

# 嵌入式系统的低功耗软件设计方法

郭 兵

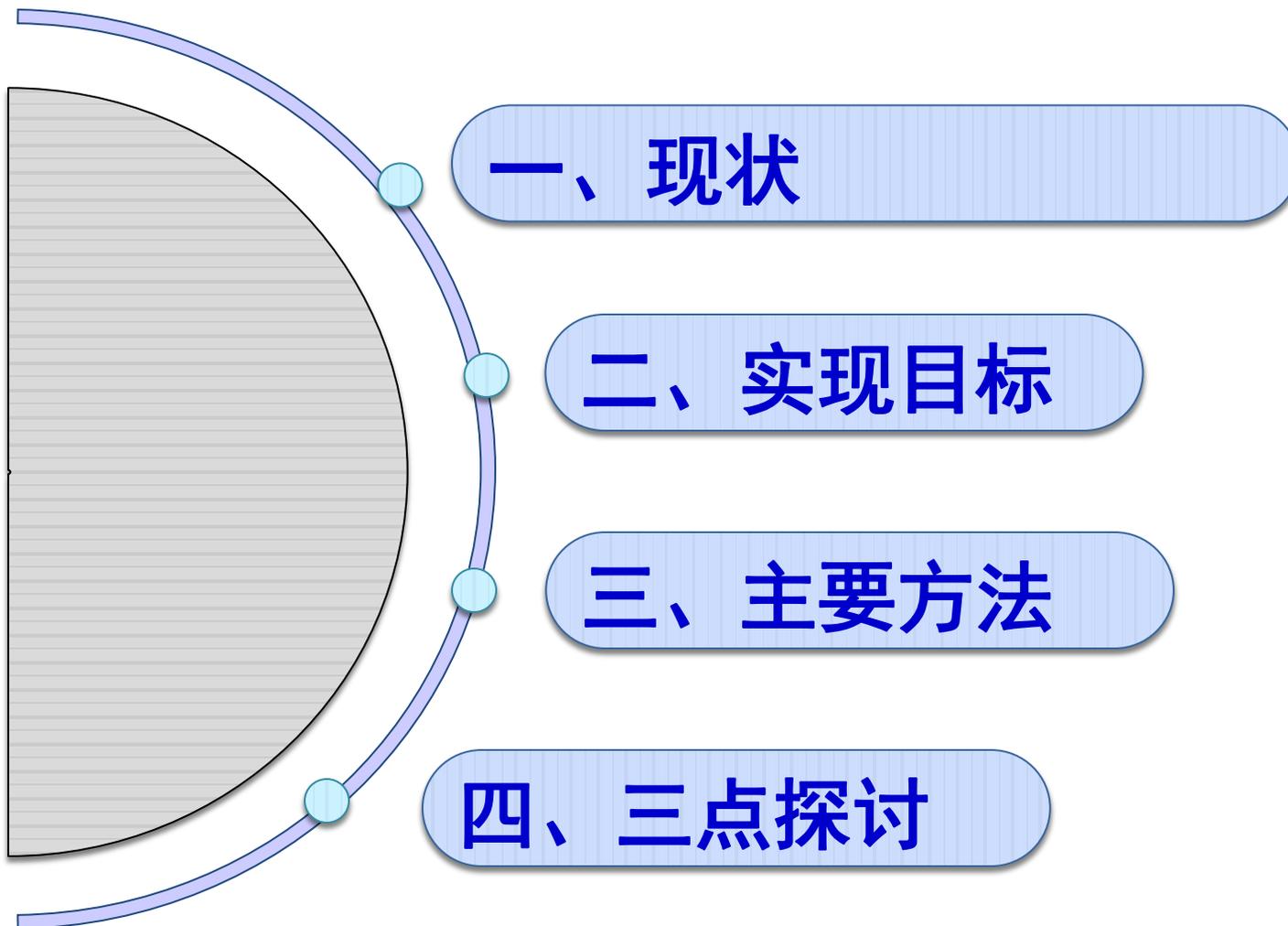
四川大学

2016年7月15日



# 提纲

---



# 一、现状 — 绿色计算要求



## 全球大背景

- 全球气候变暖等环保问题和中国的节能减排政策。
- 计算机领域的节能减排问题：12%，第五位。
- 智能手机、平板电脑等电池驱动的智能移动互联网嵌入式系统的普及，能耗问题更为迫切。



目前智能嵌入式系统中，主要考虑三项重要的设计目标：

■ 多样化功能与服务：移动电话还是移动PC？

■ 高性能：CPU、存储器、GPU、I/O接口等。

■ 便携性：电池的尺寸和容量相对有限，影响系统的续航能力、散热与服务的使用性。

# 应用前景

---

- 根据市场研究机构Gartner统计，2015年全球智能手机销量为14亿部，年增长率为14%；2015年中国智能手机销量超过4.3亿部，年增长率为2.5%。
- 据IDC预计，2015年全球平板电脑销量达到5亿台，超过PC机。
- 传统PC机市场将逐步萎缩，移动互联时代来临。相对于处理器技术的快速发展，电池技术发展严重滞后，而能耗问题始终是困扰智能手机和平板电脑应用的一个重要问题，如何合理延长现有电池供电时间已成为业内关注的热点。

---

主要技术手段  
有哪些？



# 绿色计算（Green Computing）

绿色计算之“绿”，既有自然、生态含义，更有社会、人文之意，我们可以从三个不同的视角描述绿色计算的含义：

1) 工业界的视角。业内不同厂商对绿色计算的概念都有不同的理解，也采取着不同方式开发自己的产品，但迄今业界有一个比较认可的说法。所谓绿色计算，考虑设备全生命周期内，即在设计、制造、购买和使用IT产品的时候，除了追求高性能或高性价比之外，也要考虑电力消耗、空间占用、热耗散、回收利用等因素，达到节能、环保和节约的要求。



---

2) 理论的视角。与绿色化学与技术、可持续发展的理论和方法进行比较研究，我们可以采取一种更加统一和抽象的方式定义绿色计算，即：绿色计算是一种以环境为中心的计算模式，与其它计算模式相融合（包括高性能计算、可信计算、实时计算、分布式计算、C/S计算、嵌入式计算、普适计算和云计算等），从源头上、从根本上改进/消除计算机系统的环境不友好方面，追求计算机系统与人、社会、资源、环境和经济发展的和谐关系，避免计算机系统的“过度计算”现象，即过分追求系统的性能，实现计算机系统的“绿色”特性（即环境友好性）。

