

第 1 章 嵌入式系统与微控制器概述

嵌入式系统作为计算机应用系统的一个分支，在后 PC（个人计算机）时代发展迅速，已经成为一个独立的研究与应用方向。近年来，随着物联网（IoT）应用系统的发展，给嵌入式系统带来了新的机遇。本章将讲述嵌入式系统相关的一些基础知识，包括嵌入式系统的概念、发展、特点、组成、种类、开发调试方法及嵌入式处理器概述等。通过对本章的学习，读者可以对嵌入式系统，特别是嵌入式微控制器（MCU）的相关基础知识有基本的了解。

1.1 嵌入式系统概述

嵌入式系统又称为嵌入式计算系统（embedded computing system），它是随着计算机技术、微处理器技术、通信技术、集成电路技术的发展而发展起来的，现已成为计算机应用领域的一个重要组成部分。嵌入式系统是一种专用的计算机系统，它通常被包含在一些机械或电子系统中。嵌入式系统通常是一种计算性能受限制的实时系统，它经常作为某一部件嵌入到一个完整设备中。如今，嵌入式系统在大量的设备中运行，约有 98% 的微处理器和微控制器被用于各种嵌入式系统。

嵌入式系统是硬件和软件的集合体。其硬件包含嵌入式处理器、存储器、外设接口器件等。其软件包括引导程序、嵌入式操作系统和应用程序等。嵌入式操作系统控制着应用程序与硬件的交互，应用程序控制着系统的运作。很多嵌入式系统还包括一些机械部分，如机电一体化装置、微机电（MEMS）系统、光学系统等，这些机械部分是为完成某种特定的功能而设计的，所以嵌入式系统有时也被称为嵌入式设备。

1.1.1 嵌入式系统的发展

嵌入式系统主要是伴随着计算机技术和微电子/集成电路技术的发展而快速发展的。20 世纪中叶，微电子技术处于发展初级阶段，集成电路属于中小规模发展时期，各种新材料新工艺尚未成熟，元件集成规模还比较小，工业控制系统基本使用继电器逻辑技术，还没有嵌入式系统的概念。直到 20 世纪 70 年代，随着微处理器的出现，计算机系统出现了历史性的快速发展。以微处理器为核心的微型计算机，以其小型、价廉、高可靠性特点，迅速走出机房，进入多种应用领域。基于高速数值运算能力的微型机，表现出的智能化水平引起了控制专业人士的兴趣，开始将微型机嵌入到一个对象体系中，实现对象体系的智能化控制。例如，将微型计算机经电气加固、机械加固，并配置各种外围接口电路，安装到大型舰船中构成自动驾驶仪或轮机状态监测系统。这样，计算机便失去了原来固有

的形态和通用计算机的功能。为了区别于原有的通用计算机系统，把嵌入到对象体系中并实现对象体系智能化控制的计算机称作嵌入式计算机系统，简称嵌入式系统（embedded systems）。

从 20 世纪 70 年代后期单片机的出现，到今天各式各样的嵌入式微处理器以及微控制器的大规模应用，嵌入式系统已经有了 40 多年的发展历史。

嵌入式系统的出现最初是基于单板机或单片机的。如用 Zilog 公司 Z80 微处理器设计的单板机（控制板），以及后来的 Intel 80386EX 单板机等。随着 20 世纪 70 年代后期单片机的出现，汽车、家电、工业机器、通信装置以及其他电子产品可以通过内嵌单片机，使系统获得更好的性能，且更便于使用，价格也更便宜。随着嵌入式系统规模的不断发展，嵌入式芯片的工艺不断改进，性能得到提高，成本逐步下降，在各个方面都有了广泛的应用。从企业应用到家庭、移动应用，嵌入式系统不断进入人们的工作和生活，市场也不断壮大。如图 1.1 所示，嵌入式微控制器出货量以指数级上升。1991—2013 年，ARM 架构的芯片总出货量达到 500 亿颗，而 2013—2017 年 4 年时间，ARM 架构的芯片总出货量就达到了 500 亿颗。2017—2021 年，ARM 架构的芯片出货量又高速增长，超过了 1000 亿颗。

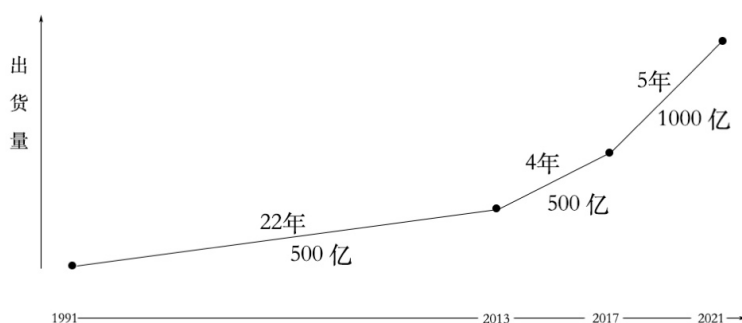


图 1.1 ARM 微控制器出货量

在嵌入式软件方面，从 20 世纪 80 年代早期，嵌入式系统开发的程序员开始使用高级语言和商业级的“操作系统”编写嵌入式应用软件，这可以缩短开发周期、降低开发成本，并提高开发效率。确切来说，那个时候的操作系统仅是一个任务管理内核，这个内核包含了一些传统操作系统的特征，包括任务管理、任务间通信、同步与相互排斥、中断支持、内存管理等功能。

20 世纪 90 年代以后，随着对实时性要求的提高，软件规模不断上升，操作系统内核逐渐发展为实时操作系统（RTOS），并作为一种基础软件平台，逐步成为嵌入式系统开发的主流。更多的公司看到了嵌入式系统的广阔发展前景，开始大力发展自己的嵌入式操作系统，如 $\mu\text{C}/\text{OS-II}$ 、FreeRTOS、Palm OS、WinCE、VxWorks、嵌入式 Linux 等操作系统。2010 年以后，随着物联网（IoT）时代的到来，又出现了一些包含物联网应用协议的物联网操作系统，如 MiCO、AliOS Things、LiteOS 等。

1.1.2 嵌入式系统的特点

嵌入式系统是一种满足特定应用需求的计算机应用系统，其最主要的特点就是专用性。

一个嵌入式系统的功能与非功能指标（包括外型、体积等）、硬件与软件，都是特定设计的，冗余度很小。另外，由于嵌入式系统应用面广、需求各异，使得各种嵌入式系统的软硬件复杂度差异很大，行业难以被垄断。

嵌入式系统的硬件核心是嵌入式处理器。嵌入式处理器一般具备以下几个特点。

- 性能、功能差异很大且覆盖面广。这是由嵌入式系统应用特点决定的，各种嵌入式应用对处理器的要求差异非常大，需要不同性能、功能的处理器来满足。
- 对实时多任务有很好的支持。支持较多的中断源，并且有较短的中断响应时间，从而使内部的代码和实时内核的执行时间减少到最低限度。
- 具有功能很强的存储区保护功能。对于多任务的应用，由于嵌入式系统的软件已经模块化，为了避免在软件模块之间出现错误的交叉作用和相互影响，需要有存储区保护功能，同时也有利于软件诊断。
- 可扩展的处理器结构、工具链完善。可以快速开发出满足应用的、具有不同性能的嵌入式处理器。在这方面，ARM 处理器有很大的优势，RISC-V 处理器也在崛起。
- 低功耗。功耗在某些嵌入式应用中是有严格要求、设计受限的，尤其是用于电池供电的便携式无线和移动设备中的嵌入式处理器，平均功耗只有 mW 甚至 μW 级。

一般地，复杂嵌入式系统的软件包括嵌入式操作系统和应用程序。对于一些简单应用的小系统，应用程序可以没有操作系统而直接在处理器上运行。对于复杂的大系统来说，为了合理地调度和管理多任务、系统资源、系统函数，以及与一些专家库函数接口，用户需要选择与嵌入式系统相对应的开发平台，这样才能保证程序执行的实时性、可靠性，并减少开发时间，保障软件质量。

