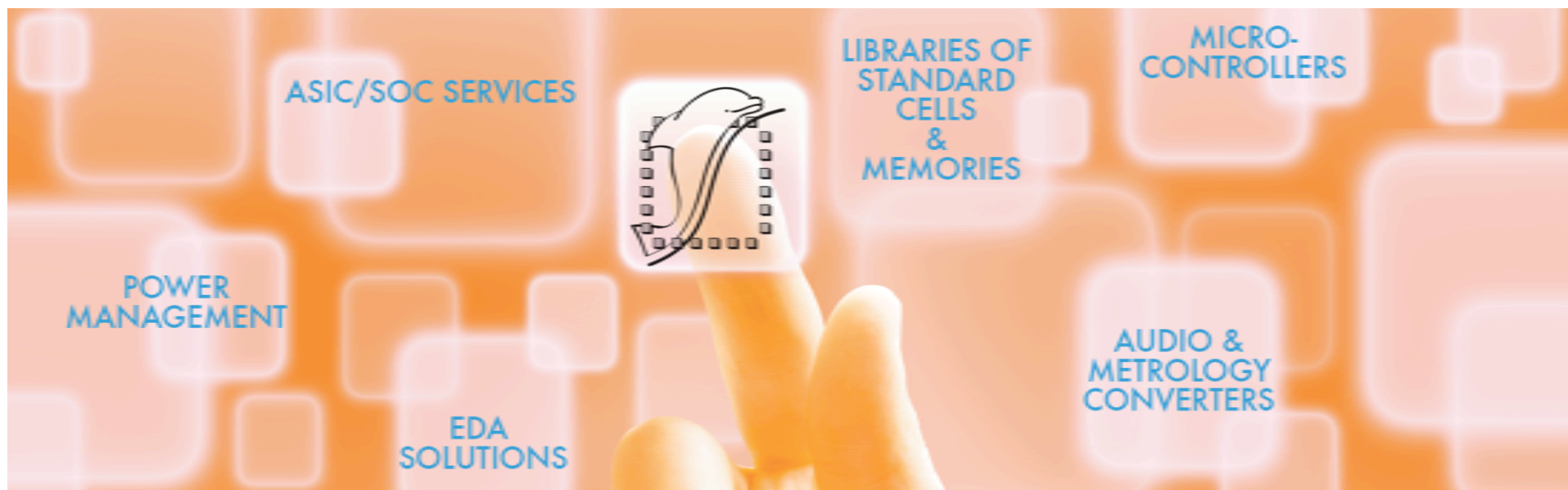


针对物联网**80x51**升级的未来

赵盈

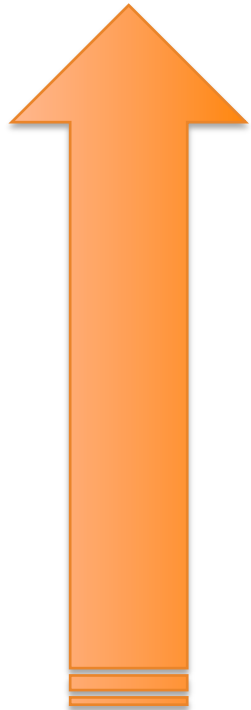
2014年12月 中国香港ICCAD



1. 物联网带给微处理器核的挑战
2. 新兴一代i351架构
 1. 拥有32位RISC性能的升级版80x51
 2. 低功耗优化
 3. 子系统面积优化
3. 创新软件开发工具
4. 日渐增长的安全对策需求
5. 80x51微控制器的光明未来