3) 交叉学科的视角。绿色计算是一种哲学，也是一种处理问题的理念和方法，各种因素相互联系、相互作用、相互影响。它是计算机科学、环境科学、经济管理科学、材料科学、医学等的交叉学科，作为一种重要的指导原则，指导企业设计出有利于经济发展、生态保护和社会增益的计算机系统，也可以称为可持续计算 (Sustainable Computing)、环境相关的计算(Environment-aware Computing) 或环境友好的计算 (Environment-friendly Computing)。

# 开源措施

---



- 提高电池等储能设施的能量密度，锂电池 → 燃料电池/超级电容/石墨烯电池/纳米材料电池等。
- 增加无线供电/太阳能薄膜/机械振动等新的供电方式。

# 节流措施



■ 在电池容量限制的条件下，一方面提高系统运行时的能量计算效率问题（即能效比，表示单位功耗产生的计算能力，单位为FLOP/Watt或MIPS/Watt）一直未得到充分重视，而存在片面追求计算能力、存储能力和通信能力的倾向。

■ 另一方面，加强系统空闲时间的电源管理，减少能量的无谓损耗。

在“开源”有限的情况下，“节流”成为延长系统续航能力和提高可用性的一个重要途径。

# 低能耗软件设计需求



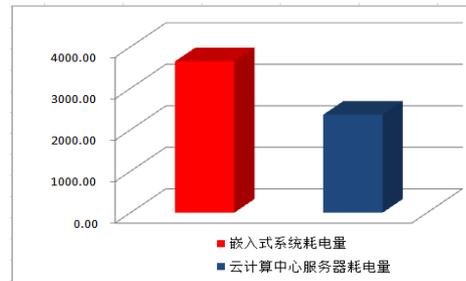
后PC时代



电力网络



环境影响



耗电比较

“无处不在的嵌入”，年销量达200亿台，关键的全球计算基础设施。

同电力网络紧密相连，运行导致大量的CO<sub>2</sub>排放，产生严重的环境影响。

2015耗电4650亿千瓦时，是服务器耗电量的2.1倍。

## 新的计算需求

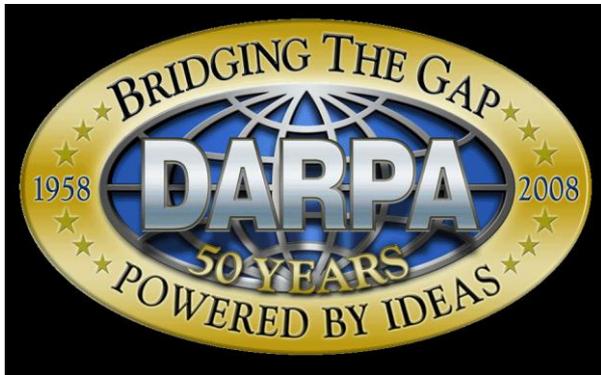
软件是硬件的消费者与管理者，“间接”导致系统能耗的产生。

低能耗软件设计方法

“后摩尔时代”，软件的低能耗设计成为控制系统能耗的重要手段。

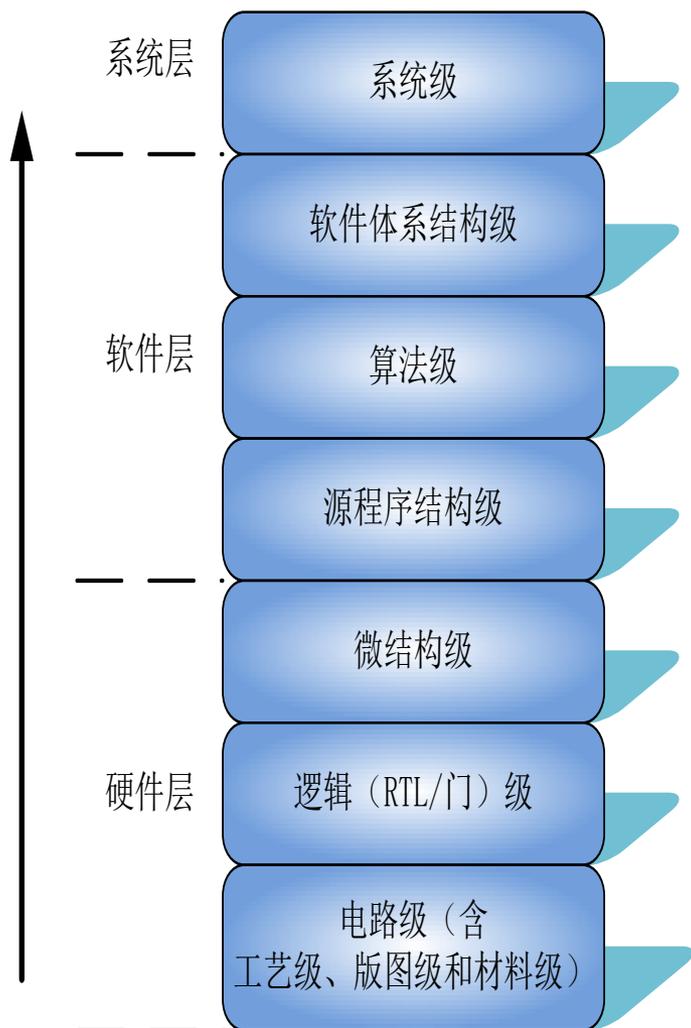
2008年和2010年：Gartner公司都将绿色计算列为十大IT关键技术之首！

# PAC/C计划



1999年，DARPA（美国国防部高级研究计划署）启动PAC/C（Power-Aware Computing and Communications感功计算与通信）计划，研究内容非常广泛，涉及到计算机系统的所有方面，包括芯片、体系结构、编译器、操作系统、网络通信、实时系统、分布式系统到应用工程等多个技术层次，开发各种相关策略、算法、协议、工具和设备，为系统提供一个完整的功耗解决方案。

# 低能耗建模与优化的主要层次



**能耗**是嵌入式系统的一个全局属性，涉及硬件、软件等多个层面的设计时方法。

■ **硬件层**：电路级解决负载电容、供电电压和时钟频率等问题；逻辑级解决电路结构和逻辑设计风格等问题；微结构级解决指令级并行、部件分配与调度和智能功率控制等问题。

■ **软件层**：源程序结构级考虑源程序的语法逻辑结构对软件功耗的影响；算法级考虑算法处理的流程和步骤对软件功耗的影响；软件体系结构级考虑软件体系结构的选择和变换对软件功耗的影响

