

嵌入式系统开发基础

3.1 基本概念

“嵌入式系统”是嵌入到具体的应用体系当中的专用计算机系统,不同的技术人员从不同的角度观察嵌入式系统,可能会得到不同的定义。IEEE(电气与电子工程师协会)对嵌入式系统的定义为:嵌入式系统是“Devices Used to Control, Monitor or Assist the Operation of Equipment, Machinery or Plants(用于控制、监视或者辅助操作机器和设备的装置)”。国内普遍认为:嵌入式系统是“以应用为中心,以计算机技术为基础,软硬件可裁剪,适应应用系统对功能、可靠性、成本、体积、功耗等严格要求的专用计算机系统”。国内的定义,更加准确地体现了嵌入式系统“嵌入”“专用”以及属于“计算机”范畴的本质特征。

嵌入式系统一般具有如下一些特点。

1. “专用”的计算机系统

与通用计算机不同,嵌入式计算机系统通常都应用于特定的应用场景,具体的应用需求决定对嵌入式处理器以及其他设备的需求。如果应用场景改变,原有的嵌入式系统可能不再适用。

2. 运行环境各异

不同的嵌入式系统有不同的运行环境,其应用范围十分广泛,从飞机、宇宙飞船到电子玩具,从环境恶劣的极地环境到高校实验室,从军用设施到民用电器,都有嵌入式系统的影子。

3. 资源可裁剪

嵌入式系统并不像通用计算机那样拥有完整的系统资源,几乎所有的嵌入式系统都根据具体的应用场景裁剪系统资源,如采集环境温、湿度的环境采集系统可能不需要屏幕,系统内部也无须硬盘来存储数据;小型的电子玩具可能不需要网络通信功能;简单的系统甚至无须操作系统来管理资源。

4. 功耗小、体积小、集成度高,成本低

嵌入式系统“嵌入”到具体的应用体系中,一般的嵌入式系统都具有低功

耗、小体积、高集成度和低成本的特点。比如,很多由 MCU 构建的嵌入式系统无须风扇散热,所有元器件集成在一块较小的电路板上,供电系统采用小型的可充电锂电池。

5. 专用的开发工具和设计方法

不同于通用计算机的开发方法,嵌入式系统的设计开发通过软硬件协同设计、采用交叉编译环境完成开发。此外,调试过程中的断点设置等都需要通过特殊的机制来实现。目前,嵌入式调试已发展出支持系统开发过程的专用工具套件。