1.1.3 嵌入式系统的组成

作为一个“专用计算机系统”的嵌入式系统，同样也是由硬件系统和软件系统两大部分组成的。一般来说，硬件系统包括处理器、存储器、I/O 外设器件、专用控制器（如图形、存储控制器）等。软件部分包括操作系统软件（一般要求实时和多任务操作）和应用软件。有时为了追求更高的执行效率，设计人员也会把这两种软件组合在一起。嵌入式系统使用的操作系统可能是相同的，但根据应用领域的不同，应用程序（应用软件）却可以千差万别。应用软件控制着系统的运作和行为，而操作系统则控制着应用程序与硬件的交互作用。

嵌入式设备一般还包括其他一些电子、机械部分，如马达驱动、电机及传动、光学系统等，以满足特殊的功能需求。此书所讲述的嵌入式系统，仅指以嵌入式处理器为核心的嵌入式控制与接口部分内容。

1. 嵌入式系统硬件部分

嵌入式系统的硬件结构如图 1.2 所示，硬件部分的核心部件就是嵌入式处理器（CPU）。本书中主要介绍的 ARM Cortex-M 处理器就是一个典型的嵌入式处理器。

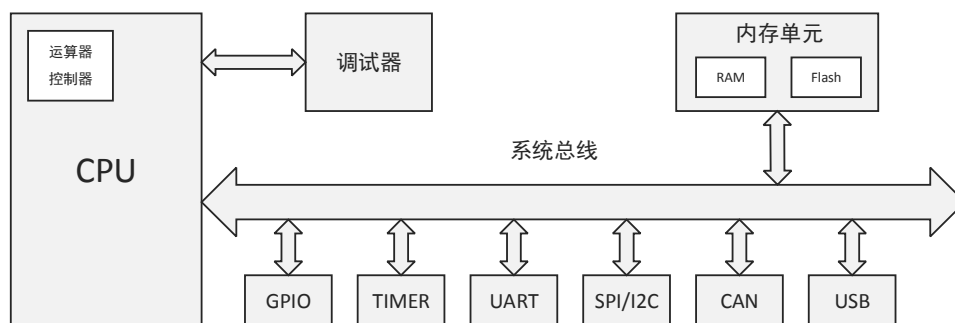


图 1.2 嵌入式系统硬件结构示意图

目前，全世界嵌入式处理器的型号已经超过 1000 种，流行的体系结构也有 10 多个，主流嵌入式处理器的字长从 8 位到 32 位、寻址空间从 64KB 到 4GB 不等，其处理速度可以从 0.1MIPS^①到 2000MIPS。一般来说，可以把嵌入式处理器分成 MPU（micro processor unit）微处理器、MCU（micro controller unit）微控制器、DSP（digital signal processor）数字信号处理器、SoC（system on chip）片上系统 4 类。

嵌入式系统的存储器由于系统规模不同而差异很大。通常 MCU 系统都使用 MCU 自带的 Flash 和 SRAM，代码固化在 Flash 中且可直接执行，无须外扩存储器。而 MPU 系统则需外扩 SDRAM 作为系统内存，外扩 NAND Flash 作为系统外存。MPU 系统的程序平时保存在外存，需要运行时由 OS 调入到内存再执行。

嵌入式系统硬件部分除了嵌入式处理器和存储器，最有特色的就是各种外设接口。也正是基于这些丰富的外设接口，才带来嵌入式系统越来越丰富的应用。现在的 ARM 微控制器内部的外设接口已非常丰富，如 GPIO、定时器、I2C、SPI、UART 和 USB 等接口基本上都是“标准”配置。在设计系统时，通常只要把微控制器和相应接口的外部设备进行简单的物理连接，就可以实现外部器件和设备的扩展。

随着高度集成化技术的发展，嵌入式处理器集成的外设越来越多，功能也越来越强，需要扩展的外部设备/接口电路变得越来越少了，整个硬件系统设计也就变得越来越简单了。比如，很多 ARM 微控制器里面就已经集成了 Flash、SRAM 和很多通信接口，有的还在其内部集成了存储控制器、网络控制器和 LCD 控制器等。

2. 嵌入式系统软件部分

简单嵌入式系统的软件可以直接在裸机上执行一个应用程序，复杂嵌入式系统软件一般是由嵌入式操作系统和应用软件两部分组成的。具体来说，嵌入式系统软件可以分成启动代码（Bootloader）和板级支持包（board support package, BSP）、操作系统内核与驱动（kernel & driver）、文件系统和应用程序（file system & application）等几部分。

Bootloader 是嵌入式系统的启动代码，主要用来初始化处理器、传递内核启动参数给嵌入式操作系统内核，使得内核可以按照我们的参数要求启动。另外 Bootloader 通常都具有搬运内核代码到 RAM 并跳转到内核代码地址运行的功能。

^① MIPS 全称为 million instruction per second，表示单字长定点指令平均执行速度，是每秒处理的百万级的机器指令数。

板级支持包（BSP）则完成不同硬件与操作系统接口的软件映射。

操作系统内核主要有 4 个功能：任务管理、任务间通信与同步、内存管理及 I/O 资源管理。驱动程序属于内核的一部分，主要是提供给上层应用程序一套可以通过处理器访问硬件外设接口和外部设备的软件接口（API）。

文件系统则可以让嵌入式软件工程师灵活方便地管理和使用系统存储资源，包括其他系统软件/中间件，如 GUI、网络协议栈等，这些都是按需选用的。

应用程序是真正针对需求的，同时可能是嵌入式软件工程师完全自主开发的。

总的来说，嵌入式系统的硬件部分可以说是整个系统的基石，嵌入式系统的软件部分则是在这个基石上面建立起来的满足不同功能的楼宇。对于任何一个需求明确的嵌入式系统来说，二者缺一不可。在对系统做了相对完整而细致的需求分析之后，通常采用软件和硬件同步进行的方式来开发，前期硬件系统的设计要比软件系统设计稍微提前，到了后期，软件系统的开发工作量会比硬件系统的开发工作量大很多。

1.1.4 嵌入式系统的种类

由于分类标准的差异，嵌入式系统的分类方法也不同。本小节根据嵌入式系统的软件复杂程度，将嵌入式系统分为以下 3 类。一个嵌入式系统的软件复杂程度一般也对应硬件的复杂程度。

1. 无操作系统支持的嵌入式系统

无操作系统支持的嵌入式系统，其软件部分规模较小，通常是一个无限循环结合中断的前后台程序，无须额外的 CPU 和存储开销。这类嵌入式系统一般适用于结构相对简单、功能单一的嵌入式应用，如家用电器、电子玩具、简单的仪器仪表和控制等。

2. 小型操作系统支持的嵌入式系统

小型操作系统支持的嵌入式系统，其软件部分一般由一个小型实时操作系统内核（RTOS）和一个小规模应用程序组成。小型实时操作系统内核的源代码一般不超过 1 万行，功能模块较少，一般只具有任务管理功能、文件系统、网络协议栈、图形用户界面等模块。这类嵌入式系统可靠性高、硬件资源消耗少，适用于人机交互相对简单、功能不太复杂的多任务应用，如基于 RTOS 的 POS 机、UPS（uninterruptible power system，不间断电源）、智能仪器仪表等。

3. 完整操作系统支持的嵌入式系统

带完整操作系统的嵌入式系统，其软件部分的核心是一个功能齐全的嵌入式操作系统（如 Linux、VxWorks、Android 等），包含文件系统、网络协议、封装良好的 API 和 GUI，可靠性高，可运行多个数据处理功能较强的应用程序。这类嵌入式系统对系统硬件资源消耗相对较高，适用于需要良好人机交互或网络功能的复杂应用，如基于 Linux 或 Android 的机顶盒、智能电视和智能手机等。

1.1.5 嵌入式系统的调试方法

调试是嵌入式系统开发过程中必不可少的环节。在嵌入式系统的调试过程中，调试器（debugger）和被调试程序（debugee）通常运行在不同的机器上，调试器运行于宿主机，被调试程序运行于目标机。调试器通过某种方式可以控制被调试程序的运行方式，并能查看和修改目标机上的内存、寄存器以及被调试程序中的变量等。嵌入式系统的这种调试被称为交叉调试，具有以下特点。

