抵御实现攻击的软件方法实验

密码攻击

- 密码攻击可分为**数学攻击、实体攻击、实现攻击**
- 如何防御实体攻击
    - 利用芯片的唯一数据进行加密
        - ID号
        - 频率校正表
- 实现攻击
    - 优势
        - 无需打开硬件封装，是相对廉价和有效的攻击方式
        - 与数学分析相比，具有较小的密钥搜索空间和较高的分析效率
        - 可分为**主动式攻击(失效分析攻击)和被动式攻击(侧信道攻击)**
    - 主动式攻击
        - **引入故意错误**如电源或时钟突变来影响CPU的执行过程
        - 常见的干扰方法有：时钟短暂扰动、电源短暂扰动等
    - 如何抵御主动式攻击
        - 前序代码检查
        - 使用内部时钟源
        - 使用内部复位(屏蔽外部复位)、低电压复位、WDT
        - **冗余循环次数判断**
            - 判断循环次数是否复合预期
                - `i == i' == target`
        - **写入数据后校验**
            - 确保数据被写入(读出来校验)
        - 使用软硬件复合RNG
            - 不要完全依赖于硬件
                - 硬件不够可靠
            - 如果不随机则可能存在重放攻击
        - SLEEP躲避干扰和攻击
        - **多次计算**
            - 对于重要的过程做多次计算
    - 被动式攻击
        - 常见的有电磁、能量、时间攻击
        - 特点
            - 有足够多的采样样本
            - 各密钥相关状态的准确采样值
        - 如何防御(立足点在于其特点)
            - 原理
                - 消除密码算法实现中信息的泄露
                    - 避免时间泄露
                        - 做完所有的比较再退出(不提前退出)
                - 增加攻击的难度
                    - 增加噪声
                        - 增加无用的信息

                - 减少有效信息量
                    - 多用寄存器和cache
                    - 用DMA做信息传递
                        - DMA的配置可变
        - 数据冗余
            - 使用数据单元(内容+校验码)
            - 解密使用，用完销毁
            - **序代码增加校验码(校验码应该和芯片唯一性数据相关)**
        - 控制冗余(状态机)
            - 设置**多重状态位**，在陈旭的执行时检查状态位，如果状态位发生了改变，程序马上退出
            - 状态转移的条件审查
            - 重要模块入口、出口检查
            - 执行序列检查
        - 执行冗余
            - 没有先后关系的代码随机顺序执行
            - 同一功能构造多个等价实现，随机选择执行
            - 随机延时和随机功耗
                - 在剩余时间延时
                - 随机启用未用硬件
            - 数据单元的等价实现
                - 不是备份
                - 利用在某个函数中若要用到变量a，则在$a_1、a_2、a_3$中进行随机选择
            - 使用泄露信息比较少的操作
                - 寄存器、cache、DMA、嵌入式汇编
            - 使用固定运行路径的代码
- **==随机的粒度如何设置==**
    - 在不同的时间粒度上随机，有长有短