6. 知识密集系统

嵌入式系统是技术、资金高度密集的系统,是一个高度分散、不断创新的知识密集体系,是先进计算机技术、半导体工艺、电子技术和通信网络技术相结合的产物。

7. 应用范围广

嵌入式系统已经渗透到人类生活的方方面面,在军事国防、信息家电、消费电子、工业控制、通信网络、智能家居、汽车电子、物联网等领域无处不在。

3.2 嵌入式系统组成

嵌入式系统是专用的计算机系统,其具有一般计算机的共性,也由硬件和软件两部分组成,图 3-1 描述了组成嵌入式系统的各个部分。

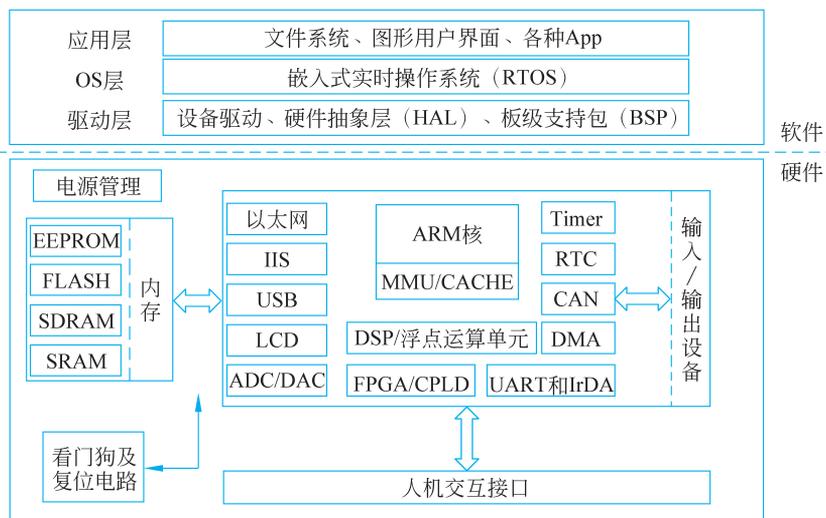


图 3-1 嵌入式系统组成

嵌入式系统硬件是嵌入式系统运行的硬件基础,提供了嵌入式系统运行的物理平台、通信接口和辅助资源。嵌入式操作系统是整个嵌入式系统的核心,控制整个系统的运行,统筹管理系统资源,提供人机交互界面等;嵌入式软件一般分为嵌入式操作系统和嵌入式

应用软件两大部分。嵌入式应用由于与各类应用对象密切相关,而实际应用场景却纷繁复杂,很难用一种架构或模型加以描述。

3.3 嵌入式系统硬件组成

嵌入式系统的硬件组成基本上以处理器为核心,由存储器、I/O 设备、通信模块、电源以及 ADC/DCA、UART、LCD、IIS、CAN 总线等各类辅助电路共同组成。在实际应用中,除了微处理器和内存、电源外,外围资源一般都会根据具体应用场景做相应的裁剪和定制,有利于降低成本。

嵌入式处理器一般可分为微处理器(Microprocessor)、微控制器(Microcontroller Unit)以及嵌入式 DSP(Digital Signal Processor)。嵌入式微处理器就是与通用计算处理器相对应的 CPU,主要应用在一些高性能场景,如消费电子、平板电脑等领域;嵌入式微控制器为开辟单片机市场而开发,芯片上同时封装有 MCU、内存和其他一些功能电路,常见的有 80C51、STM32 等。DSP 芯片是专用于处理数字信号的芯片,当前一些 MPU 和 MCU 中都集成有 DSP 处理单元,大大提高了芯片性能。

嵌入式外围电路是除了处理器之外的其他设备和功能电路,主要有存储器、通信接口、输入/输出设备、设备扩展接口和电源及辅助设备这 5 大类。

存储器: 嵌入式系统的主要功能部件,为存储数据和程序提供保障。常见的存储器有静态易失性存储器(RAM、SRAM),动态存储器(DRAM)、非易失性存储器(ROM、EPROM、EEPROM、FLASH 等)、电子盘等。

通信接口: 嵌入式领域常见的通信接口有 RS-232 串口、USB 接口、IrDA(红外接口)、SPI(串行外围设备接口)、I²C、CAN、蓝牙、以太网、通用输入/输出 GPIO 口等。嵌入式开发中,使用 UART 作为调试信息输入/输出口是常见的做法。

输入/输出设备: 触摸屏、键盘、红外遥控器等。

设备扩展接口: 随着嵌入式系统功能日益复杂,对大容量存储器的扩展需求等变得日益突出,为系统配置相关的扩展接口也是一些嵌入式电子产品在开发过程中常见的做法。

电源及辅助设备: 嵌入式系统一般都配备电池作为供电系统,电池技术的进步和日益小型化使嵌入式电子产品的体积进一步缩小。此外,更低功耗的处理器和更先进的电源管理方案使得设备续航能力也得到改进。

3.4 嵌入式软件开发

嵌入式软件整体上包括驱动层、OS 层以及应用程序,不同岗位的工程师在嵌入式软件开发过程中可能会从事不同类型的软件开发工作。比如,驱动工程师可能致力于对某几类设备驱动程序的开发、OS 设计人员侧重于设计和开发嵌入式操作系统、大多数的嵌入式软件工程师则从事嵌入式应用程序开发工作。

随着半导体工艺的发展和硬件性能的持续提升,基于嵌入式实时操作系统构建应用

程序已经成为嵌入式领域进行应用开发的首选方法。一般而言,嵌入式软件开发无法在嵌入式系统本身有限的资源平台上完成,几乎都需要交叉开发环境,即在通用计算机上搭建开发环境,完成对软件的编辑、编译等工作,然后将系统代码下载到嵌入式平台运行。一般将开发平台称为宿主机,将嵌入式设备称为目标机。目前,主流的微处理器产品都配备功能完备的交叉开发工具,使得嵌入式开发的门槛进一步降低。图 3-2 展示了嵌入式开发的典型环境。