- 调试器和被调试程序运行在不同的机器上。调试器运行在一般的 PC 或者工作站（即宿主机），被调试程序运行在实际的嵌入式设备上（即目标机）。而对于通用计算机应用系统，调试器和被调试程序通常运行在同一台计算机上，例如，在 Windows 平台上使用 Visual C++ 等语言开发应用，调试器进程通过操作系统提供的接口来控制被调试程序。
- 宿主机上的调试器通过某种通信方式与目标机上的被调试程序建立物理连接。常见的通信方式有串口、以太网口、USB-JTAG 等。
- 在目标机上一般有调试器的某种代理（agent），这种代理可以是某种软件（如监视器），也可以是某种支持调试的硬件（如 JTAG），用于解释和执行目标机接收到的来自宿主机的各种命令（如设置断点、读内存、写内存等），并将结果返回给宿主机，配合宿主机调试器来调试目标机上的程序。
- 目标机也可以是一种虚拟机。在这种情形下，调试器和被调试程序似乎运行在同一台计算机上。但是调试方式的本质没有变化，即被调试程序都是被下载到目标机，对被调试程序的调试并不是直接通过宿主机操作系统的调试来完成，而是通过虚拟机代理的方式来完成。

交叉调试的方式一般分为以下几种。

1. 软件模拟器

软件模拟器是运行在宿主机上的纯软件工具，它通过模拟目标机的指令系统或目标机操作系统的系统调用，以此来达到在宿主机上运行并调试嵌入式应用程序的目的。

软件模拟器分为两类：指令集模拟器和系统调用级模拟器。

（1）指令集模拟器

指令集模拟器是在宿主机上模拟目标机的指令系统。它相当于在宿主机上建立了一台虚拟的目标机，该目标机的 CPU 型号与宿主机的 CPU 不同，例如，宿主机的 CPU 是 Intel Pentium，虚拟目标机的 CPU 可能是 ARM。功能强大的指令集模拟器不仅可以模拟目标机的指令系统，还可以模拟目标机的外设，如串口、网口、键盘等。

ARMulator 是由 ARM 公司早期推出的面向集成开发环境 ADS（ARM developer suite）中提供的指令集模拟器。它与运行在通用计算机（通常是 x86 体系结构）上的调试器相连接，可以模拟 ARM 微处理器体系结构和指令集，并提供开发和调试 ARM 程序的软件仿真环境。目前，Keil MDK、IAR EWARM 等集成开发环境（IDE）也都提供类似的软件仿真

功能，嵌入式软件工程师无须 ARM 开发板，在 IDE 中借助软件仿真功能即可对 ARM 源代码进行仿真调试。软件仿真一般应用于验证软件结构和一些算法，无法对硬件异步触发和时序逻辑做出精确仿真。

(2) 系统调用级模拟器

系统调用级模拟器是在宿主机上模拟目标机操作系统的系统调用。它相当于在宿主机上安装了目标机的操作系统，使得基于目标机操作系统的应用程序可以在宿主机上运行。目前，常用的系统调用级模拟器有 Android 模拟器和 iOS 模拟器等。

总而言之，软件模拟器可以在无须硬件支持的情况下，借助开发工具提供的虚拟平台进行软件开发和调试，使得嵌入式系统的软件和硬件可以并行开发，以提高嵌入式开发的效率、降低开发的成本。但使用软件模拟器调试，模拟环境与实际运行环境差别较大，被调试程序的执行时间与在目标机真实执行环境中的执行时间差别较大。而且除了常见的设备，不能模拟目标机所有的外围设备，一般仅用于嵌入式开发的早期阶段。

2. ROM 监控器

ROM 监控器也是一种早期的调试方法。如图 1.3 所示，在 ROM 监控器方式下，嵌入式系统的调试环境由主机端的调试器、目标机端的监控器（ROM monitor）以及二者间的连接（包括物理连接和逻辑连接）构成。ARM 公司的 Angel 即属于此类调试方式，ARM 公司提供的各种调试工具包均支持基于 Angel 的调试方式。

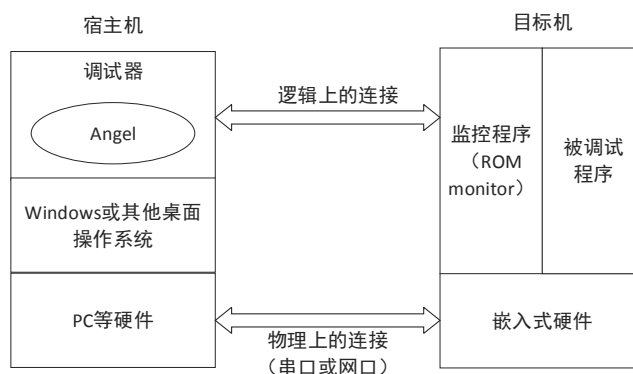


图 1.3 ROM 监控器调试环境

ROM 监控器方式下，调试器大部分驻留在宿主机，余下的部分驻留在目标机作为调试代理。驻留在目标机的部分称为 ROM 监控器，是被固化在目标机的 ROM 中且目标机复位后首先被执行的一段程序。它对目标机进行一些必要的初始化，同时初始化自己的程序空间，然后通过指定的通信端口并遵循远程调试协议等待宿主机端调试器的命令，例如，被调试程序的下载、目标机内存和寄存器的读/写、设置断点和单步执行被调试程序等，监控目标机上被调试程序的运行，与宿主机端的调试器一起完成对目标机上应用程序的调试。

综上所述，ROM 监控器调试方式简单方便，可以支持多种高级调试功能，如代码分析和系统分析等，而且成本低廉，不需要专门的硬件调试和仿真设备。但它本身要占用目标机的一部分资源（CPU、ROM 和通信资源等），且当 ROM 监控器占用目标机 CPU 时，应

用程序无法响应外部中断，不便于调试有时间特性的应用程序。

3. ROM 仿真器

ROM 仿真器，英文全称 ROM emulator，可以认为是一种用于替代目标机上的 ROM/Flash 芯片的工具。如图 1.4 所示，ROM 仿真器的外形是一个有两根电缆的盒子，一边通过 ROM/Flash 芯片的插座与目标机相连，另一边通过通信口与宿主机相连。对于目标机上的 CPU，ROM 仿真器就像一个只读存储器芯片，这样目标机就可以不设置 ROM 芯片，而是利用 ROM 仿真器提供的 ROM 空间来代替。而对于宿主机上的调试器，ROM 仿真器上的 ROM 芯片的地址可以实时映射到目标机 ROM 的地址空间，从而仿真目标机的 ROM。

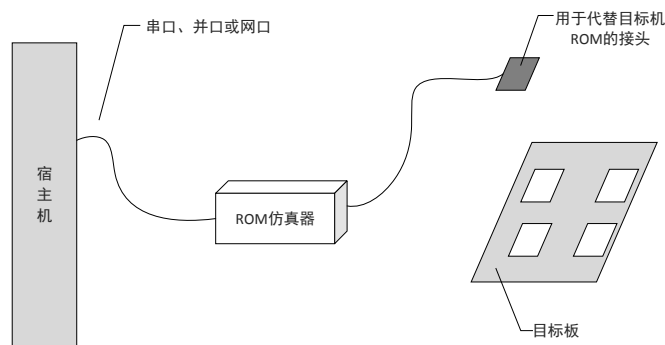


图 1.4 ROM 仿真器调试环境

实质上，ROM 仿真器是一种不完全的调试方式。虽然避免了每次修改程序后都必须重新烧写到目标机 ROM 中这一费时费力的操作，但 ROM 仿真器设备通常只是为目标机提供一个 ROM 芯片的替代，并在目标机和宿主机间建立一条高速的通信通道，因此，它经常与 ROM 监控器结合起来形成一种完备的调试方式。

4. 在线仿真器

在线仿真是最直接的仿真调试方法。在线仿真器（in-circuit emulator, ICE）是一种用于替代目标机上的 CPU 来模拟目标机上 CPU 行为的设备。它有自己的仿真 CPU、RAM 和 ROM，可以执行目标机 CPU 的指令，不再依赖目标机的处理器和内存。