■ **系统层**：系统级侧重于软/硬件协同、交互、控制和管理等措施，如软/硬件划分与交互等。

# 硬件层面的能耗优化

## 电路级优化

目前 CMOS管的功耗一般可以表示为：

$$P = P_d + P_{sc} + P_I$$

- $P_d$ ：动态功耗，表示当电路中的节点电压发生0→1 或者 1→0 跳变时，节点电容进行充放电时所消耗的能量。
- $P_{sc}$ ：瞬态功耗，是节点跳变时，驱动管与负载管瞬间同时导通而产生的功耗。
- $P_I$ ：静态功耗，漏电流造成的功耗。

90nm以上工艺时，动态功耗是最主要的功耗，可表示为：

$$P_d = CNV^2f$$

其中：C表示电路负载电容，N表示电路每秒的信号反转次数，V表示工作电压，f表示工作频率。

90nm以下工艺时，静态功耗是最主要的功耗，采用DPM是主要方法。

## RTL/门级优化

- **RTL**: 主要考虑资源共享、门控时钟、时序优化、状态分配以及预计算等设计结构和设计风格。
- **门级**: 公因子提取、工艺映射、门尺寸优化等。

## 微结构级优化: 硬件系统或CPU体系结构

- **Gated Clock**。
- **同步电路/异步电路**。
- **流水线/多核**。
- **多电压分配**

## 其它硬件级别优化

- **版图级**: 布局/布线。
- **工艺级**: TTL/NMOS/PMOS/CMOS, 180nm → 30nm。
- **材料级**: 决定晶体管基本的电气指标, 如电阻/电容/电感。

# 软件层面的能耗优化

---

## 源程序结构级优化

- 表达式变化、指令变换、指令重排。
- 优化数据表示。
- 循环结构优化。
- 程序结构重排。

## 算法级优化

- 消除重复计算，减少操作步数。
- 改善数据结构，减少操作数。
- 压缩数据存储空间。
- 算法选择。

## 软件体系结构级优化

- 体系结构风格的选择。
- 体系结构风格的变换，包括Components和Connectors的拆分、合并、简化、替换等操作。

## 其它级别优化

- 系统体系结构级：软/硬件协同设计，DPM，DVS、存储器分离等。
- 编译器级：编译器许多与机器无关的前端优化工作和源程序结构级优化是等价的，编译器只是包含了这些优化方法的工具实现，同时，编译器的优化能力也存在一定的局限性，因此，可将此部分优化工作归入源程序结构级优化中。

# 不同级别优化间的关系

■功耗优化技术的不同层次所考虑的问题和方法是不一样的，如在电路级，着眼于供电电压、时钟频率和负载电容等因素；在微结构级，主要是对CPU的流水线和并行结构进行改进，降低能耗。

■功耗优化技术的不同层次并不是完全独立的，为达到同一个优化目标往往可在多个层次上进行设计，如为了降低电路的有效翻转率（或切换活动因子），可采用在系统级改变系统的软/硬件划分、逻辑级改变具体门电路的逻辑安排等措施，一般高层服务需要下层服务的支持，高层次的措施通过低层次的措施发挥作用。

■通常，不同层次的设计能够降低的能耗比例也不一样，技术层次越高，能耗降低的效果越明显，如通过在系统级软/硬件划分可能降低电路30%的翻转率，而通过在逻辑级重新安排逻辑结构却只能降低5%的翻转率。

■不同层次的措施，开发工作量也会有所不同，如软/硬件划分时，需要进行大量的分析和计算后才能确定有效的体系结构，而硬件逻辑的重新安排大部分工作可通过硬件设计综合软件实现。因此，在低功耗设计时，可根据系统的具体要求选择合适的能耗设计层次与方法。



# 软件能耗



软件能耗是指在软件运行期间，驱动相关硬件活动产生的系统能耗总和，系统休眠或空闲时（无软件运行时）硬件产生的一些系统能耗或一些无关硬件产生的能耗一般不归为软件能耗。

主要特点：

■软件是硬件的消费者与管理者，软件本身不会产生能耗，能耗只是软件执行时的“副产品”，软件的指令执行和数据存取等操作驱动了底层硬件的电路活动，“间接”导致了系统能耗的产生，软件是能量的主动消费者，是产生系统能耗的“主动”因素和“活跃”因素，决定了系统能耗的实际值。

■明确定义和划分软件能耗，更便于直接观察在系统运行状态下软件实现方式（包括软件体系结构、数据结构、算法和语法结构）对系统能耗的影响程度，反映了软件的实现方式对软件运行效率的影响。

