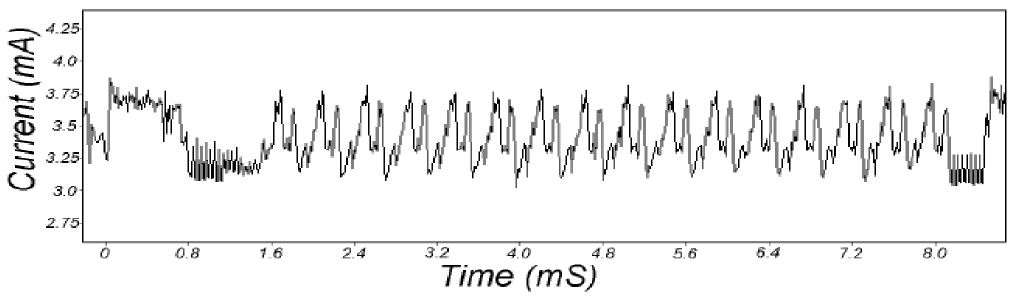
A random permutation based method for secure hardware implementations

Robin Schrijvers

Master IW Elektronica-ICT

Introduction

Problem : Side Channel Attacks



Differential power analysis:
Patterns of data flow can be recognized

=> Derive secret keys in FPGA, but especially in ASIC

Solution: Randomizing measured data via random permutation

Test platform: S-BOX in ASIC

PRESENT cipher S-BOX:

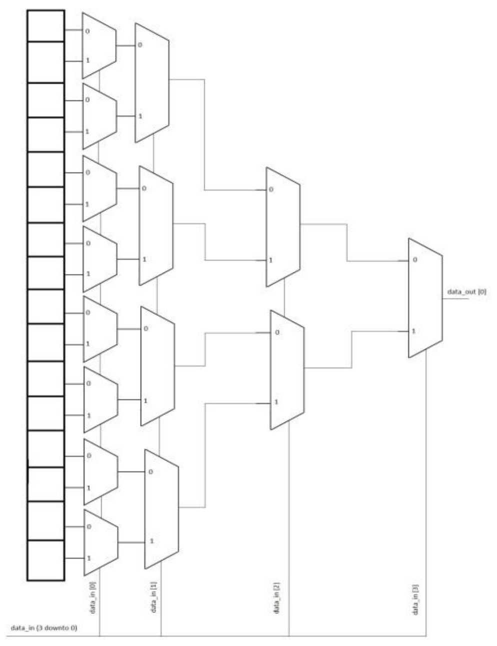
x	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
$S[x]$	C	5	6	B	9	0	A	D	3	E	F	8	4	7	1	2

Problem: implementing S-BOX in ASIC

Methods and implementations

4x4 S-BOX in ASIC

4 times



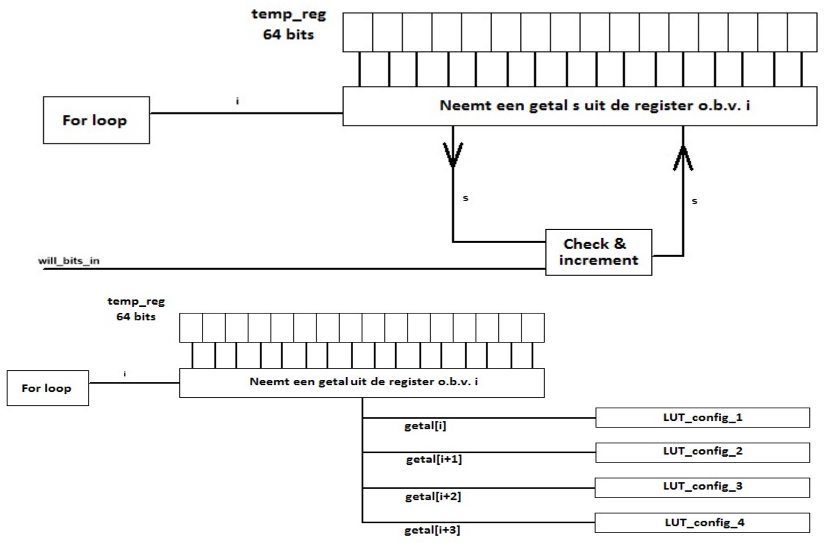
- Configurable LUT from FPGA to ASIC
- Registers are configurable
  ⇒ Permutation generation
- NanGateOpenCell library