如图 1.5 所示，使用 ICE 调试前，要完成 ICE 和目标机的连接，通常先将目标机的 CPU 取下，然后将 ICE 的 CPU 引出线接到目标机的 CPU 插槽中。调试时，目标机的应用程序驻留在目标机的内存中，监控器即调试代理驻留在 ICE 的存储器中，使用 ICE 的 CPU 和存储器、目标机的输入输出接口调试目标机内存中的应用程序。调试完成后，再使用目标板上的处理器和存储器运行应用程序。

在线仿真器调试方法具有以下特点。

- 在线仿真器能同时支持软件断点和硬件断点的设置。通常，软件断点只到指令级别，在目标机的被调试程序中，软件断点只能指定在取某一指令时停止运行。而在硬件断点方式下，多种事件的发生都可以使目标机的被调试程序在一个硬件断点上停止运行，这些事件包括内存读/写、I/O 读/写以及中断等。

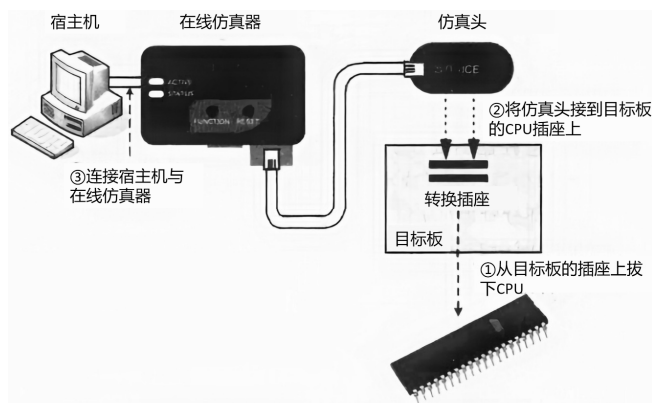


图 1.5 在线仿真器调试环境

- ❑ 在线仿真器能设置各种复杂的断点和触发器。例如，可以指定目标机的被调试程序在“当变量 var 等于 80 且寄存器 R1 等于 1”时停止运行。
- ❑ 在线仿真器能实时跟踪目标机的被调试程序的运行，并可实现选择性跟踪。在 ICE 上有大量 RAM，专门用来存储执行过的每个指令周期的信息，使用户可以得知各个事件发生的精确次序。
- ❑ 在线仿真器能在不中断目标机的被调试程序运行的情况下查看内存和变量，即可实现非干扰的调试查询。

在线仿真器是较为有效的嵌入式系统调试方式，尤其适合调试实时应用系统、硬件设备驱动程序以及对硬件进行功能测试。目前，在线仿真器一般用于低速和中速的嵌入式系统，例如，大多数 8 位 MCS-51 单片机仿真器。但是在 32 位高速嵌入式处理器领域，过高的时钟频率和复杂的芯片封装形式导致其对 ICE 的技术要求很高，价格也非常昂贵，因此在线仿真器目前已很少使用。

5. 片上调试

由于传统的 ICE 难以满足高速嵌入式系统，越来越多的嵌入式处理器（如 ARM 系列）借助片上调试（on-chip debugging, OCD）技术进行嵌入式系统的调试。

片上调试是内置于目标板 CPU 芯片内的调试模块提供了一种调试功能，可以把它看成是一种廉价的 ICE 功能，它的价格只有 ICE 的几十分之一，却提供了几乎全部的 ICE 功能。OCD 采用两级模式，即将 CPU 的工作模式分为正常模式和调试模式。在正常模式下，目标机的 CPU 从内存读取指令执行。如图 1.6 所示，在调试模式下，目标机的 CPU 从调试端口读取指令，通过调试端口可以控制目标机的 CPU 进入和退出调试模式。这样宿主机的调试器可以直接向目标机发送要执行的指令。通过这种形式来读写目标机的内存和各种寄存器，并控制目标被调试程序的运行以及完成各种复杂的调试功能。

OCD 价格低廉，不占用目标机的资源，调试环境与程序最终运行环境基本一致，支持软硬件断点，可以精确计算程序的执行时间，并提供实时跟踪和时序分析等功能。但是 OCD 也存在以下不足之处：调试的实时性不如 ICE 强；不支持非干扰的调试查询，即无法在不中断调试程序运行的情况下查看内存和变量；使用范围受限，即不支持没有 OCD 功能的 CPU。

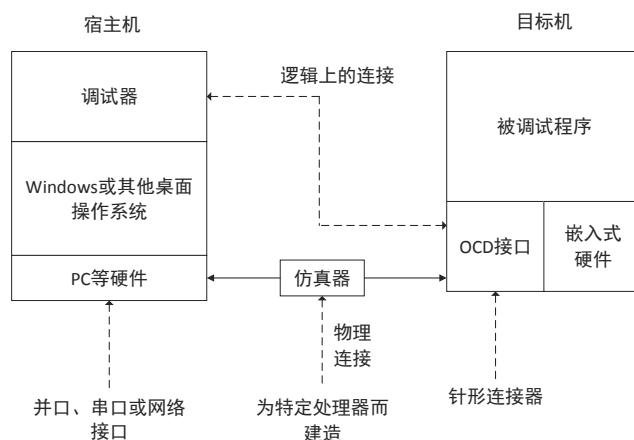


图 1.6 片上调试方式下的调试环境

现在比较常用的OCD实现有后台调试模式（background debugging mode, BDM）、联合测试工作组（joint test access group, JTAG）和片上仿真器（on chip emulation, ONCE）等。其中，JTAG是目前主流的OCD方式，也是ARM处理器一般都具备的调试接口。

（1）JTAG 标准

JTAG 是一种关于测试访问端口和边界扫描结构的国际标准，由联合测试工作组（JTAG）提出，于1990年被电气和电子工程师协会（Institute of Electrical and Electronics Engineers, IEEE）批准为IEEE 1149.1规范，也被称为JTAG标准或JTAG协议，用于芯片内部测试及对程序进行调试、下载。它在内部封装了专门的测试电路——测试访问端口（test access port, TAP），通过专用的JTAG仿真器对内部节点进行测试。JTAG是JLink在线调试的一种模式。

JTAG是一个开放的协议，目前被全球各大电子企业广泛采用，已经成为电子行业内片上测试技术的一种标准。现在，大多数嵌入式处理器都支持JTAG标准，如32位ARM处理器，不论出自哪个半导体厂商，都采用兼容的JTAG接口。具有JTAG接口的芯片（如ARM处理器）都有若干个JTAG引脚。

JTAG在线调试的另外一种简化模式是SWD（serial wire debug, 串行线调试）。图1.7是JTAG和SWD标准的接口排列，相比于JTAG接口，SWD进一步减少了调试所占用的MCU引脚数。它只需要串行时钟线SWCLK和串行数据线SWDIO两个硬件引脚信号即可完成嵌入式系统的调试。目前主流的MCU和调试工具一般都支持SWD调试模式。

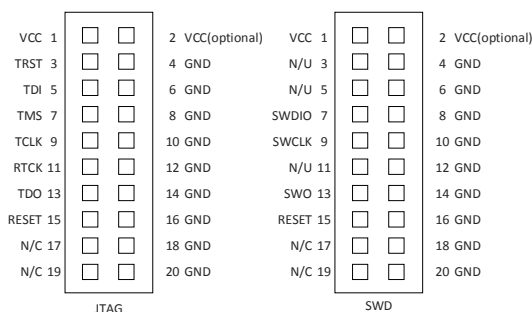


图 1.7 JTAG 和 SWD 接口定义

表 1.1 给出了 JTAG/SWD 调试接口的引脚描述。其具体描述如表 1.1 所示, 其中, JTAG 引脚定义包括 TDI 和 TDO 引脚, 因此数据流从进入 CPU 核心到输出 CPU 核心会形成一个很长的循环。

表 1.1 JTAG (IEEE 1149.1) /SWD 接口的引脚描述

引 脚	描 述
TCK/SWCLK	同步 JTAG 端口逻辑操作的时钟输入
TMS/SWDIO	测试模式选择输入, 在 TCK 的上升沿被采样到内部状态机控制器 (TAP 控制器) 序列
TDI	输入测试数据流, 在 TCK 的上升沿被采样
TDO	输出测试数据流, 在 TCK 的下降沿被采样
TRST	低位有效的异步复位