# 低能耗软件设计的挑战

## 低能耗软件设计的难点

1

多性能约束条件下低能耗软件设计

2

软件的应用与环境自适应性

## 低能耗软件设计的分类

1

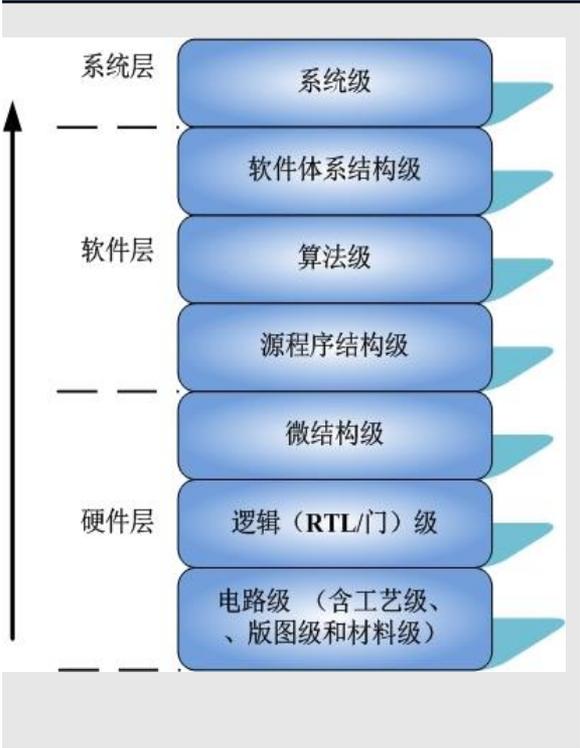
低能耗 by 软件：通过软件方式降低能耗，即动态设计方法。

2

低能耗 of 软件：软件本身的低能耗，即静态设计方法。

改变传统软件设计方法，从追求软件开发效率到软件开发效率和执行效率并重！

# 研究现状

嵌入式系统的低能耗	低能耗软件的静态设计	低能耗软件的动态设计
 <p>系统层</p> <p>系统级</p> <p>软件体系结构级</p> <p>软件层</p> <p>算法级</p> <p>源程序结构级</p> <p>微结构级</p> <p>硬件层</p> <p>逻辑 (RTL/门) 级</p> <p>电路级 (含工艺级、版图级和材料级)</p>	<p>1 源程序结构级优化：表达式变化、优化数据表示、程序结构重排；指令变换、指令重排、循环结构优化、存储器和Cache分配等。</p> <p>2 算法级优化：消除重复计算、改善数据结构、压缩数据存储空间、算法选择等。</p> <p>3 软件体系结构级优化：体系结构的选择、变换和简化等。</p>	<p>1 DPM技术：为CPU和I/O设备提供电源管理的功能及接口，选择运行、待机、休眠等多种工作模式。</p> <p>2 DVFS技术：根据负载情况，动态改变和调节CPU的运行频率与工作电压。</p> <p>3 RTOS功耗相关任务调度模型及算法：通过任务能耗特性的研究，改善任务的资源分配与调度。</p>