Area on ASIC: **4063,7µm²**

Random permutation

```
Array perm_array, numbers;
for (int i = 0; i < 16 ; i++) {
        int x = numbers[i];

        for (int j = 0; j< 16 ; j++) {
                y = perm_array[j];
                if (x >= y) {
                        y = y + 1;
                        perm_array[j] = y;
                }
        }
        perm_array.shift();
}
```



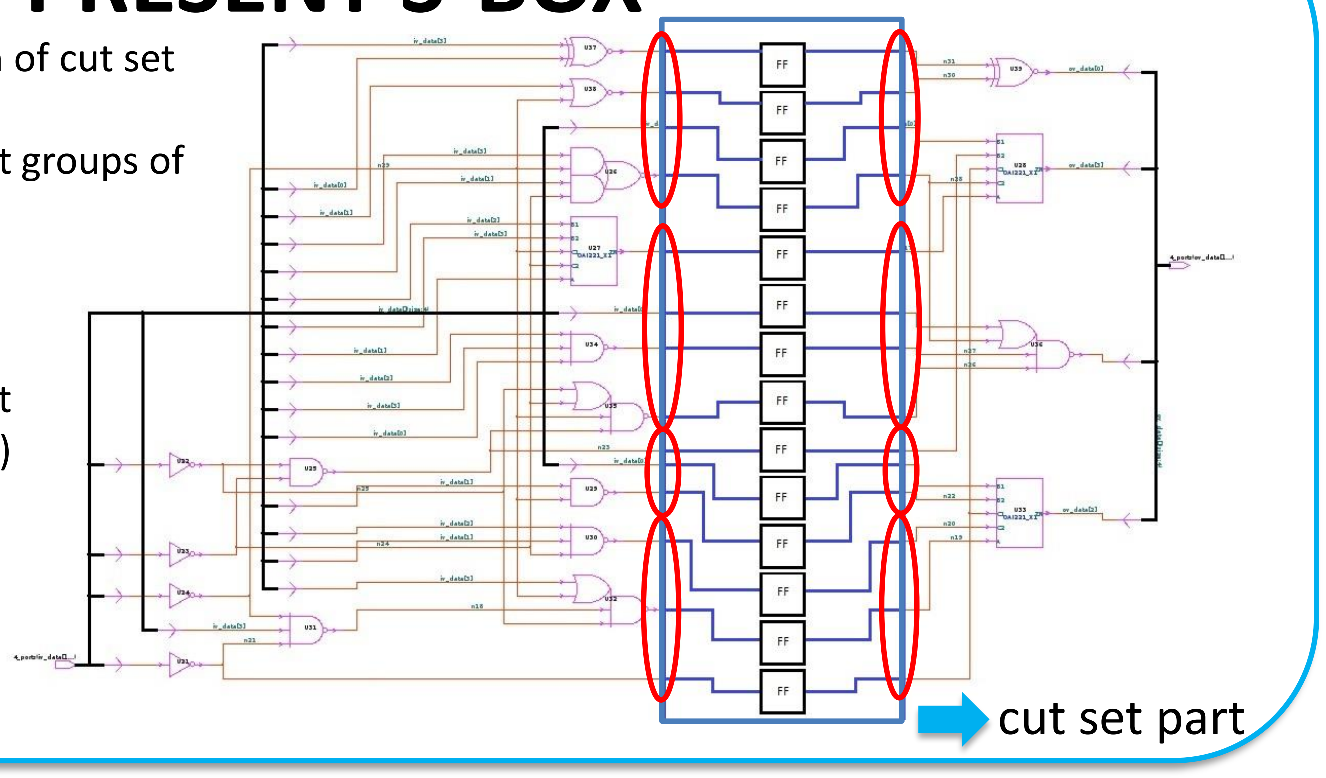
Inverse permutation based on addressing

PRESENT S-BOX

- Stored data in FF's = permutation of cut set input
  - Permutation of input groups of cut set
- Multiplexers decide which permutation will be selected
  - Data group of cut set = SELECT (3 or 4 bits)

Area of new S-BOX design:
7909,2 µm²

Maximum clock frequency:
1,72 GHz



cut set part

Conclusions

- Copied LUT's from FPGA to ASIC
  - Cascade of multiplexers
  - **4063,7 µm²**
- Dynamic use of S-BOX with permutation generation possible
- Randomized the data on intersections where attackers perform DPA's
  - Stored random permutation of input groups in FF's
  - Cut set: **7909,2,2 µm²**
  - Maximum clock frequency: **1,72 GHz**
- Further research:
  - Improve speed and area
  - Copy more fields of FPGA to ASIC with NanGateOpenCell library

Promotoren / Copromotoren: Prof. dr. ir. Nele Mentens
Bohan Yang