Chapter 7

HIDING

Power analysis attacks work because the power consumption of cryptographic devices depends on intermediate values of the executed cryptographic algorithms. Therefore, the goal of countermeasures is to avoid or at least to reduce these dependencies. In case of hiding, this is done by breaking the link between the power consumption of the devices and the processed data values. Hence, cryptographic devices that are protected by hiding execute cryptographic algorithms in the same way as unprotected devices. In particular, they calculate the same intermediate values. Yet, the hiding countermeasures make it difficult for an attacker to find exploitable information in power traces.

In this chapter, we first provide a general description of hiding. Subsequently, we analyze how hiding can be implemented at the architecture level of cryptographic devices. In particular, we discuss examples of countermeasures for software and for hardware implementations of cryptographic algorithms. Finally, we analyze hiding at the cell level. We discuss different logic styles that have been proposed to counteract power analysis attacks.

7.1 General Description

The goal of hiding countermeasures is to make the power consumption of cryptographic devices independent of the intermediate values and independent of the operations that are performed. There are essentially two approaches to achieve this independence. The first approach is to build devices in such a way that the power consumption is random. This means that in each clock cycle a random amount of power is consumed. The second approach is build devices that consume an equal amount of power for all operations and for all data values. Hence, equal amounts of power are consumed in each clock cycle.

The power consumption of a cryptographic device is independent of the performed operations and the processed data values, if it has one of the following two properties:

- The device consumes random amounts of power in each clock cycle.
- The device consumes equal amounts of power in each clock cycle.

The ideal goal of making the power consumption perfectly random or equal cannot be reached in practice. However, there are several proposals on how to get close to this goal. These proposals can be divided into two groups. The first group of proposals randomizes the power consumption by performing the operations of the executed cryptographic algorithms at different moments of time during each execution. These proposals hence only affect the time dimension of the power consumption. This is different in case of the second group. The goal of these proposals is to make the power consumption random or equal by directly changing the power consumption characteristics of the performed operations. The second group of proposals hence affects the amplitude dimension of the power consumption.

7.1.1 Time Dimension

In step 2 of the description of DPA attacks (see Section 6.1), we have pointed out that the recorded power traces should be correctly aligned for these attacks. This means that the power consumption of each operation should be located at the same position in each power trace. If this condition is not fulfilled, DPA attacks require significantly more power traces. This observation is the motivation for designers of cryptographic devices to randomize the execution of the cryptographic algorithms, *i.e.* the devices perform the operations of the algorithms at different moments of time during each execution. This makes the power consumption appear to be more or less random for an attacker. The more random the execution of an algorithm is, the more difficult it becomes to attack the device. The most commonly used techniques to randomize the execution of cryptographic algorithms are the random insertion of dummy operations and the shuffling of operations.

The power consumption a cryptographic device can be randomized by performing the operations of the executed algorithm at different moments of time during each execution. This can be done by randomly inserting dummy operations or by shuffling.