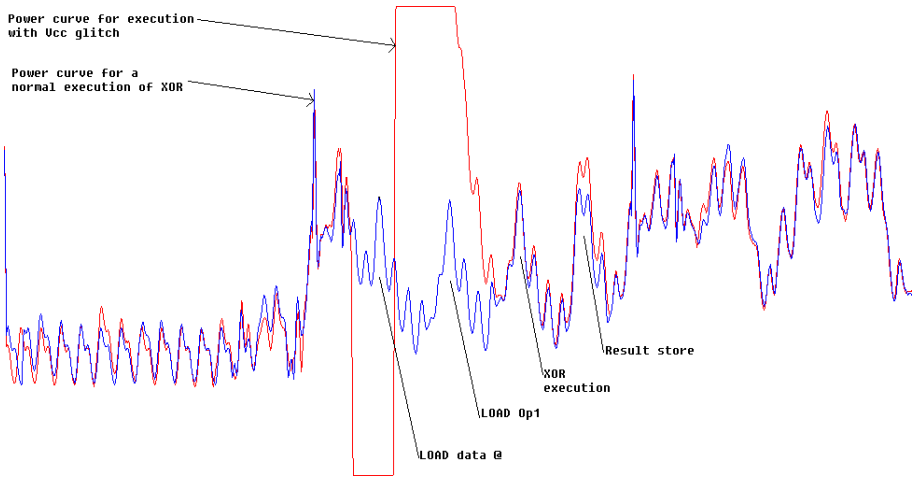

Fig. 7. Vcc glitch during LOAD Op1

Those are a few examples of how 'long' glitches can corrupt the functioning of a self-timed system. We are currently looking into this aspect.

4 Design-time Security Analysis

We have seen that in many ways, the SC-XAP exhibits several interesting security properties like lower data-dependant power leakage and resistance to optical probing for most of the processor. We did not predict that the asynchronous nature of the circuit could facilitate attacks like EMA and voltage glitches. Moreover, other security flaws, like the low resistance of the ALU and the register bank to optical probing, are linked to unfortunate design trade-offs in the SC-XAP.

The design and evaluation of the Springbank test chip is typical of the design process for secure processors. We began with a requirements specification which included security properties. This allowed us to identify key design criteria which steered the design process. However, we lacked design time validation of the security criteria and we now know that some side cases were overlooked. Even more worryingly, our colleagues working on attack technologies developed new attacks which we had not even considered during the design process. What we seem to have recreated in our research project was a microcosm of current industrial practice.

Dissatisfied with ad hoc evaluation post design, we have begun a research programme to investigate design time security validation techniques. In the last section of this paper, we give an insight of the on-going work about *Design-time Analysis* which is bound to become important for the future design of secure processors. Design-time analysis is performed during the design process whereby we should try to simulate the behavior of the processor along with the current consumed and the energy radiated.

4.1 Simulating side-channel information leakage

To confirm the source of the imbalance observed during the side-channel analysis, we simulated the operation of the ALU executing the same XOR operations described in Section 2.1. The power simulation results were collected using Primepower™ [4], a gate level power estimation tool. The power estimation includes capacitance and resistance values extracted from layout. The results of the simulation are shown in Figure 8. Even in the absence of a power model for the memory system we observe a similar data-dependent power difference during the execution of the XOR operations. Using a simple second order low pass filter model for the power distribution network provides more comparable data (Figure 9). The power dissipation curves for the XOR operations differ in shape to those measured as they include no power for memory accesses. This produces a significant drop in power at the point where one XOR operand is fetched from memory.

We hence see how data dependant leakage may be detected at design time via systematic simulation. Such simulations allow design comparisons to be made, though it is harder to predict the exact values of emissions. The simulations we have undertaken for power are based upon switching activity. In this case, capacitance masks some of the information. Similarly, for electromagnetic radiation, one has to consider wave interference. None the less, switching activity simulation gives a good approximation to the energy being consumed over time which is a good approximation for DPA and DEMA.