## 二、实现目标

---

### 三大难题

1 嵌入式系统的多样化  
功能与低能耗的矛盾

2 嵌入式系统的高性能  
与低能耗的矛盾

3 嵌入式系统的可用性  
与低能耗的矛盾

迫切需要以嵌入式系统的低能耗为目标，从静态设计和动态设计两方面，勾勒出低能耗软件的设计理论和方法体系！

# 总体方案

以解决低能耗软件设计所面临的三个难题为核心，建立低能耗软件的设计理论和方法体系，实现嵌入式系统能量资源的最优应用适应性分配与使用。

重要应用上有贡献

面向智能终端和路由器的实例应用与验证工作

关键技术上有创新

能量优先的任务运行模式

系统资源的自适应软件节能配置策略

高效可定制化的软件体系结构

能量感知的编程模式

低能耗软件的演化与控制

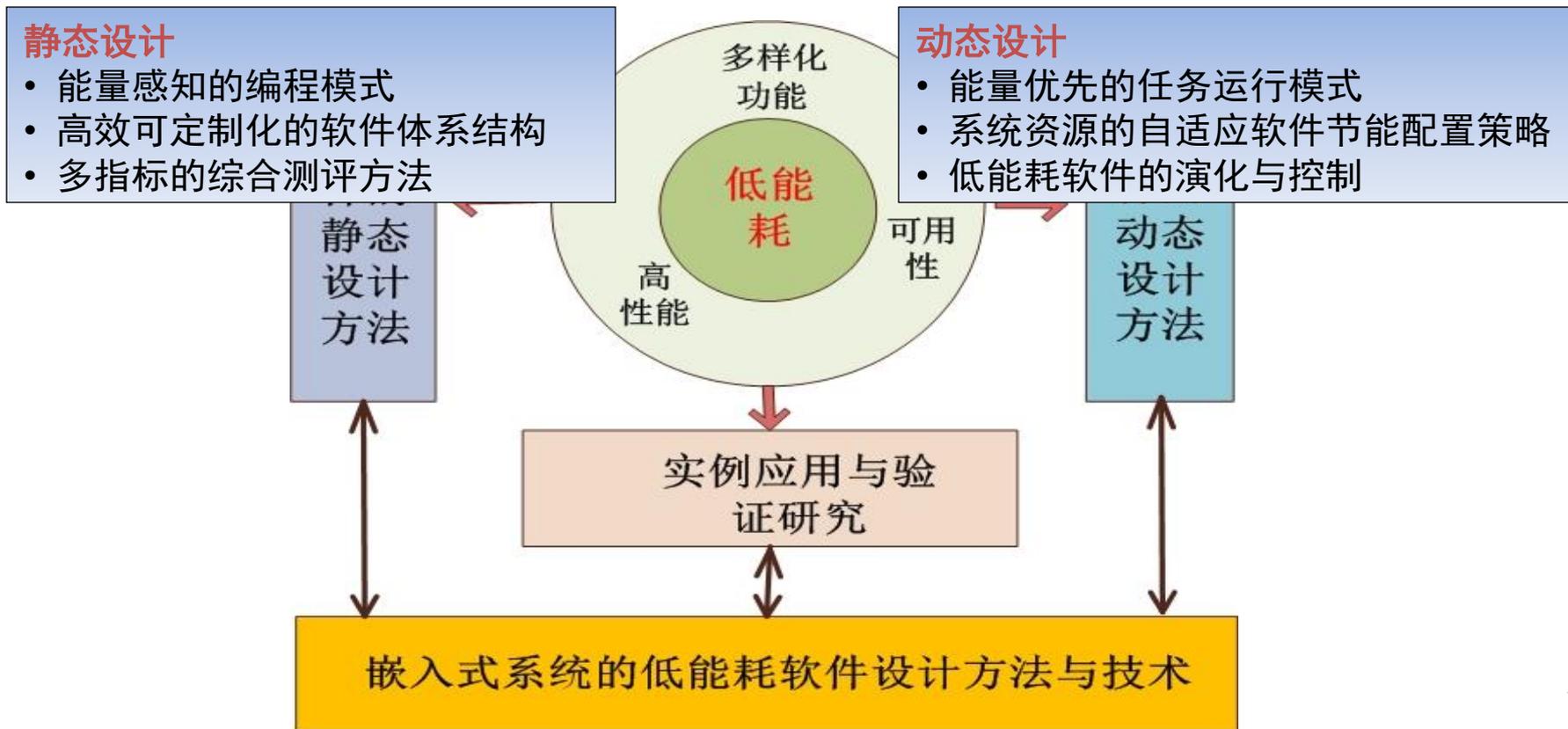
多指标的综合测评方法

基础理论上有所突破

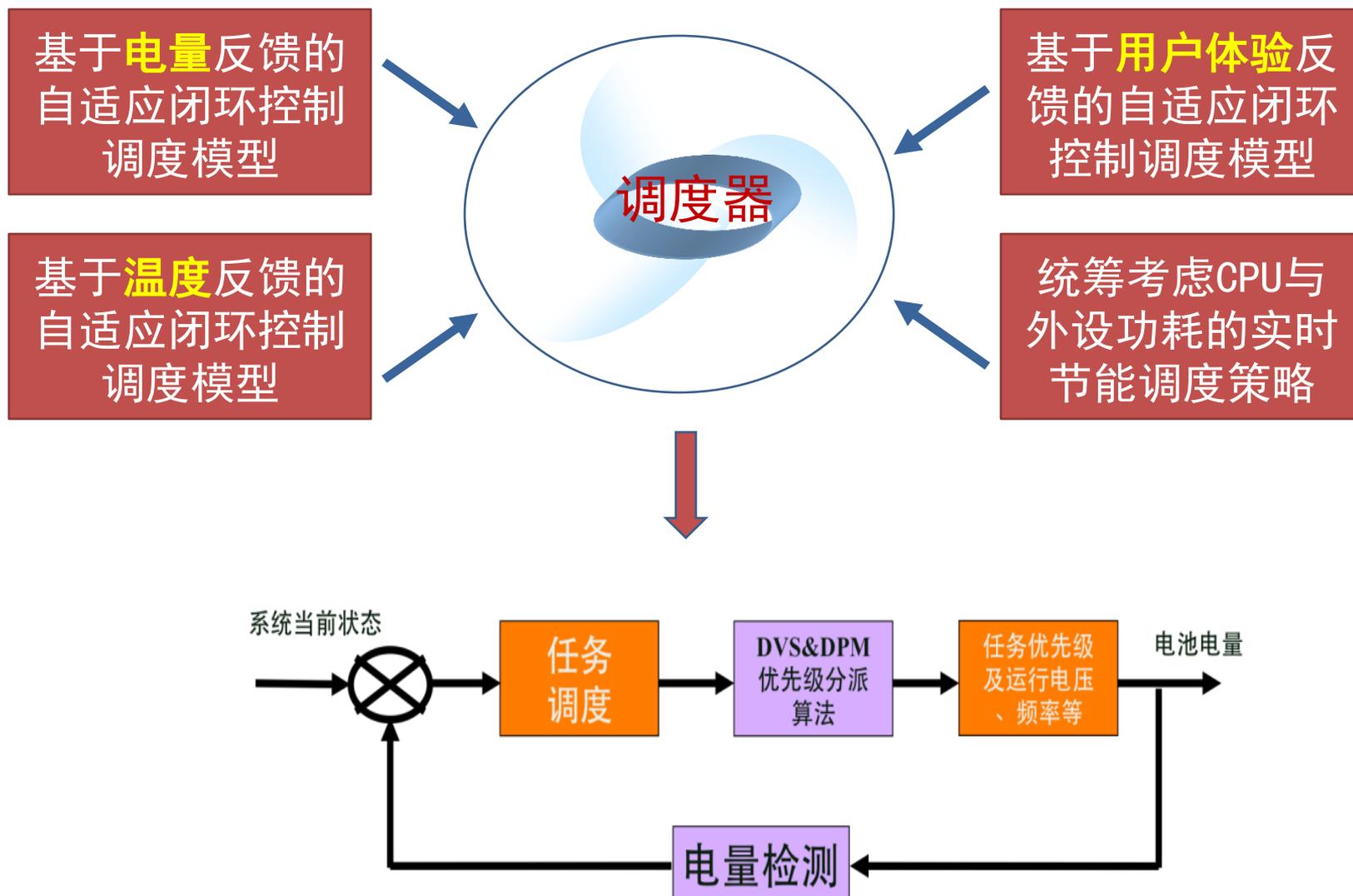
低能耗软件的设计理论和方法体系

# 三、主要方法

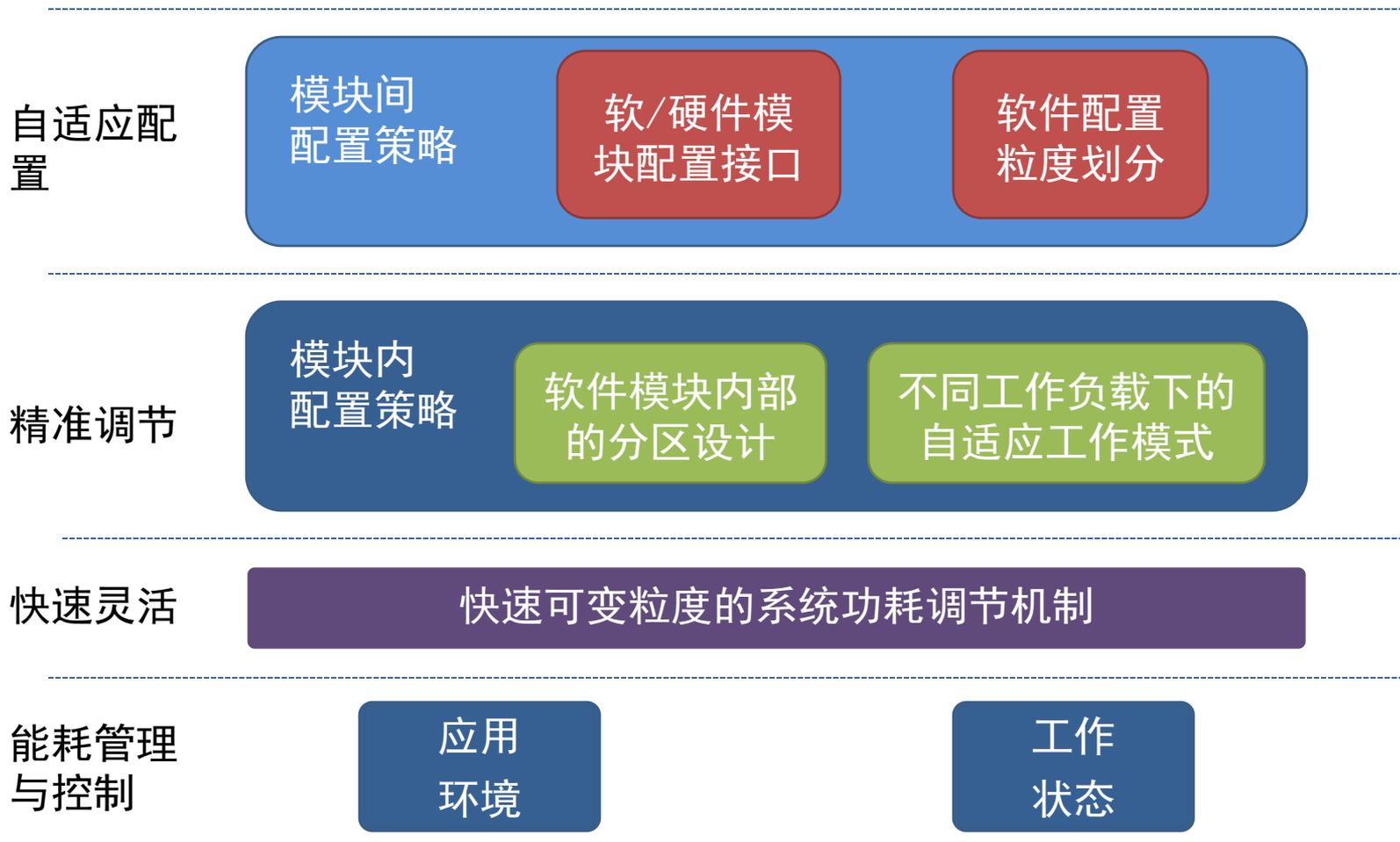
嵌入式系统的低功耗软件设计方法与技术是在嵌入式系统环境下，研究在多性能约束条件下（包括高性能、可用性等多维度）嵌入式软件（RTOS、中间件和应用软件）能耗的管理、控制、优化与评价等理论和技术问题，是一个多约束的组合优化问题。