处理能力要求



高端需求
32-64 位应用程序处理器
Linux 或 Android 安卓操作系统



中端需求
16-32 位 MCU
实时操作系统(RTOS)或JAVA OS



比典型的**8051**还要高的处理能力，但不需要程序处理器

初级要求
8-16 位 MCU
专用程序且无需操作系统

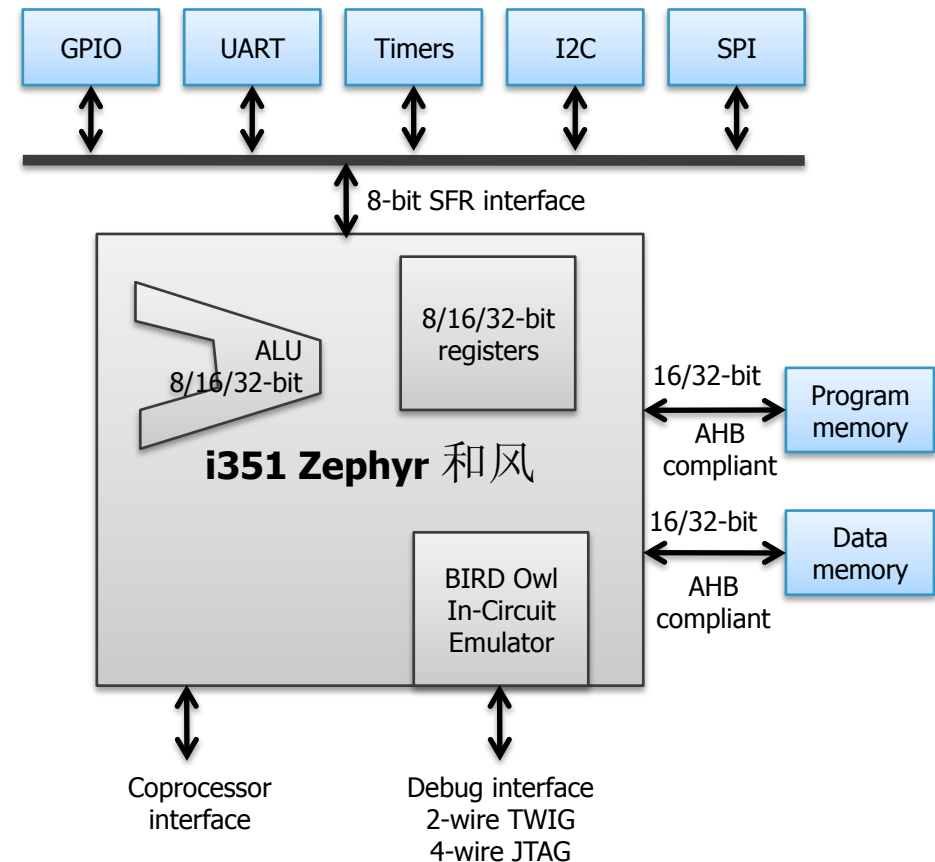


从典型**8051**时代演变到高安全对策要求



1. 物联网带给微处理器核的挑战
2. 新兴一代**i351**架构
 1. 拥有32位RISC性能的升级版80x51
 2. 低功耗优化
 3. 子系统面积优化
3. 创新软件开发工具
4. 日渐增长的安全对策需求
5. 80x51微控制器的光明未来

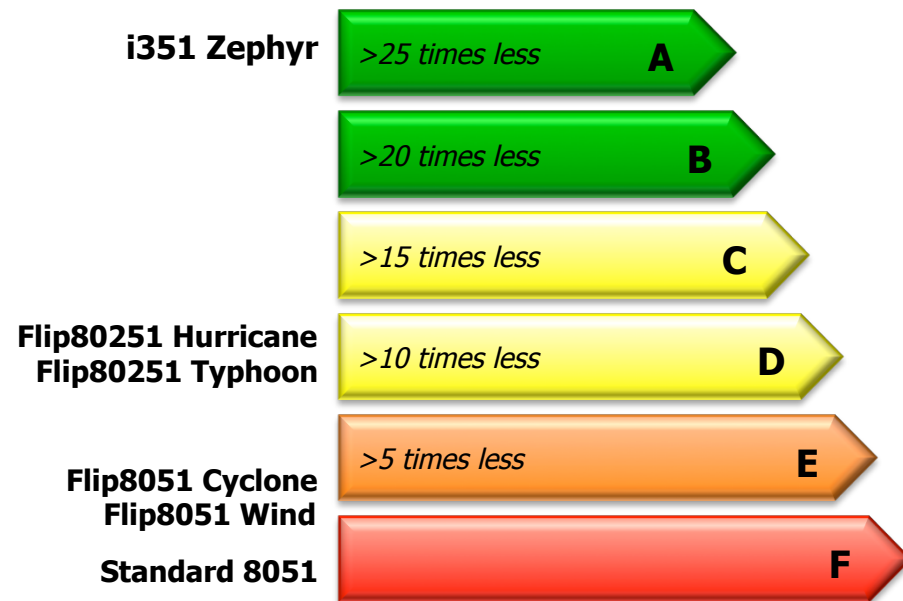
- 现有**80x51**应用的轻松迁移
- 针对子系统级功耗与面积优化的创新架构
- 利用专用指令及足够小的数据路径来独一无二地灵活处理**8, 16或32**位字
 - ➔ 日渐增长的连通功能 (**USB, 蓝牙...**)及先进的模拟感应器的要求
 - ➔ 高代码密度
- 存储器映射功能增强到支持**4GB**的线型映射
- 管线 (**pipeline**)架构优化后大多数的指令都可以在一个周期下完成执行
 - ➔ 处理能力: **1 DMIPS/MHz**