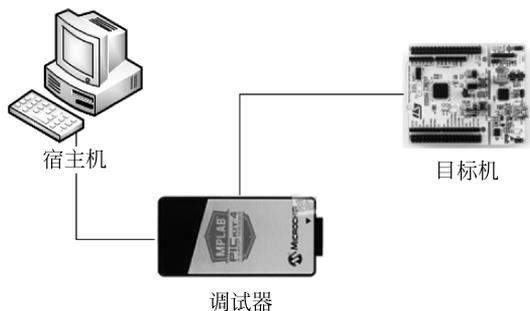


图 3-2 嵌入式开发环境

基于交叉开发环境进行嵌入式应用开发主要包括如下 5 个基本阶段:搭建开发环境、编辑源代码、交叉编译、链接与重定位以及下载与调试,如图 3-3 所示。

1. 搭建开发环境

在进行嵌入式开发之前,必须了解具备交叉编译功能的开发工具套件。目前的各大处理器基本上都提供了一整套的嵌入式开发套件,内置集成了交叉编译环境,使用难度并不大。

按照发布形式,嵌入式开发环境主要有开放和商用 2 种类型。其中,开放的交叉开发环境以 gcc 为典型代表,它支持多种平台的交叉编译,由 GNU 负责维护。商用的交叉开发环境有 μ Vision MDK、Tornado 等。

根据使用方式,交叉开发环境主要分为使用 Makefile 和 IDE 两种。前者要求程序员通过编写 Makefile 来管理开发项目,用户友好性一般;后者提供了对用户友好的操作界面,方便管理和控制项目开发,如 μ Vision MDK。

2. 源代码编辑

一般来说,嵌入式软件中的程序启动代码、硬件初始化代码、操作系统移植代码大都采用汇编语言编写。当然,也有一些平台提供实现硬件初始化功能的 C 语言函数,使得开发难度进一步降低。C/C++ 语言是嵌入式软件开发中使用最普遍的编程语言,当然也有 Python、Java 等语言在一些嵌入式开发领域也逐渐被采用,如 Android App 开发。