4.2 Design-time analysis of fault tolerance

A range of physical phenomena that can trigger faults may be modelled. We can then model a wide range of attack scenarios from single to multiple transistor failures. Given bounds on the control the attacker has, we can determine whether

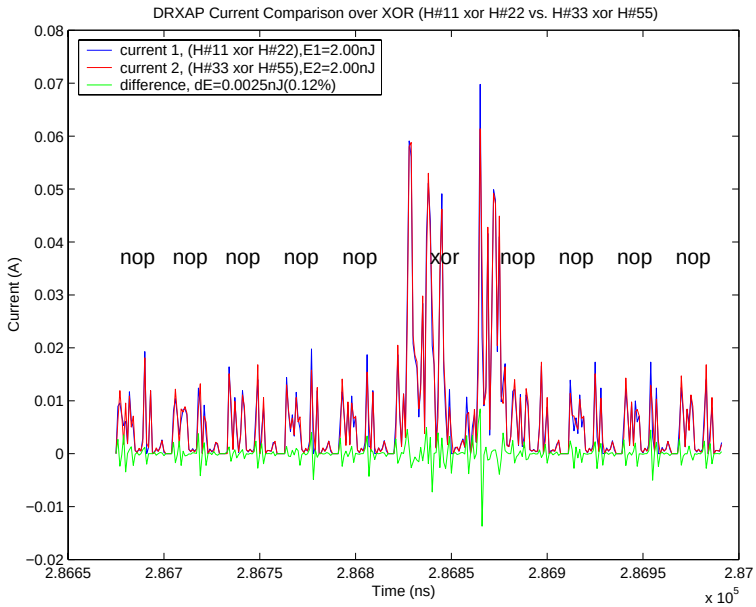

Fig. 8. Power Simulation. Secure XAP executing XOR

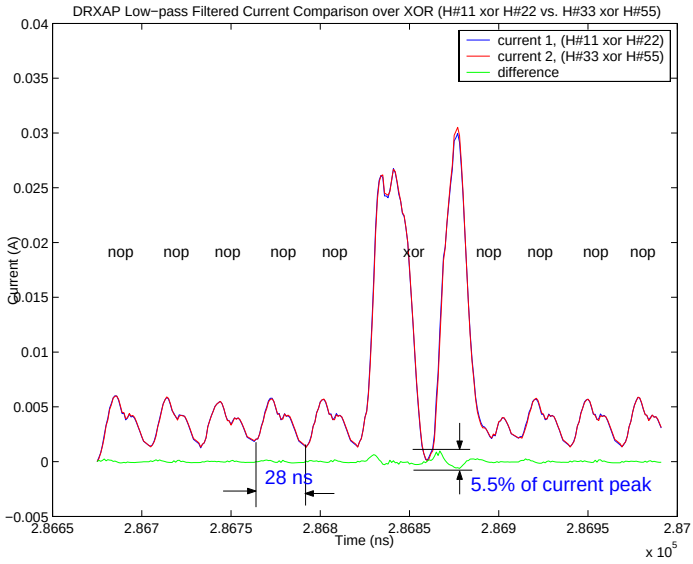

Fig. 9. Power Simulation. Secure XAP executing XOR, low pass filter applied

a fault can be injected without being detected. No matter what the source of the fault is, the end result is to somehow modify the data being manipulated. The aim of the game is to detect any attempt to cause bit flips and this is being investigated right now: had we identified, at simulation level, that no alarm signal was propagated when the ALU or the registers were tampered with, we would