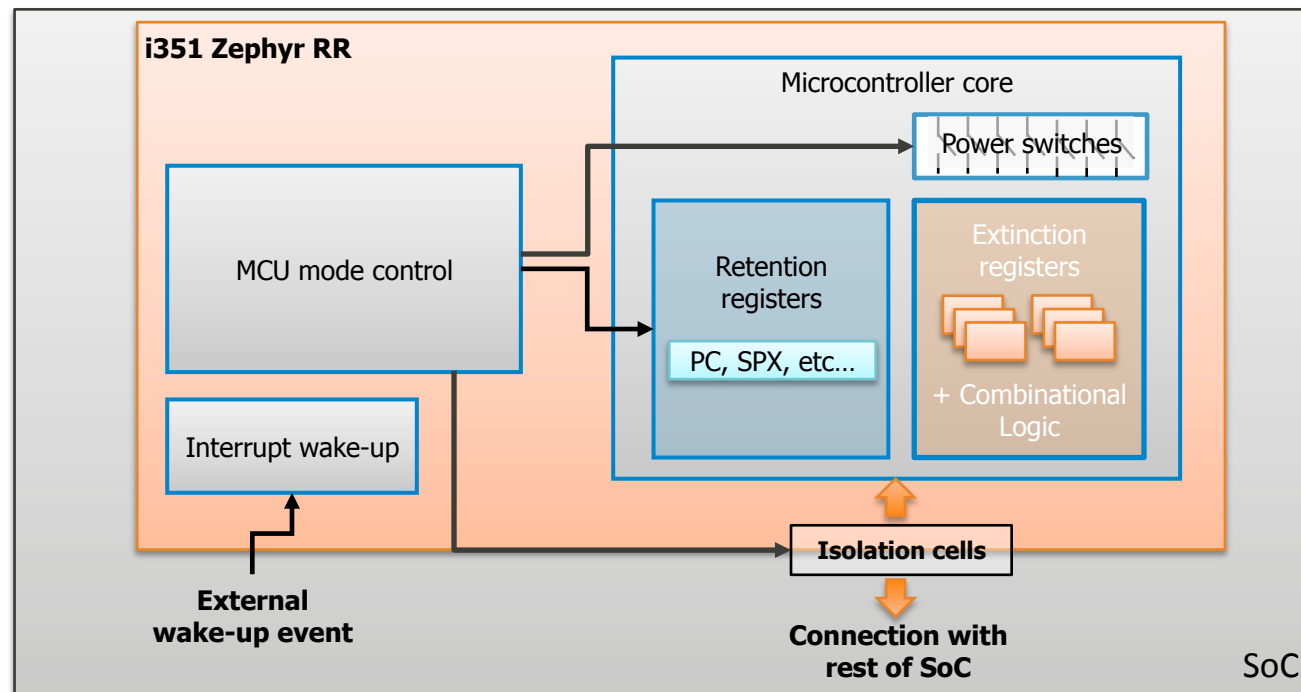
*Dhrystone v2.1 / 10.000 loops

- 低深度管线(pipeline)
 - ➔ 可在管线冲刷情况下(pipeline flush)减少不必要从存储器取数据的次数
- 可变大小的指令字 (instruction word)
 - ➔ 减少代码大小
 - ➔ 减低取数据次数
- 执行操作下会自动配合数据类型及大小
- 用最小序列的指令来处理一个复杂的任务
 - ➔ 减低操作频率
 - ➔ 让处理器多停留在睡眠模式