# 方法一：能量优先的任务运行模式



# 方法二：系统资源的自适应软件节能配置策略



# 方法三：高效可定制化的软件体系结构

## 软件体系结构的扁平化风格

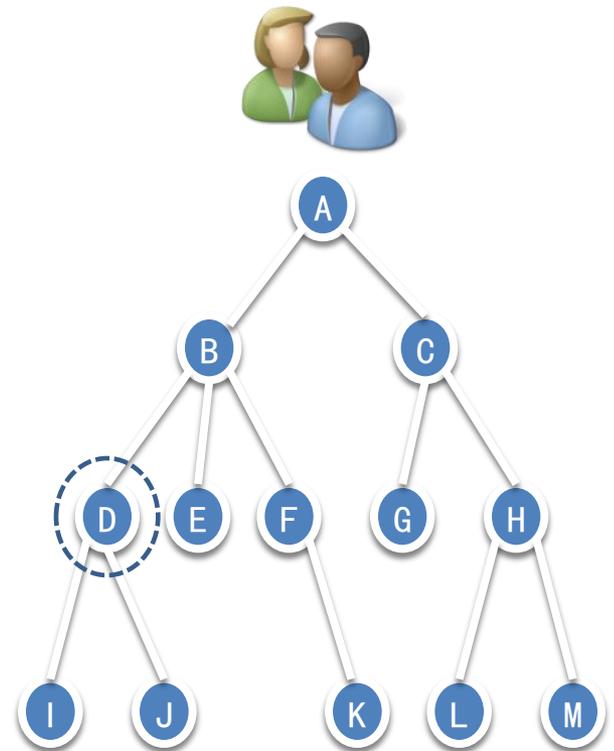
- 层次式系统和并发进程，减少执行层次，简化互联结构
- 提高软件运行效率，避免层次间能耗叠加现象

## 软件体系结构风格的能耗评价

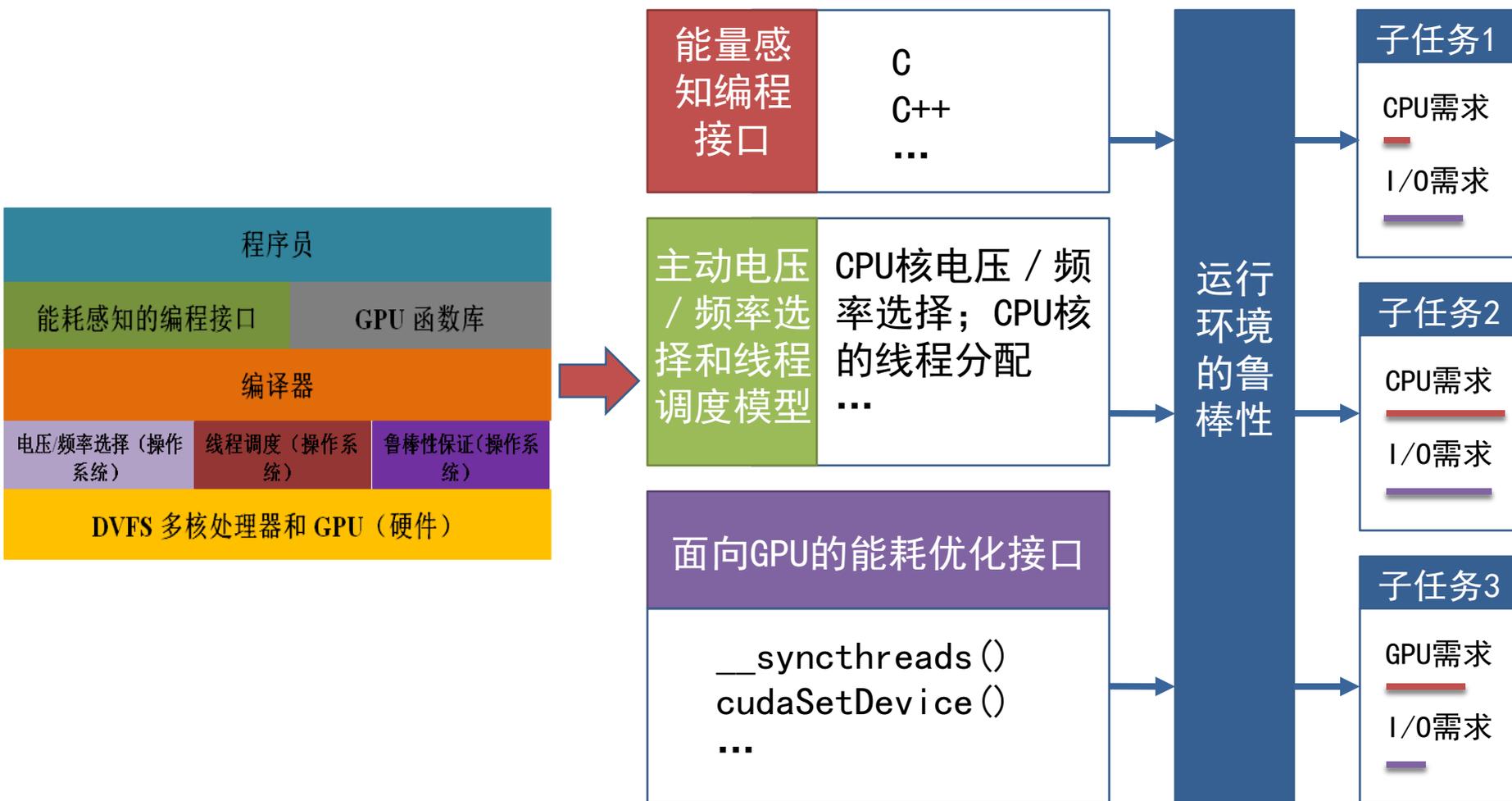
- 定性分析和定量分析
- ATAM和调度理论的混合方法，进行能耗关注点的针对性改进