3. 编译

编译器主要负责将源代码编译成特定的目标代码,在编译过程中会检查语法错误,最

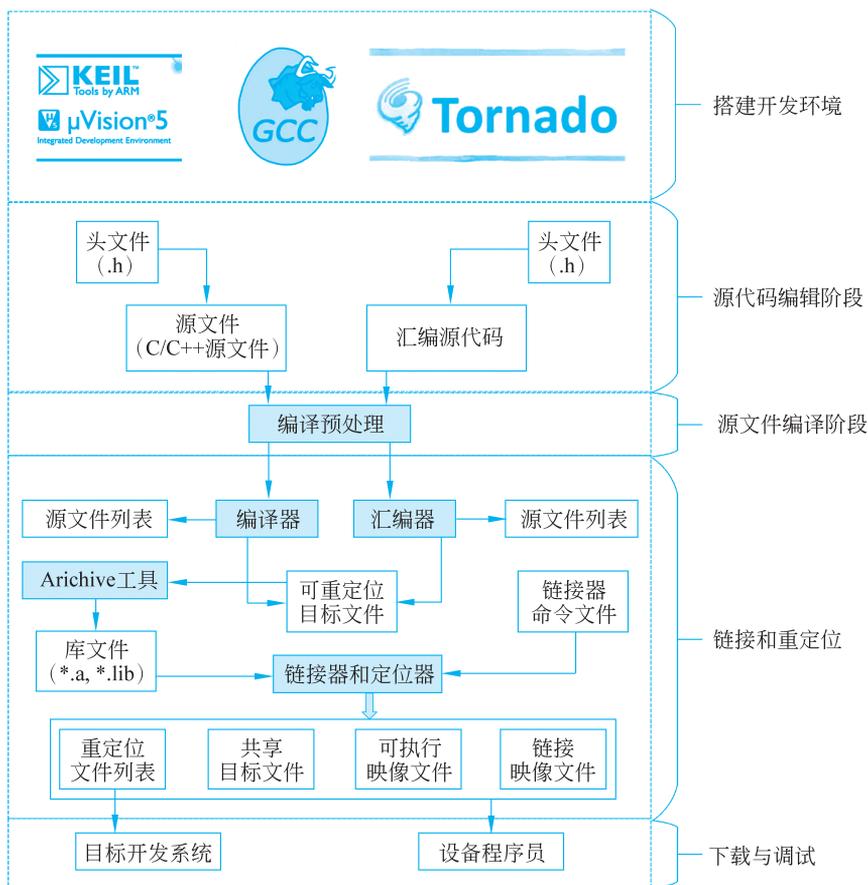


图 3-3 嵌入式软件开发流程

终生成目标代码。目标代码有 2 类：COFF (Common Object File Format) 与 ELF (Extended Linker Format)。在目标文件中规定了信息组织方式,即目标文件格式。目标文件格式的规范使得不同供应商提供的开发工具能够严格遵循,以便实现互操作和信息共享。

4. 链接和重定位

程序要在内存中运行,除了编译之外,还要经过链接的步骤。编译器只能在一个模块内部完成符号名到地址的转换工作,不同模块间的符号解析需要由链接器完成。链接器主要完成符号解析和重定位两项工作。

5. 下载与调试

下载就是将可执行的映像文件烧写到嵌入式目标设备的 ROM 中,嵌入式交叉开发环境基本上都具备 Flash 编程功能。当然,用户也可使用专门的代码烧写工具完成代码的下载。

嵌入式系统调试分为软件调试和硬件调试 2 种。软件调试时通过软件调试器调试嵌入式系统软件,硬件调试通过仿真调试器完成调试过程。

通常调试软件部分的调试器都在宿主机上运行。软件调试工具一般都具有 ISS 功能,即完成代码在无硬件调试环境下的模拟调试。模拟调试与真实的硬件环境有较大的差异,一般用于开发者编程练习或代码的初步调试。

硬件调试可获得比软件调试更多的调试功能和更高的调试性能。硬件调试器的原理是通过仿真硬件的真正执行过程,让开发者在调试过程中可以时刻获得硬件执行情况。硬件调试器主要有在线仿真器(In-Circuit Emulator, ICE)和在线调试器(In-Circuit Debugger, ICD)。ICE 在嵌入式硬件开发中完全仿造调试目标 CPU,成本高昂。ICD 调试是更常见的调试方法,通过连接 ICD 和目标板的调试端口,发送调试命令和接收调试信息,实现必要的调试功能。比如,ARM CPU 通过设置 JTAG 调试口完成对 CPU 的调试,几乎达到了与 ICE 调试相当的效果。

3.5 嵌入式系统开发流程

嵌入式系统是运行于特定目标环境的专用计算机系统,功能专一,在实现预期功能时往往通过软硬件协同设计实现。考虑到实现成本,在器件选型时,各种资源往往只是满足需求即可,预留给用户使用的额外资源并不多。因此,嵌入式系统开发必然有其自身的许多特点。

1. 影响嵌入式系统开发的因素

嵌入式系统是以实际应用为主要考虑对象的专用计算机系统,其特点是软硬件可配置,功能可靠,成本低,体积小,功耗低,实时性强。在嵌入式系统设计和开发过程中一般要考虑如下几个因素。

- (1) 功能可靠性。
- (2) 系统实时性。
- (3) 体积符合规范。
- (4) 接口符合规范。
- (5) 配置精简。
- (6) 功耗管理严格。

2. 软硬件协同设计

嵌入式系统设计是使用一组物理硬件、设计和开发一系列软件来完成所需要功能的过程。整个嵌入式系统是硬件和软件的结合体。由于嵌入式系统是一个专用系统,嵌入式系统在设计过程中,软件设计和硬件设计紧密耦合、相互协调。软硬件协同设计是嵌入式系统设计的基本方法,这种方法的特点是设计时从系统功能考虑,把实现时的软硬件因素统统考虑进去,包括充分利用硬件芯片级的功能,以最大限度地利用有效资源,缩短开发周期,以取得较好的设计效果。嵌入式系统软硬件协同设计流程如图 3-4 所示。