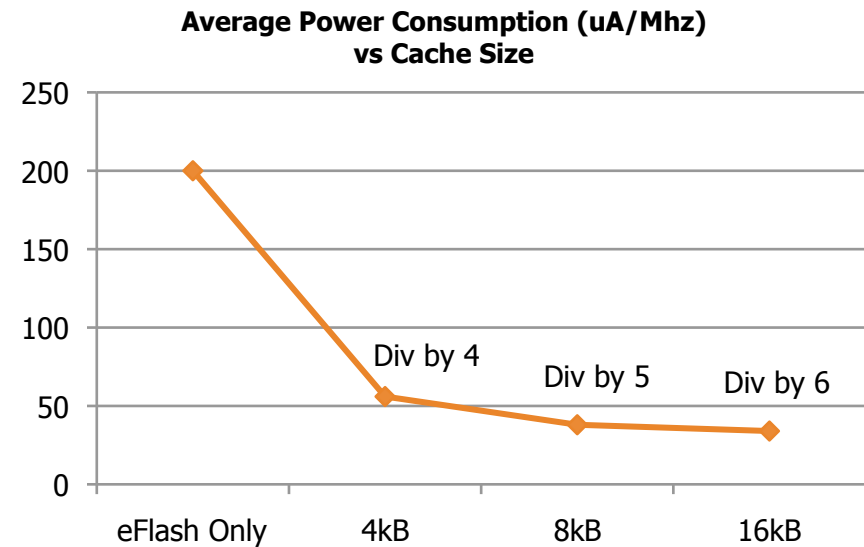
功耗级别



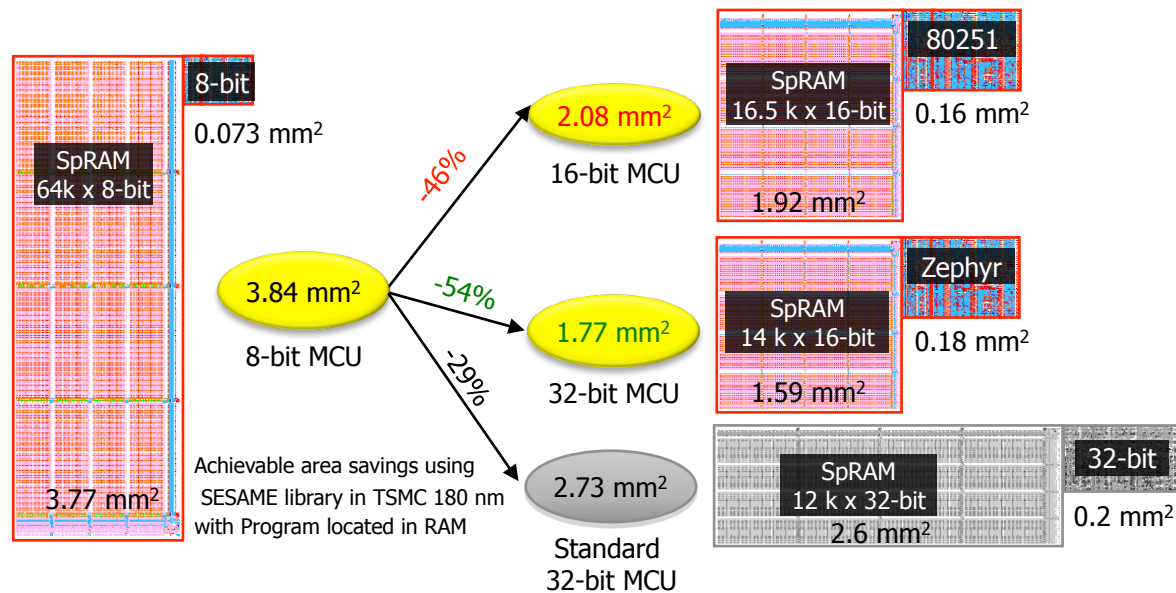
- 边缘物体大部分的时间（可到**99.5%**的使用期限）都在待机 / 睡眠模式上
- 通用功耗模式
- + “及时保留Retention Ready” 模式
 - ➔ 关掉微架构内核下， 能够用最小的存储器保留执行环境
 - ➔ 功耗可小至**10**倍
 - ➔ 快速关闭及唤醒时间 ($< 1\mu\text{s}$)