(2) 基于 JTAG/SWD 的嵌入式调试环境

基于 JTAG/SWD 的嵌入式调试环境中, 目标机上含有 JTAG/SWD 接口模块的 CPU, 通过 JTAG 仿真器与宿主机相连。只要目标机 CPU 的时钟系统正常, 嵌入式开发者即可利用宿主机上嵌入式集成开发工具中的调试工具程序, 通过 JTAG/SWD 接口使用独立目标机 CPU 指令系统的 JTAG 命令访问 CPU 的内部寄存器和挂载在 CPU 总线上的设备, 如 Flash、RAM 和内置模块 (如 GPIO、Timer 和 URAT 等) 的寄存器, 达到调试的目的。

JTAG 的嵌入式调试使用测试访问端口和边界扫描技术与目标机的 CPU 通信。与 ROM 监控器的调试方式相比, 基于 JTAG 的嵌入式调试方式不仅功能强大, 而且无须目标存储器, 不占用目标机用户资源; 与在线仿真器的调试方式相比, 它的成本非常低廉。因此, 在宿主机上使用嵌入式集成开发工具配合 JTAG 仿真器进行的基于 JTAG/SWD 的嵌入式调试, 已成为嵌入式系统目前最有效、使用最广泛的一种调试方式。

(3) JTAG 仿真器

JTAG 仿真器又称为 JTAG 适配器, 是基于 JTAG 的嵌入式调试环境中不可或缺的环节。通常它一边通过 USB 接口与宿主机连接, 一边通过 JTAG/SWD 接口与目标机的芯片 (通常是 CPU) 连接, 将宿主机调试工具软件的调试命令解析成 JTAG/SWD 的信号时序 (即协议转换), 以设置 TAP (test access port, 测试存取端口) 控制器的工作状态, 控制对边界扫描寄存器的操作, 完成对目标机的芯片的调试工作。

JTAG 仿真器不仅是嵌入式程序调试的重要工具, 也是嵌入式软件固化 (烧录) 的工具。嵌入式软件固化是指将调试完毕的二进制可执行映像文件烧写到目标机的非易失性存储器 (Flash) 中, 这个工作往往需要借助专门的烧写设备和烧写软件来完成。对于不支持 JTAG 的 CPU (如 MCS-51 等), 通常需要使用被称为“编程器”的专用硬件设备和宿主机上的烧写软件来完成嵌入式软件的固化工作。对于支持 JTAG/SWD 的 CPU (如 ARM MCU 等), 只需通过 JTAG/SWD 接口连接 JTAG 仿真器, 借助宿主机的调试工具或烧写工具即可完成嵌入式软件的固化工作。

1.1.6 嵌入式系统的应用

嵌入式系统在消费类电子产品、仪器仪表、工业自动化、国防、运输和航空航天领域

等产业中有广泛的应用。分类简述如下。

1. 消费电子

嵌入式系统在消费类电子产品应用领域的发展较为迅速，需求量也较大。由嵌入式系统构成的消费类电子产品已经成为现实生活中必不可少的一部分。各种智能家电、智能音箱、流媒体电视等信息家电产品，以及大家熟悉的智能手机、智能手表等都是具有不同处理能力和存储需求的嵌入式系统。

2. 智能仪器、仪表

这类产品广泛应用于工业现场，很多产品对于开发人员来说也是必备工具，如网络分析仪、数字示波器、热成像仪等。通常，这些嵌入式设备中都有一个应用处理器和一个运算处理器，可以完成一定的数据采集、分析、存储、打印、显示等功能。

3. 网络通信

这些产品多数应用于通信机柜设备中，如路由器、交换机、家庭媒体网关等。

4. 过程控制

主要指在工业控制领域中的应用，包括自动化生产过程中各种动作流程的控制，如流水线控制与检测、金属加工控制、汽车电子等。

5. 航空航天

航空航天等领域中也需要大量的嵌入式系统，如火星探测器、火箭发射主控系统、卫星信号测控系统、飞机的控制系统、探月机器人等。

6. 生物电子医学系统

如今在医学保健行业取得的较大进步，很多都应归功于嵌入式系统的应用与发展，如X光机的控制部件、EEG和ECG设备、CT、超声波检测设备、核磁共振设备、结肠镜和内窥镜等。除此之外，嵌入式医疗设备和保健设备也在不断发展，如家用的心电监测设备等。

近年来，随着物联网技术和应用的发展，嵌入式系统又进入了网络化新时代。本书后续章节将介绍物联网相关技术，以及两个物联网应用示例。

1.2 嵌入式处理器概述

本节将简要介绍嵌入式处理器的分类和特点。嵌入式处理器各有千秋，其应用范围也有很大不同。本节将重点介绍嵌入式微控制器（MCU）的发展及特点。

1.2.1 嵌入式处理器

嵌入式处理器是嵌入式系统的核心部件，是控制系统运行的硬件单元，其功耗、体积、成本、可靠性、速度、处理能力、电磁兼容性等方面均受到应用要求的制约。从目前仍在

大规模应用的 8 位单片机，到广受青睐的 32 位、64 位嵌入式 CPU，以及未来发展方向之一的多核处理器，嵌入式处理器应用范围越来越广泛。目前全世界嵌入式处理器的品种总量已经超过 1000 种，流行体系结构包括 MCU、MPU 等 10 多个系列。鉴于嵌入式系统广阔的发展前景，很多半导体制造商都开始大规模生产嵌入式处理器，如单片机、DSP、FPGA（field-programmable gate array，现场可编程逻辑门阵列）等都有着各式各样的品种和型号。处理器也从之前的单核向多核、SoC 方向发展，速度越来越快，性能越来越强，价格也越来越低。

1.2.2 嵌入式处理器的分类

根据不同的应用场景和需求，嵌入式处理器主要分成以下几类。

1. 嵌入式微处理器

嵌入式微处理器（micro-processor unit, MPU）是由通用计算机的 CPU 演变而来的。它在 CPU 的基础上，只保留与嵌入式应用功能紧密相关的硬件，去除其他不必要的部分，这样就能够以较低的功耗和资源满足嵌入式应用的特殊要求。目前，MPU 以 ARM Cortex-A、MIPS、X86 等内核为主，其工作主频一般都大于 500MHz，外扩大容量 SDRAM，运行 Linux、Android、iOS 等完整的操作系统。与通用 CPU 相比，嵌入式微处理器具有体积小、成本低、可靠性高等优点，显著特征是具有 32 位以上的处理能力和较高的性能。下面以微处理器 Am335X 为例介绍嵌入式微处理器的主要构成和特点。Am335X 是 TI（德州仪器）基于 ARM Cortex-A8 内核设计的一款微处理器，该处理器增强了在图像、图形处理和诸如 PROFIBUS（程序总线网络）等工业接口方面的性能。图 1.8 所示是 Am335X 的功能框图。