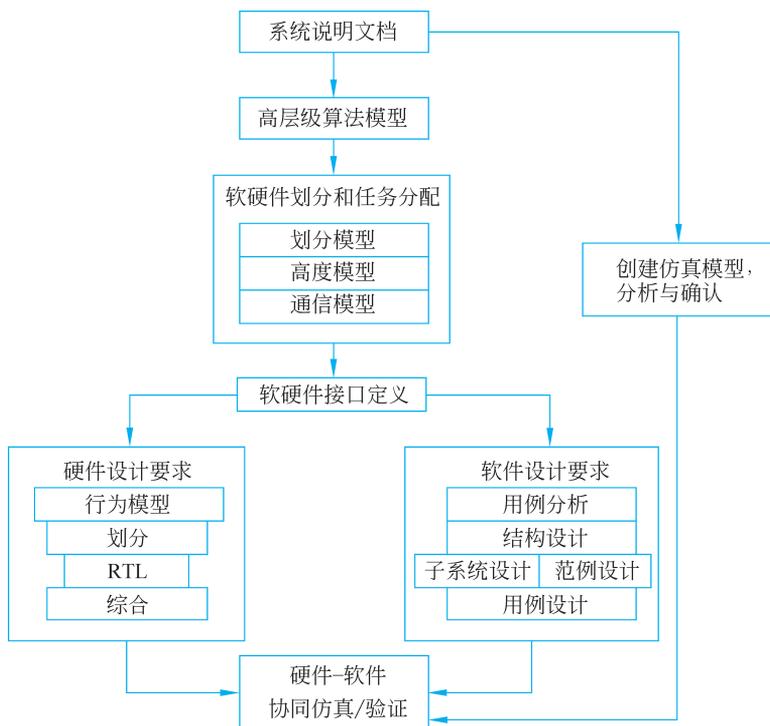


图 3-4 软硬件协同设计流程

软硬件协同设计的流程从确定系统的需求开始,包括系统要求的功能、性能、功耗、成本、可靠性、开发时间等。系统需求由项目开发小组和业务专家共同开发制定,最终形成需求规格说明书。

系统设计首先确定系统所需要的功能。复杂系统设计方法中最常用的是将整个系统划分为较简单的子系统以及这些子系统的模块组合,然后以一种选定的语言对各个对象子系统加以描述,生成设计规格说明文档。

3. 嵌入式系统开发的基本流程

嵌入式系统开发将硬件、软件、人力资源、市场等各种因素结合起来。任何一个嵌入式系统都是软硬件结合体,是软件和硬件协同综合开发的结果,这是嵌入式系统最大的特点。面向具体应用的嵌入式开发决定了嵌入式开发的方法、流程各不相同,但总体上按照如图 3-5 所示的流程进行。

(1) 系统需求分析。系统需求分析旨在了解用户在实际应用中面临什么样的问题,需要什么样的产品。需求分析过程建立在与用户充分沟通的基础上,用户有时候难以准确、充分地表达他们的需求,更不知道如何利用计算机去实现自己想要的产品。因此,需求分析过程是嵌入式系统设计和开发的重中之重,比较理想的状态是用户能够密切配合,需求分析人员和用户之间能够有效沟通(最好是能够合伙办公一段时间),最终由系统需求分析师整理、总结和归纳用户需求,形成需求规格说明书并得到用户的确认。需求规格