- 低功耗缓存微控制器(Cache Controller)
 - ➔ 功耗小到6倍以上
 - ➔ 速度快到3倍以上
- 直接存储器存取 (DMA) 控制器
 - ➔ 处理多种传输操作，并让处理器及其程序存储器停留在睡眠模式
- 协处理器接口
 - ➔ 增加客制及专用指令来执行特别程序（例如加密，数据处理...）



- 减低代码大小是至关重要的步骤来达到减小面积及节省设备成本
 - 可变大小的指令制实现超高代码密度
 - ➔ 高大小粒度的指令可确保每个指令都用最小位编码
 - ➔ 避免在代码存储器里存上无必要的信息
- => 可采用例如Flash闪存般的较小非易失性存储器

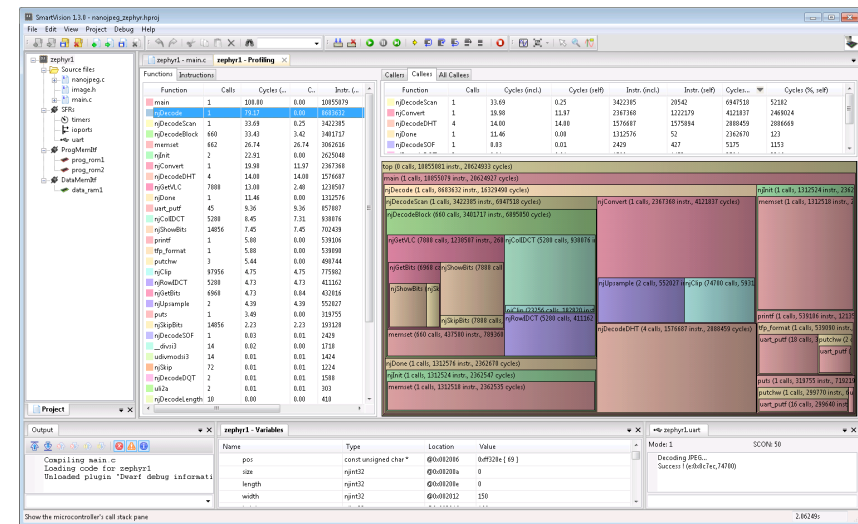
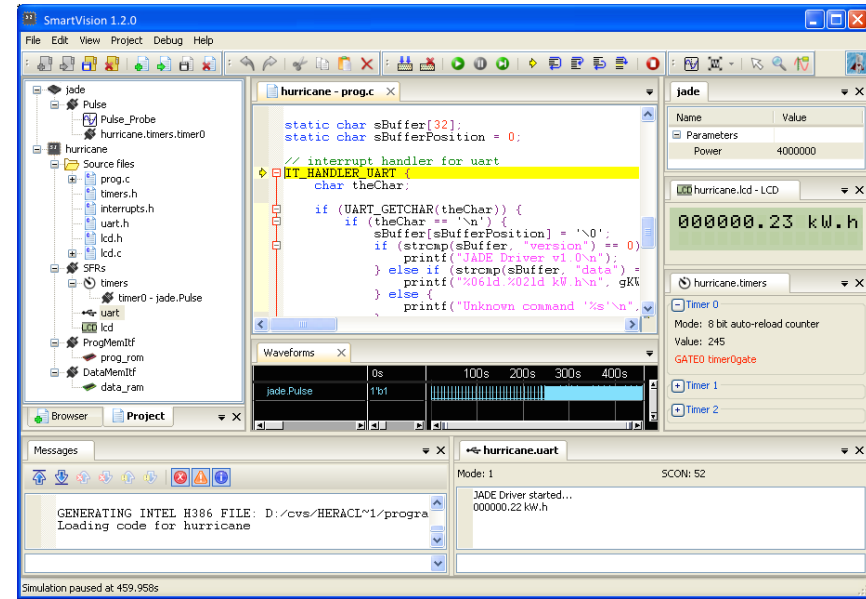