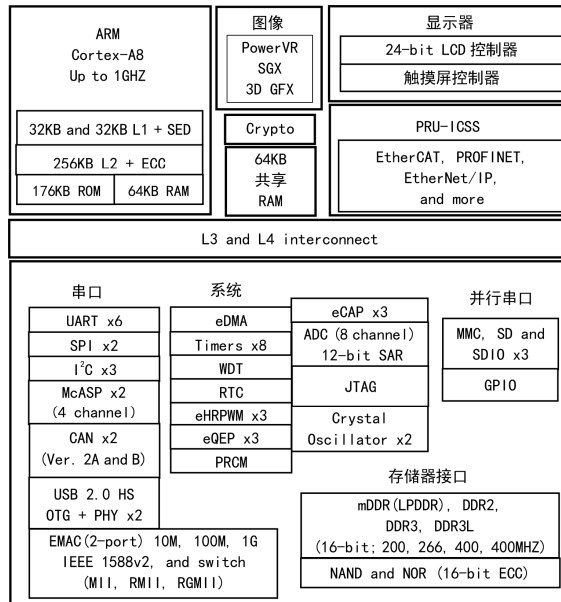


图 1.8 Am335X 功能框图

该处理器有以下特点。

- ❑ 采用 ARM Cortex-A8 内核，最高主频可达 720MHz。
- ❑ 带 NEON 协处理器，二级高速缓存。
- ❑ 带 24 位 LCD (liquid crystal display, 液晶显示器) 控制器和触摸屏控制器，分辨率高达 2048×2048。
- ❑ 带两个具有集成 PHY (physical layer, 物理层) 的 USB2.0 高速 OTG (on The Go)。
- ❑ 最多支持 6 个 UART (universal asynchronous receiver/transmitter, 通用异步收发传输器)。
- ❑ 集成两个工业用途千兆以太网 MAC (10/100/1000MHz)。
- ❑ 多达两个控制器局域网 (CAN) 端口，支持 CAN2.0 A 和 B。
- ❑ 集成两个 PRU (pseudo-random upstream, 上行伪随机序列) 模。
- ❑ 2 路多功能音频通道。
- ❑ 多路 SPI (serial peripheral interface, 串行外设接口)、IIC (inter-integrated circuit, 集成电路总线)、定时器、PWM (pulse width modulation, 脉冲宽度调制)、DMA (direct memory access, 直接内存存取)、RTC (real-time clock, 实时时钟芯片) 等常用外设。
- ❑ 自带 SGX530 3D 图形加速引擎。

2. 嵌入式微控制器

嵌入式微控制器 (micro controller unit, MCU) 是目前嵌入式系统应用的主流，它内部集成 ROM/Flash、EPROM、RAM、定时器、I/O、通信接口等各种必要功能和外设。和微处理器相比，无论是在品种数量还是生产厂商方面，微控制器都占上风。微控制器的最大特点是单片化、体积小，功耗和成本低，可靠性高。

目前，MCU 主要有 ARM Cortex-M、PIC、MSP430、MCS-51 等内核，其工作主频大多在 8~200MHz，内嵌 SRAM (几百字节至几百千字节)，可在 RTOS 或无操作系统 (裸机) 环境上运行。现代 MCU 朝着更高性能和更低功耗方向发展，一些新的技术、外设正在应用到 MCU 中。物联网时代为 MCU 创造了大量新的应用机会，MCU 产品性能的提高也使得其应用范围更加广泛、市场更加多样化。本书后续章节将对常用 MCU 的类别、特点及各种外设做详细的介绍。

3. 数字信号处理器

嵌入式数字信号处理器 (digital signal processor, DSP) 是专门用于信号处理的处理器，它对系统结构和指令算法进行了特殊设计，结构上一般采用独立的代码、数据、I/O 总线，保证流水线的畅通，具有极强的乘法-累加 (MAC) 计算能力，同时具有很高的编译效率和指令执行速度，常用于音视频编解码、马达控制等。近年来，由于微处理器和微控制器性能的快速提升，以及 FPGA 硬件运算引擎的兴起，DSP 的应用空间受到了一定影响。

DSP 的理论算法在 20 世纪 70 年代就已经出现，但是由于专门的 DSP 处理器还未出现，所以 DSP 理论算法只能通过 MPU 等分立元件实现。但是 MPU 存在一些缺点，例如，较低的处理速度无法满足 DSP 的算法要求；应用领域仅局限于一些尖端的高科技领域等。随

随着大规模集成电路技术的发展，1982 年世界上诞生了首枚 DSP 芯片，其运算速度比 MPU 快几十倍，在语音合成和编码解码器中得到了广泛应用。到 20 世纪 80 年代中期，随着 CMOS（complementary metal oxide semiconductor，互补金属氧化物半导体）技术的进步与发展，第二代基于 CMOS 工艺的 DSP 芯片应运而生，其存储容量和运算速度都得到了很大提高，自此 DSP 芯片成为语音处理、图像硬件处理技术的基础。到 20 世纪 80 年代后期，DSP 的运算速度进一步提高，应用领域也扩大到了通信和计算机方面。20 世纪 90 年代后，DSP 发展到了第五代产品，集成度更高，使用范围也更加广阔。如图 1.9 所示，展示了 DSP SMV320C6727B 器件的功能方框图。

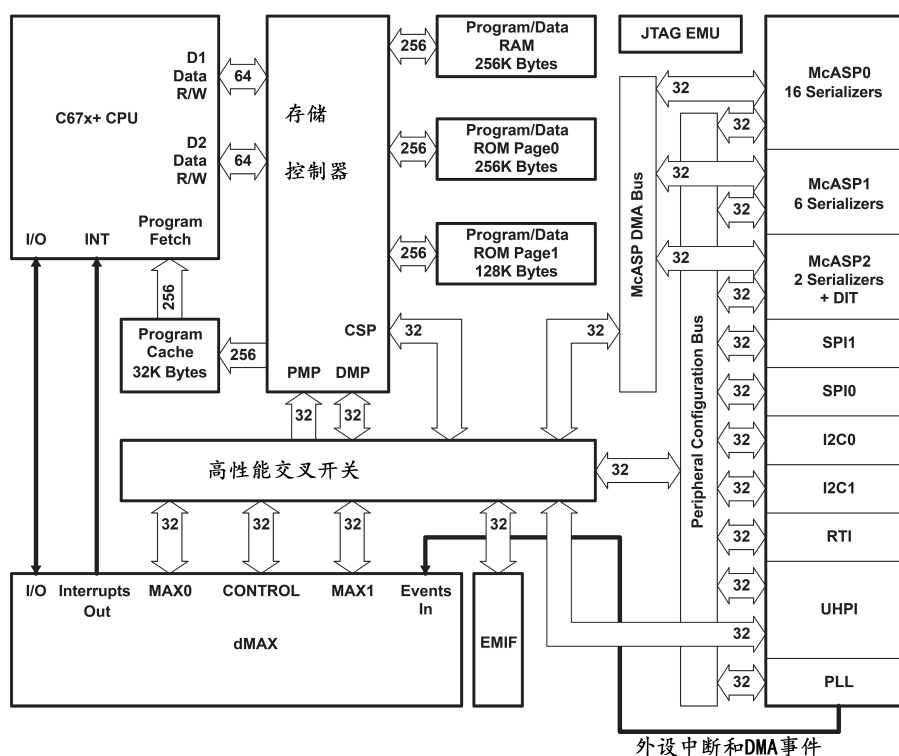


图 1.9 C6727B DSP 功能方框图

SMV320C6727B 是 TI（德州仪器）C67x 系列高性能 32/64 位浮点数字信号处理器的下一代产品。由图 1.9 可知，C6727B 主要包含以下模块。

（1）增强型 C67x + CPU

C67x + CPU 是 C671x DSP 上使用的 C67x CPU 的增强型版本。它与 C67x CPU 兼容，但是极大提升了每个时钟周期内的速度、代码密度和浮点性能。此 CPU 本身支持 32 位定点、32 位单精度浮点和 64 位双精度浮点算术运算。

（2）高效的存储器系统

此存储器控制器将大型片载 256 KB RAM 和 384 KB ROM 映射为统一程序和数据存储器。

（3）高性能纵横开关

一个高性能纵横开关被用作不同总线主控（如 CPU、dMAX、UHPI）与不同目标（外

设和存储器)之间的中央集线器。

(4) 用于实现灵活性和扩展的外部存储器接口 (EMIF)

C6727B 上的外部存储器接口支持一个单组 SDRAM 和单组异步存储器。EMIF 数据宽度为 16 位宽。SDRAM 支持的存储器包括 x16 和 x32 SDRAM 器件, 这些器件具有 1、2 或 4 组。C6727B 将 SDRAM 支持扩展至 256Mbit 和 512Mbit 器件。

(5) 针对高速并行 I/O 的通用主机端口接口 (UHPI)

UHPI 是一个并行接口, 通过这个接口, 一个外部主机 CPU 能够访问 DSP 上的存储器。

(6) 多通道音频串口 (McASP0、McASP1 和 McASP2)