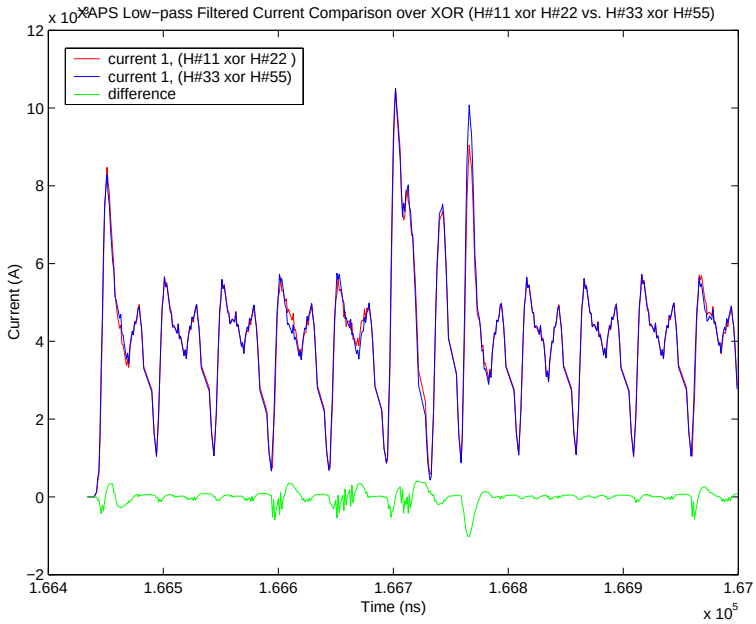

Fig. 10. Power Simulation. Synchronous XAP executing XOR, low pass filtered

have redesigned those weak parts of the circuit. Systematic testing of faults can then be undertaken at a small module level through exhaustive simulation. Such simulations can take into account a wide range of conditions (e.g. single and multiple transistor failure induced by optical probing) in much the same way that traditional fault simulation is undertaken. Where alarms are generated at the small module's level, it is then possible to reason about the propagation of alarm signals at a more abstract level for larger systems.

5 Conclusion

This paper has presented the first ever security evaluation of an asynchronous smart-card system. The secure asynchronous processor (SC-XAP) has shown interesting tamper-resistance properties. None the less we have identified weaknesses in our first attempt together with possible refinements to overcome these issues.

Asynchronous circuits could become a trustworthy platform for secure computing. Circuit area is inevitably going to be larger than a simple synchronous design, but this has to be balanced against large memory (and thus chip area) savings that are possible if fewer software countermeasures are required. The lack of ECAD tool support for asynchronous circuit design is another issue, though we were able to make use of commercial place & route tools, standard cell libraries, etc, and we were able to complete the design with just a small research team.

Finally, we mention the concept of design-time security analysis. These techniques are centered around the simulation of a wide range of measurements and

fault possibilities for a wide range (preferably exhaustive) set of input data. We have demonstrated that power attacks and optical probing can be simulated. However, our longer term goal is to be able to make more general statements about the level of security attained which go far beyond current known attacks. With such an approach, we believe that security by design may become a far more powerful technique for processor designers.

References

[1] S. Moore, R. Anderson, P. Cunningham, R. Mullins and G. Taylor, "Improving Smart Card Security using Self-timed Circuits," in *Proc. 8th IEEE International Symposium on Asynchronous Circuits and Systems – ASYNC '02*, pp. 23–58, IEEE 2002.

[2] L.A. Plana, P.A. Riocreux, W.J. Bainbridge, A. Bardsley, J.D. Garside and S. Temple, "SPA - a synthesisable amulet core for smartcard applications," in *Proc. 8th IEEE International Symposium on Asynchronous Circuits and Systems – ASYNC '02*, pp. 201–210, IEEE 2002.

[3] R. Mayer-Sommer, "Smartly Analyzing the Simplicity and the Power of Simple Power Analysis on Smartcards," in *Proc. 2nd International Workshop Cryptographic Hardware and Embedded Systems – CHES '2000*, LNCS 1965, pp. 78, 2000.

[4] http://www.synopsys.com/products/power/primepower_ds.html

[5] D. Boneh, R.A. DeMillo and R.J. Lipton "On the importance of checking cryptographic protocols for faults," in *Proc. Advances in Cryptology – EUROCRYPT '97*, LNCS, pp. 274-285, 1994.

[6] E. Biham, and A. Shamir, "Differential fault analysis of secret key cryptosystems," in *Proc. 17th International Advances in Cryptology Conference – CRYPTO '97*, LNCS 1294, pp. 513–525, 1997.

[7] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Proc. 19th International Advances in Cryptology Conference – CRYPTO '99*, pp. 388–397, 1999.

[8] K. Gandolfi, C. Mourtel, and F. Olivier, "Electromagnetic analyis: Concrete results," in *Cryptographic Hardware and Embedded Systmes (CHES 2001)*, LNCS 2162, 2001.

[9] J.-J. Quisquater and D. Samyde, "ElectroMagnetic analysis EMA: Measures and coutermeasures for smart cards," in *Smart Card Programming and Security*, pp. 200–210, LNCS 2140, 2001.

[10] R. Anderson and M. Kuhn, "Tamper resistance – a cautionary note," in *Second USENIX Workshop on Electronic Commerce*, (Oakland, California), pp. 1–11, USENIX, Nov. 1996.

Acknowledgements

The authors would like to thank Sergei Skorobogatov and Ross Anderson (University of Cambridge) provided useful insights into attack techniques. Special thanks are also due to the Gemplus Card Security Group for their testing and analysis of the Springbank test chip, in particular to Jean-François Dhem and Christophe Mourtel for useful comments on this paper. We thank the European Commission for funding some of this research as part of the G3Card project (IST-1999-13515).