1. 物联网带给微处理器核的挑战
2. 新兴一代i351架构
 1. 拥有32位RISC性能的升级版80x51
 2. 低功耗优化
 3. 子系统面积优化
3. 创新软件开发工具
4. 日渐增长的安全对策需求
5. 80x51微控制器的光明未来

微控制器处理内核必须有一系列的方案配合：

- 一组外设
- 在线仿真器(In-Circuit Emulator)来达到片上调试
- USB调试适配器
- 集成开发环境(IDE) 包括
 - ➔ 指令集仿真器(ISS)
 - ➔ 编译工具链
 - ➔ 调试工具(Debugger)

- 创新的IDE SmartVision 应对程序开发及调试
 - 软件仿真, 硬件仿真及片上仿真
- 物联网设备包含新的外设
 - 可在SmartVision中给其外设建模
 - 可用已有的Verilog RTL 模型
 - 可开发 C 模型
 - 整体外设建模下可找出原本只能在测试板(prototype)上找到的错误
- 检查应用程序与SoC集成商定义的红控制网络的配合
 - 例如程序中需要使用某个外设, 但它被控制成关闭状态下 → 错误检测警告
 - 某个外设一直处于激活状态, 虽然控制网络失可以将其关闭的 → 功耗优化的信息提示
- 可验证并显示I/O指令对电源调节器及其它现代外设网络的影响



- 拥有最新技术的尖端编译器及编译工具链 (汇编器assembler, linker)
- 基于底层虚拟机LLVM的编译器
 - ➔ 开放源代码
 - ➔ 一组模块化和可重复使用的编译器及工具链技术
- 关键优势
 - ➔ 可编译C/C++ (和 Java) 语言
 - ➔ 可全面进入程序符号(program symbols)下进行协调
 - ➔ 可提出源代码上下文感知的修正建议
 - ➔ 可用C/C++源代码静态分析去自动找出应用程序中的错误
- 轻松地 从现有8051 或 80251 C 代码迁移



1. 物联网带给微处理器核的挑战
2. 新兴一代i351架构
 1. 拥有32位RISC性能的升级版80x51
 2. 低功耗优化
 3. 子系统面积优化
3. 创新软件开发工具
4. 日渐增长的安全对策需求
5. 80x51微控制器的光明未来