多通道音频串口 (McASP) 可以与编解码器 (CODEC)、数模转换器 (DAC)、模数转换器 (ADC) 和其他器件无缝对接。

(7) 集成电路间串行端口 (I2C0、I2C1)

C6727B 包含两个集成电路间 (I2C) 串行端口。一个典型应用如下: 将一个 I2C 串行端口配置为一个受外部用户接口微控制器控制的端口; 另外一个 I2C 端口可被 C6727B DSP 用来控制外部外设器件。这两个 I2C 串行端口与 SPI0 串行端口引脚复用。

(8) 串行外设接口端口 (SPI0、SPI1)

与 I2C 串行端口的情况一样, C6727B DSP 也包含两个串行外设接口 (SPI) 串行端口。这使得一个 SPI 端口可被配置为一个受控端口来控制 DSP, 而另外一个 SPI 串行端口被 DSP 用来控制外设。

(9) 实时中断定时器 (RTI)

实时中断定时器模块包括两个 32 位计数器和预分频器对、两个输入捕捉、具有自动升级功能的 4 个比较、针对增强型系统稳健耐用性的数字安全看门狗 (可选)。

(10) 时钟生成 (PLL 和 OSC)

DSP 包含一个灵活的、软件设定的锁相环 (PLL) 时钟发生器。通过分割 PLL 输出, 可生成 3 个不同的时钟域 (SYSCLK1、SYSCLK2 和 SYSCLK3)。SYSCLK1 是 CPU、存储控制器和存储器使用的时钟。SYSCLK2 是外设子系统和 dMAX 使用的时钟。SYSCLK3 只由 EMIF 使用。

4. 片上系统

片上系统 (system on chip, SoC) 是追求产品系统最大包容的集成器件, 目前在嵌入式系统领域很受欢迎。SoC 最大的特点是成功实现了软硬件无缝结合, 它可以直接在处理器芯片内嵌入操作系统的代码模块。将整个嵌入式系统集成到一块芯片中去, 这一做法使得系统电路板变得很简洁, 对于减小系统的体积和功耗、提高系统的可靠性和设计生产效率非常有利。

SoC 是把微处理器和某些特定应用外设结合在一起、为某些应用而定制的专用芯片。如 CC3220 嵌入式 Wi-Fi 芯片就是一个专用于物联网应用的嵌入式 Wi-Fi SoC。SoC 有如下两个显著的特点。一是硬件规模大, 通常基于 IP 核 (intellectual property core, 知识产权核) 的设计模式。所谓的 IP 核是指某一方提供的、形式为逻辑单元或芯片的可重用模块。IP 核通常已经通过了设计验证, 设计人员以 IP 核为基础进行设计, 可以缩短设计所需的周期。

IP 核可以通过协议由一方提供给另一方，或由一方独自占有。IP 核的概念源于产品设计的专利证书和源代码的版权等。设计人员能够以 IP 核为基础进行专用集成电路或现场可编程逻辑门阵列的逻辑设计，以减少设计周期。二是软件比重大，需要进行软硬件协同设计。目前在对性能和功耗要求极高的终端芯片领域，SoC 已占据主导地位，SoC 芯片也在多媒体、网络及系统逻辑等应用领域中发挥重要作用。

图 1.10 为 Zynq-7000 SoC 的结构框图，它是业界首款 All Programmable SoC，也是同类产品市场的先锋。凭借高性能、低价格等优势，该产品成为如小型蜂窝基站、多摄像头驾驶员辅助系统、工业自动化机器视觉、医疗内窥镜和 4K^①超高清电视应用领域的最佳选择。如图 1.10 所示，该芯片主要包含了处理系统（processing system）、可编程逻辑、可配置加密引擎等。处理系统又包含了应用程序处理器单元（APU）、存储器接口、I/O 外设、互连等 4 个部分。

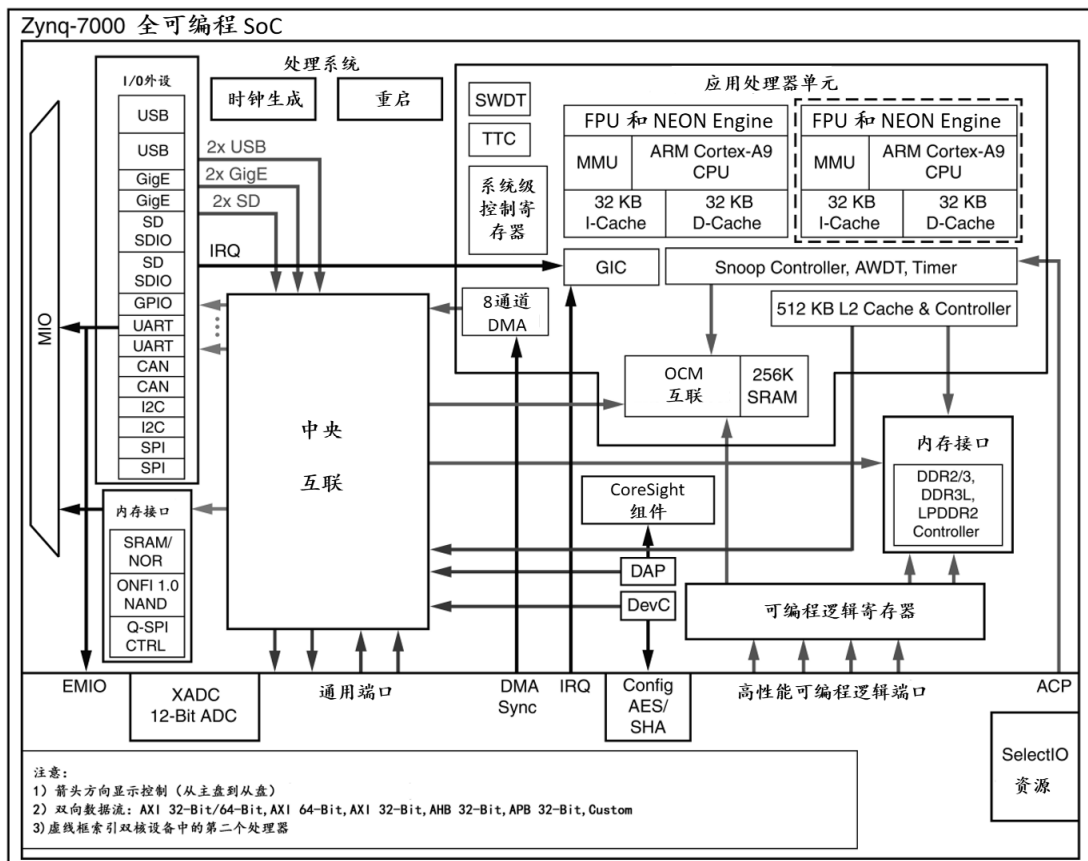


图 1.10 Zynq-7000 SoC 功能框图

应用处理器单元（application processor unit, APU）位于处理系统里的中心位置。应用处理器单元具有双 ARM Cortex-A9 核，还具有高速缓冲、DMA、定时器、中断控制、浮点和

^① 4K 指物理分辨率为 3840×2160。

NEON 协处理、硬件加速器等功能和特点，是处理器的核心部分。SCU (snoop control unit, 窥探控制单元) 用来保持双核之间的数据 Cache 的一致性，即第一个 A9 处理器写存储时，只是写在了缓存里，没有进主存，如果第二个 A9 读操作涉及第一个写脏了的数据段，SCU 要保证第二个 A9 的缓存里是最新的数据。如果第二个 A9 写同样数据段的数据，需要在第一个 A9 中体现出写的内容。SCU 的存在使得两个核组成互相联系的“双核”。