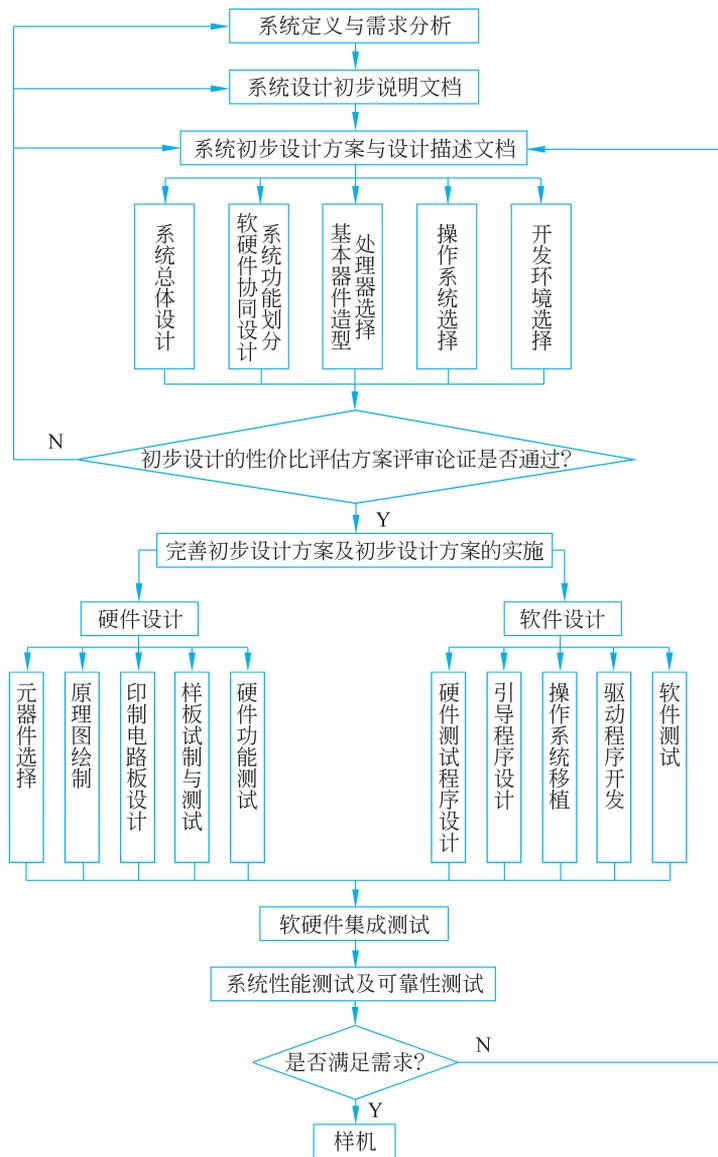


图 3-5 嵌入式系统开发流程

说明书不但面向用户,更重要的是面向系统的设计开发人员,它构成了系统的逻辑模型,是后续设计和实现工作的基础,它必须完整、清晰地体现用户需求。

(2) 系统设计方案的初步确立。确定系统需求之后,项目组就要完成对系统的初步设计工作并撰写与系统设计相关的一系列文档,包括系统概要设计说明书、系统总体设计方案、功能模块划分说明、软硬件协同设计方案、处理器选择与基本接口设计说明、操作系统选择与开发环境搭建说明等初步说明文档。文档内容反映对系统进行初步设计的具体工作,在设计过程中可能会用到流程图、成本效益分析等理论和工具。在此基础上,才有可能完成系统的详细设计方案。

(3) 设计方案评审论证。在系统开始软硬件开发之前,需要召开由项目组领导、用户以及领域内专家等参加的评审论证会,旨在确认系统设计方案的正确性、无歧义性、安全性、可靠性、可验证性、可理解性、可修改性等多个方面,评审论证通过后才可进入下一步的具体实施阶段。

(4) 项目实施。进一步完善设计方案,安排人力资源开始项目实施。除了技术问题,项目的实施过程实际也是一个管理过程。实践经验表明,对过程的跟踪管理是保证项目质量的有效手段,成熟的商业公司都积累了丰富的项目管理经验。系统在设计之初的软硬件划分对项目实施有着巨大影响,特别对软件部分的设计与实现影响很大,对系统性能和成本也起着决定性的作用。一般来说,依靠硬件实现的部分往往能够获得更高的性能,对于性能要求较低的部分功能可以通过软件来实现。

(5) 软硬件集成测试。软硬件集成测试是将软件系统下载到硬件系统进行综合测试,以验证系统功能是否正确无误。与单纯的软件产品不同,软硬件集成测试在嵌入式系统开发中往往较为复杂且更为费时,特别需要相应的工具支持才有可能确保系统的稳定运行。

(6) 性能测试及可靠性测试。测试系统性能,确保系统运行可靠,能满足任务设计书中所规定的各项指标。如果测试结果中发现问题,就需要走 Bug 修复流程。如果软件的性能和可靠性满足要求,就可以将软件彻底固化在硬件中。