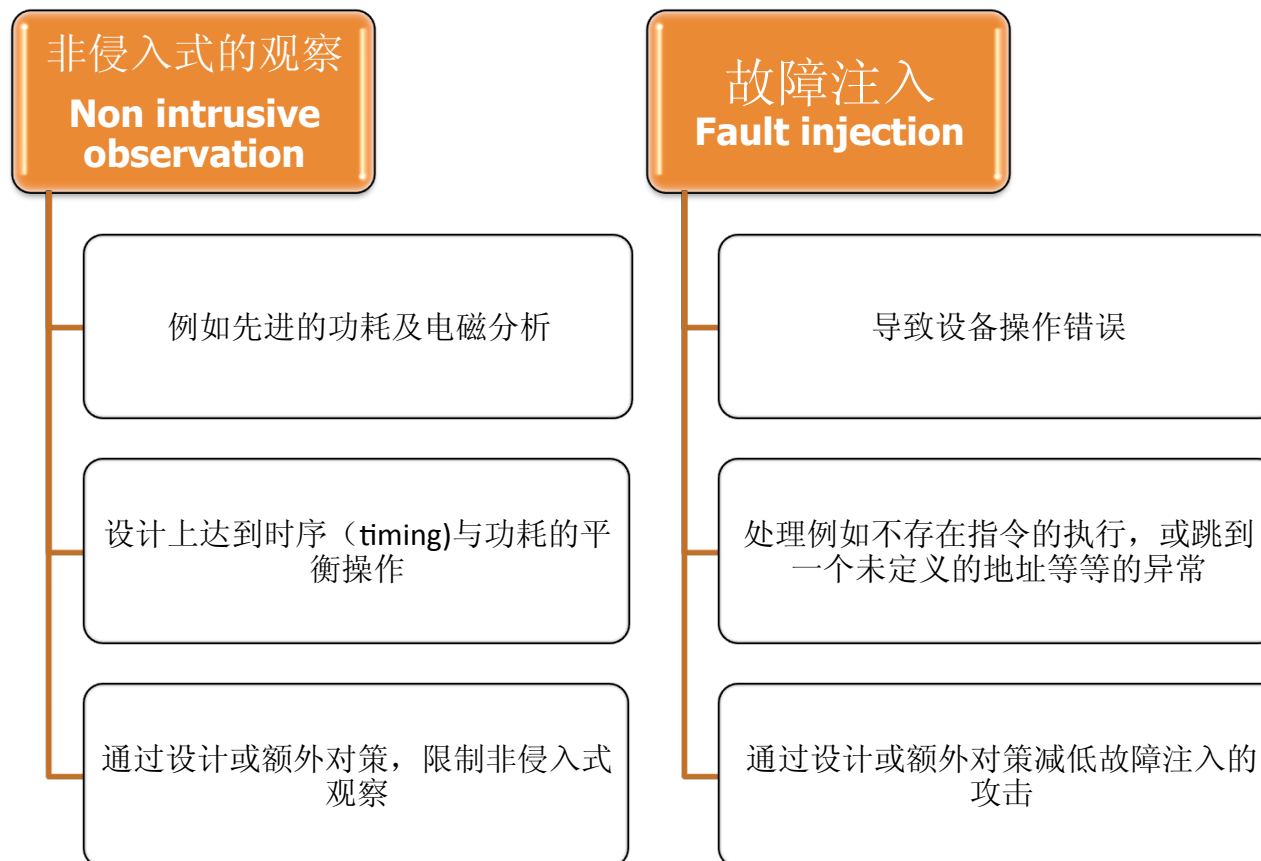
- 物联网设备收集大量数据
 - ➔ 高侵权风险导致高安全对策需求
- 很多丑闻
 - ➔ 很多明星私人照片被泄漏全球
 - ➔ 私人与公共视频监控映像泄漏
 - ➔ ...



- 安全对策的目的
 - ➔ 控制应用程序存取关键数据
 - ❖ 也就是确保资产保密性
 - ➔ 防御数据被修改
 - ❖ 也就是确保数据完整性



- 需要安全对策来抵抗不同类型的攻击



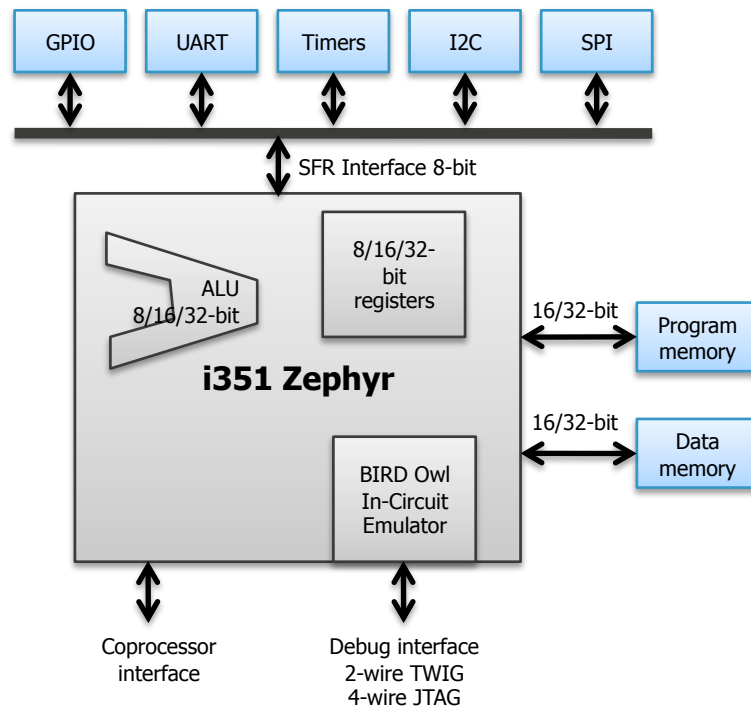
- 两种级别特权来管理存储器系统的存取权限（固件及嵌入软件可升级的系统）
 - ➔ **特权模式**:如操作系统关键程序(OS routines), 加密 (cryptography), 自举法 (bootstrapping) 等 ...
 - ➔ **非特权模式**:而其它任务则在存储器受限下运行
- 创新的安全对策方案:一系列的防御对策
 - ➔ **冗余对策的机密组**
 - ❖ 每个客户的安全专家都可以依照其安全需求来要求定制配置
 - ➔ 此防御对策才能避免多米诺效应
 - ❖ 每个子系统都有其特定的安全对策配置, 因此某个特定的子系统被攻破也不会导致多米诺效应
- Zephyr 加上 “Armory” 武装版本能够**对抗非倾入式观察及故障注入攻击**