## 软件构件的可定制化机制

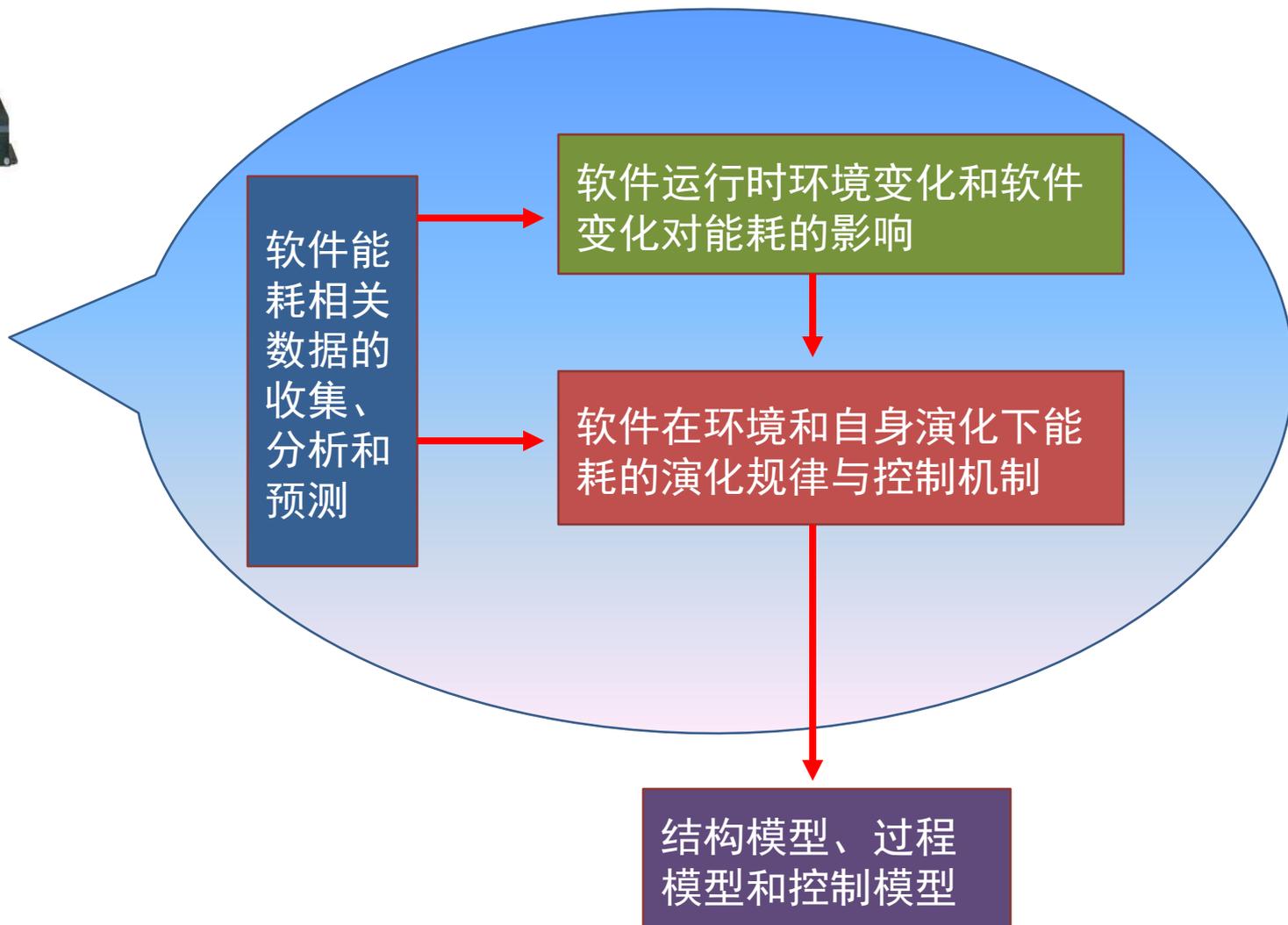
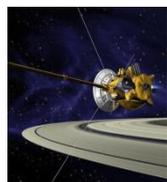
- 编译时构件的裁剪机制和运行时构件的加载/卸载机制
- 构件间拆分、合并、简化、替换等操作功能和接口方法