3.6 嵌入式系统的发展趋势

以移动终端、智能物联网、消费电子、信息家电、5G 通信技术为代表的物联网时代不仅为嵌入式市场展现了美好的应用前景,拓展了应用场景,也对嵌入式技术提出新的挑战,主要有:更加灵活的网络连接需求、轻盈便捷的移动应用、更加强大的多媒体信息处理能力、更低功耗要求、更加友好的人机交互能力、二次开发与动态升级能力、智能化等。面对这些需求,嵌入式系统未来的发展可能会在如下几个方面有所体现。

1. 向着标准化方向发展,逐渐形成行业标准

不同于 PC 只有一两种平台,嵌入式系统在发展过程中积极吸取 PC 发展经验,可能会形成不同的行业标准。统一的行业标准具有开发、技术共享、软硬件可重用、组件兼容、维护方便、便于合作开发等诸多优势,是增强行业竞争力的有效手段。

在飞行器控制、工业控制、汽车电子、智能物联网等行业最有可能出现嵌入式开发标准。比如,欧盟汽车产业联盟规定以汽车电子开放式系统及其接口(德文: Offene Systeme and deren Schnittstellen für die Elektronik im Kraftfahrzeug, OSEK)作为开发汽车嵌入式系统的标准。航空电子工程委员会(Airlines Electronics Engineering Committee, AEEC)制定了航空电子嵌入式实时操作系统应用编程接口——ARINC653。我国的电视产业联盟也在制定本行业的开放式软件标准,以提高中国数字电视产品的竞争力。走行业开放系统道路、建立行业性的嵌入式软/硬件开发平台,是加快嵌入式系统发展的捷径之一。

2. 多核处理器的应用范围越来越广

多核处理器把多个处理器核心集成到同一个芯片上。芯片上更高的通信带宽和更短的通信时延使多核处理器在并行性方面具有优势,通过动态调节电压、负载优化分布等策略可有效降低功耗。

多核处理器有同构和异构之分。同构多核处理器将多个结构相同的处理器核心集成在同一个芯片中,已经广泛应用在 PC 领域;而异构多核处理器将 CPU 内核和 DSP、FPGA 等功能模块集成在同一个芯片中。异构多核处理器是嵌入式系统发展的一个重要方向,将结构、功能、功耗、运算性能各不相同的多个核心集成在芯片上,并通过任务分工和划分将不同的任务分配给不同的核心,使每个核心都各司其职,计算资源进一步得到优化,降低整体功耗。

3. 更加强大的开发工具和操作系统支持

为了满足日益增长的用户需求,嵌入式系统设计师一方面采用更加强大的嵌入式处理器来增强系统处理能力。另一方面,采用嵌入式实时操作系统和强大的交叉工具开发技术来管理日益增多的系统资源,降低系统控制的复杂性,简化应用程序设计,保障嵌入式应用质量和缩短开发周期。

嵌入式系统在现有基础上,通过使用嵌入式实时操作系统,结合用户需求,可在不同型号的处理器上构建应用系统,使设计和开发的软件系统具备可移植性、可伸缩性、可裁剪、实时性强、可靠性高等优势。另一方面,嵌入式开发工具会向着多平台支持、覆盖嵌入式软件开发全过程、更加高效和高密度集成的方向发展。

4. 嵌入式系统的物联网智能应用成为必然趋势

在物联网技术日益渗透到方方面面的同时,“嵌入式系统联网”已经成为必然需求。嵌入式系统联网需要 TCP/IP 协议簇的支持,物联网时代的各类智能硬件都需要系统具备 IEEE 1394、USB、CAN、Bluetooth、WiFi 以及 IrDA 等通信接口,提供网络支撑功能将会成为嵌入式系统发展的必然趋势。

5. 出现新的嵌入式计算模型

伴随着物联网技术的进一步发展,行业数字化在敏捷连接、实时业务、数据优化、应用智能、安全与隐私保护等方面的关键需求需要新的计算模型来支撑。例如,融合网络、计算、存储、应用核心能力的边缘计算模型,具备二次开发能力的可编程嵌入式系统以及支持分布式计算的嵌入式系统等。