1. 物联网带给微处理器核的挑战
2. 新兴一代i351架构
 1. 拥有32位RISC性能的升级版80x51
 2. 低功耗优化
 3. 子系统面积优化
3. 创新软件开发工具
4. 日渐增长的安全对策需求
5. **80x51**微控制器的光明未来

微处理器内核介绍

Flip80351 Zephyr和风



- 架构
 - ➔ 32位 (传承8051)
- 处理能力
 - ➔ 1 DMIPS/MHz
 - ➔ 2 Coremark/MHz
- 子系统优化
 - ➔ 比典型的8051面积小至 54%
 - ➔ 比典型的8051功耗小至26倍
- 软件开发工具
 - ➔ SmartVision IDE
 - ➔ BIRD 在线仿真
- 安全对策系统
 - ➔ Armory, 安全对策武装库

传承80x51 的演变

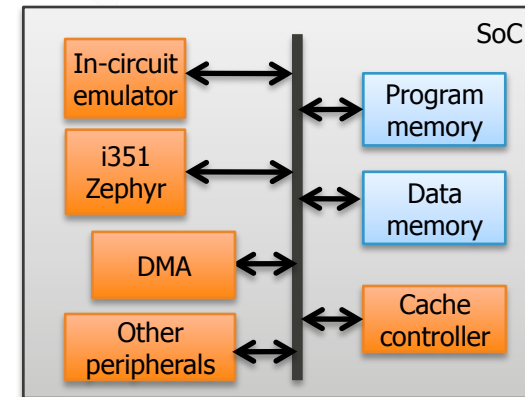
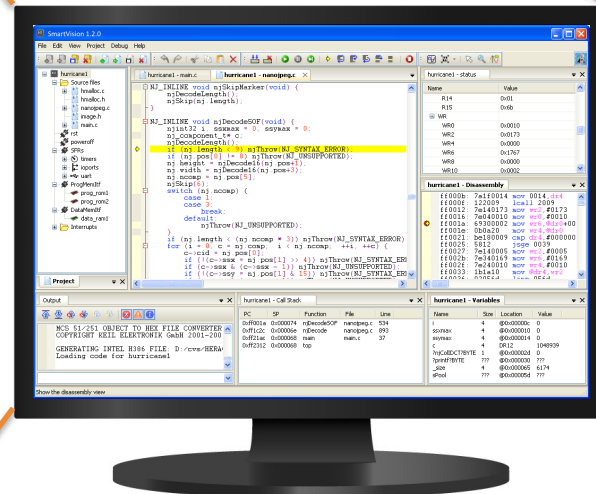
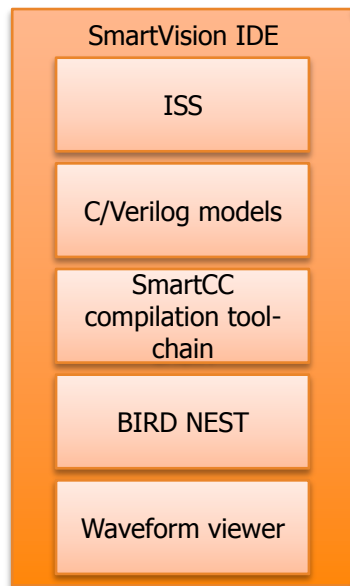
低系统延迟

公众安全

功耗优化

互联互通的边缘设备

高处理性能



Dolphin Integration microcontroller offering

先进 IDE

及时保留

小系统面积

个人隐私

客制安全对策实施

嵌入式 RTOS