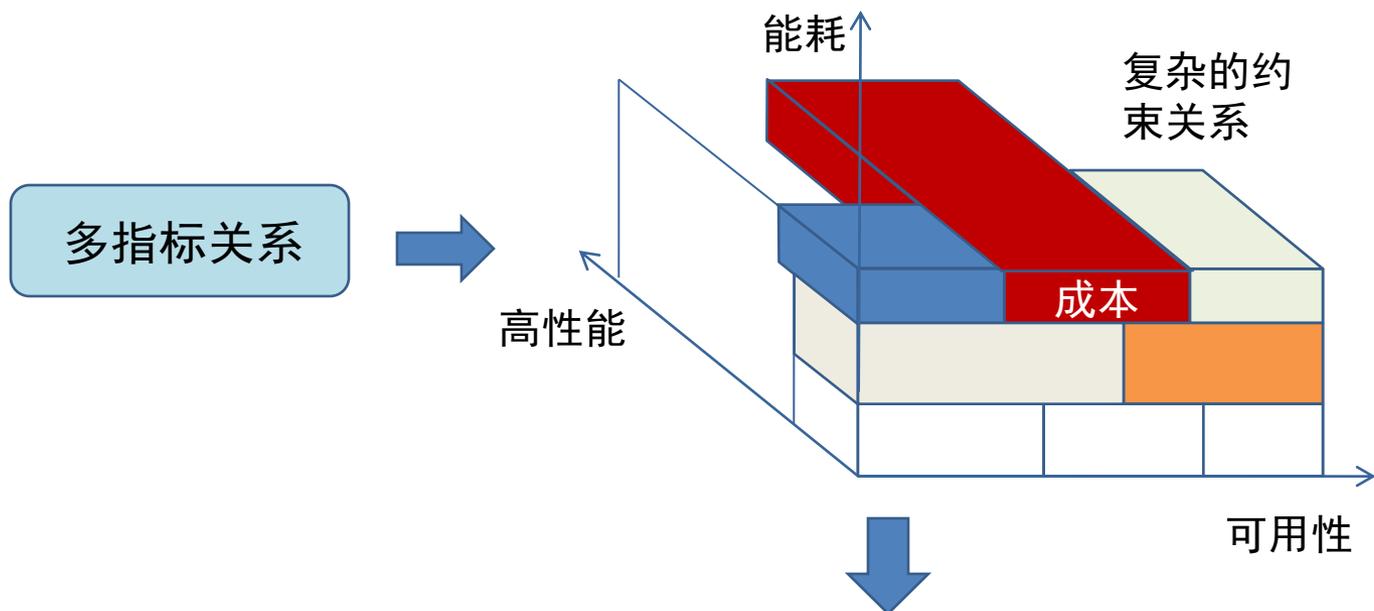
# 方法四：能量感知的编程模式



# 方法五：低能耗软件的演化与控制



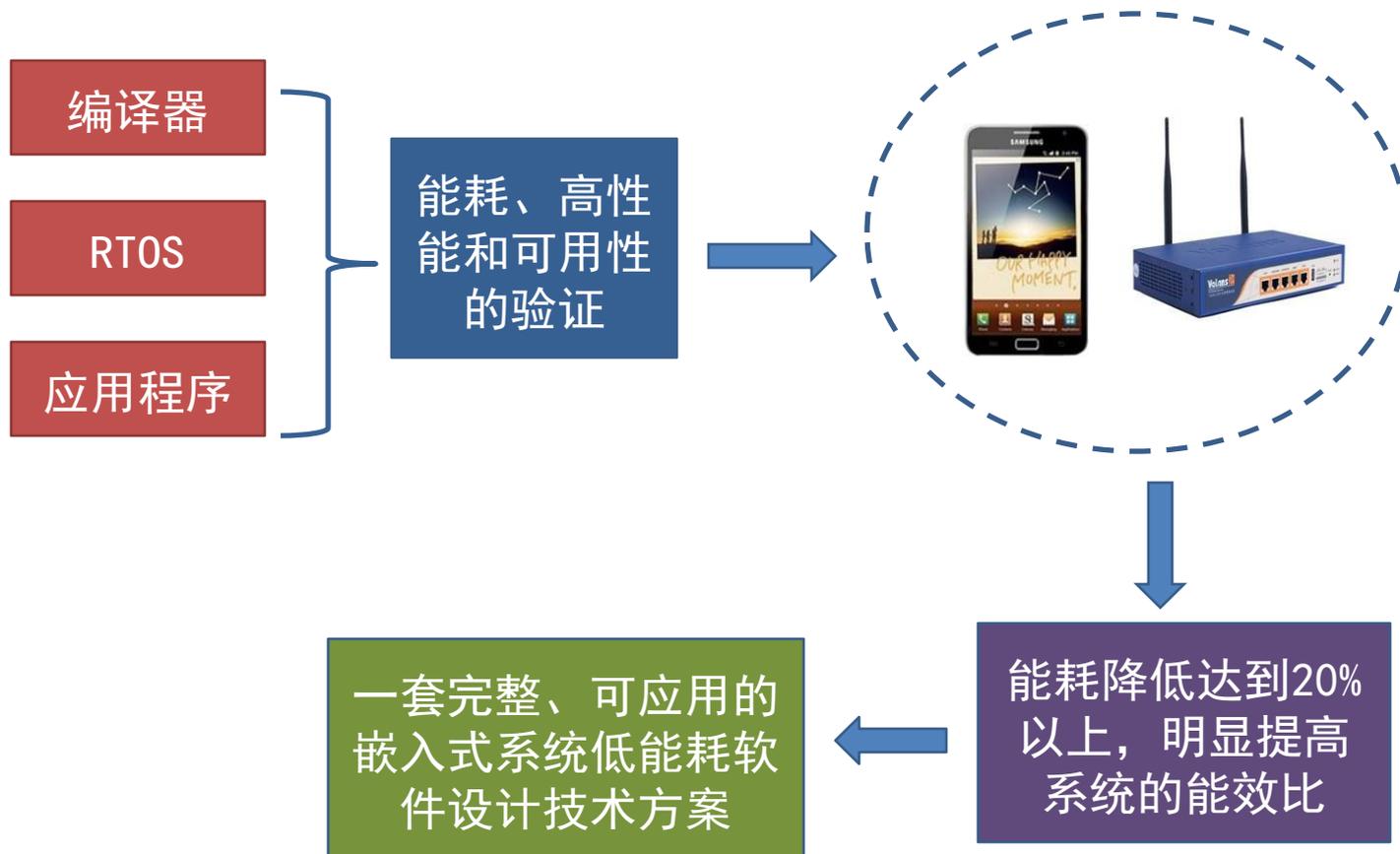
# 方法六：多指标的综合测评方法



低能耗软件多指标的综合测评方法

- 基于支持向量机（SVM）的评价模型
- 基于层次分析法的模糊评价模型（AHP）

# 方法七：实例应用与验证研究



# 关键技术



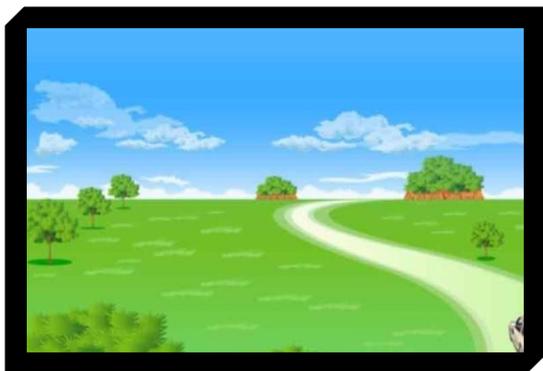
# 四、三点探讨



## 智能移动终端的综合节能技术

智能移动终端的能耗一般由硬件单元功耗、软件任务运行效率及系统资源配置策略等因素决定。主要措施包括：

- 新型高能效的可定制异构多核体系结构
- 能量感知（能量优先）的任务运行模式
- 软/硬件资源的自适应节能配置策略



■软件能耗与可信性的关系：都是软件的重要属性，二者有矛盾。

■绿色计算与绿色制造：软件的绿色设计是计算机技术未来发展的趋势，而嵌入式系统是多数设备的计算和控制核心，智能、柔性、绿色是工业4.0的重要特点，绿色计算是绿色制造的重要组成部分。



谢 谢!