1.2.3 典型 MCU 介绍

微控制器 (MCU) 诞生于 20 世纪 70 年代末，早期被称为单片微型计算机，简称单片机。它作为微型计算机的一个重要分支，应用广泛且发展迅速。从微控制器诞生至今，已发展成为上百种系列的近千个机种。随着技术的不断进步和发展，根据总线或数据寄存器的宽度，MCU 历经 4 位、8 位、16 位、32 位等阶段，性能和片上外设资源也越来越强大和丰富，在控制方面优势十分明显。常见的 MCU 系列包括 MCS-51 系列 (8 位)、PIC 系列 (8~32 位)、MSP430 系列 (16 位)、ARM Cortex-M 系列 (32 位)、RISC-V 系列 (32 位) 等。4 位 MCU 大部分应用在计算器、遥控器、呼叫器、儿童玩具等电子器件中，现在已比较少用；8 位 MCU 大部分应用在电表、简单控制器、键盘等电子器件中；16 位 MCU 大部分应用在仪器仪表、马达控制、数码相机等领域；32 位 MCU 大部分应用在可编程控制器 (PLC)、电机控制、数据采集系统、打印机等方面。目前，MCU 正朝着多品种方向发展，并且将进一步向着低功耗、小体积、大容量、高性能、低价格等方向发展。

32 位处理器除了常用的 ARM 处理器，还有 RISC-V 处理器。RISC 的英文全称是 reduced instruction set computer，即精简指令集计算机。RISC-V 是基于 RISC 原理建立的免费开放指令集架构 (ISA)，V 是罗马数字，代表第五代 RISC (精简指令集计算机)。RISC-V 完全开源，RISC-V 基金会不会收取高额的授权费；架构简单，RISC-V 基础指令集只有 40 多条，加上其他的模块化扩展指令总共几十条指令；易于移植 *nix，RISC-V 提供了特权级指令和用户级指令，同时提供了详细的 RISC-V 特权级指令规范和 RISC 用户级指令规范的详细信息，使开发者能非常方便地移植 Linux 和 UNIX 系统到 RISC-V 平台；模块化设计，RISC-V 架构不仅短小精悍，而且其不同的部分还能以模块化的方式组织在一起，从而试图通过一套统一的架构满足各种不同的应用场景，并且提供完整的工具链，RISC-V 社区已经提供了比较完整的工具链，并且 RISC-V 基金会会持续维护该工具链。

下面分别介绍一下目前常用的 8 位、16 位和 32 位 MCU。

1. 8 位 MCU——MCS-51 系列

MCS-51 系列单片机是 Intel 于 20 世纪 70 年代的产品，发展至今已有很多变种。目前有很多半导体芯片公司生产基于 MCS-51 内核的单片机，把具有数据处理能力的中央处理器 CPU (MCS-51 内核)、随机存储器 RAM、程序存储器 ROM/Flash、多种 I/O 口和中断系统、定时器/计时器等功能集成到一块硅片上，构成一个小而完善的单片机系统。MCS-51 单片机内部结构框架如图 1.11 所示。

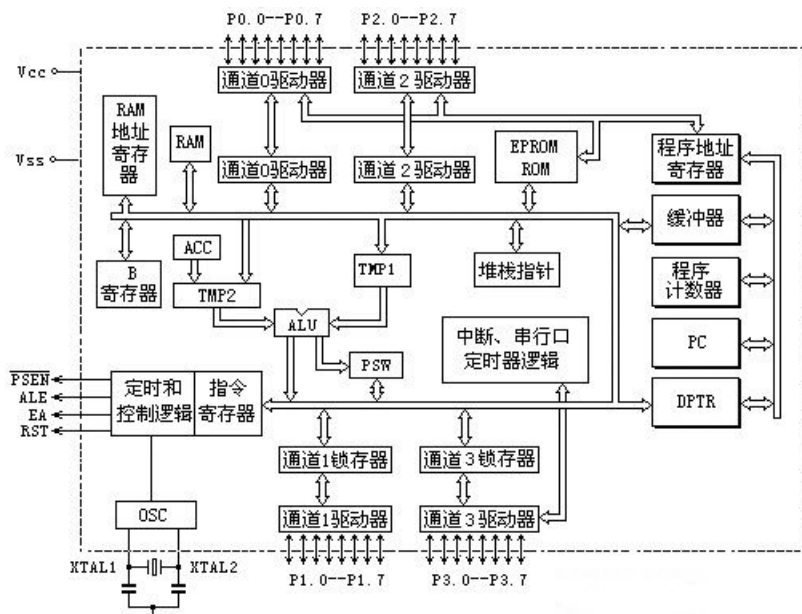


图 1.11 MCS-51 单片机内部结构框图

MCS-51 系列单片机的内部组成及其特点如下。

- ❑ 8 位 CPU。
- ❑ 片内带振荡器，频率范围一般为 1.2~12MHz。
- ❑ 片内通常带 128~256B 的数据存储器。
- ❑ 片内通常带 2~32KB 的程序存储器。
- ❑ 程序存储器的寻址空间为 64KB。
- ❑ 片外数据存储器的寻址空间为 64KB。
- ❑ 128 个用户位寻址空间。
- ❑ 多个特殊功能寄存器。
- ❑ 4 个 8 位并行 I/O 接口。
- ❑ 两个 16 位定时、计数器。
- ❑ 总共有 5 个中断源，具有两个优先级别。
- ❑ 一个全双工的串行接口，可多机通信。
- ❑ 111 条指令，包含乘法指令和除法指令。
- ❑ 片内采用单总线结构。
- ❑ 有较强的位处理能力。
- ❑ 采用单一 +5 V 电源。

2. 16 位 MCU——MSP430

MSP430 系列单片机是美国德州仪器 (TI) 于 1996 年开始推向市场的一种 16 位超低功耗、具有精简指令集 (RISC) 的混合信号处理器 (mixed signal processor)。该系列单片机多应用于需要电池供电的便携式仪器仪表中。图 1.12 所示是 MSP430F169 结构框图。

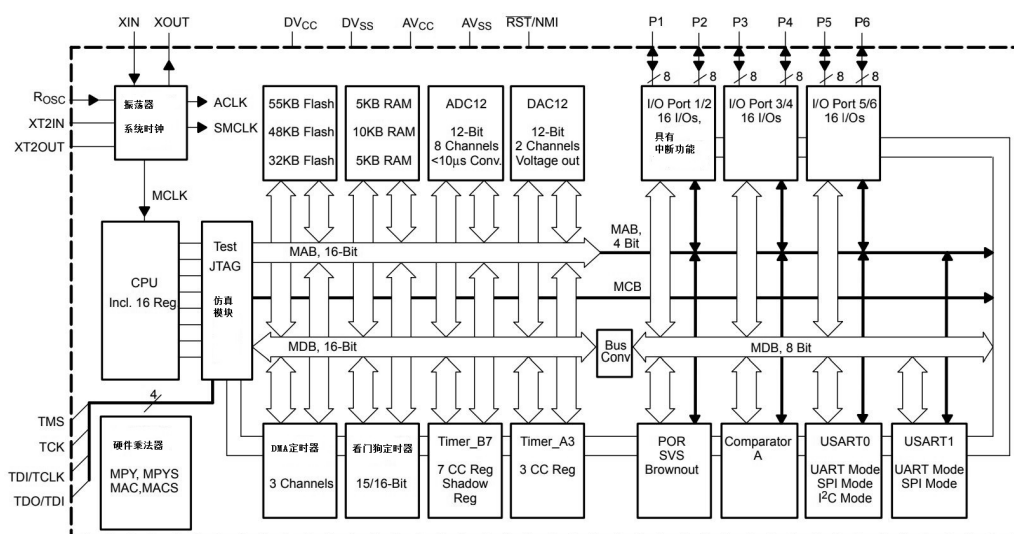


图 1.12 MSP430F169 结构框图

MSP430F169 单片机的功能部件和特点如下。

- ❑ 60KB+256B 可低电压工作的 Flash 模块：可用于存储控制器器件（firmware）程序代码和数据。在 LPM4 待机模式下，存储在 2KB 随机存取内存（RAM）中的数据仍可保持。
- ❑ 多种工作时钟系统：内建一组基本 RC 振荡频率 DCO（digitally-controlled oscillator，数控振荡器），当外部电源供应给芯片的工作电压为 3.6V 时，其最大工作频率为 8MHz，同时该数据会随着芯片工作温度、电压而有所改变。
- ❑ 计时模块提供一组看门狗定时器：系统宕机时可由其将系统重置。它包含一组基本定时器、两组功能完整的 16 位定时器，有上计数、下计数、连续、暂停模式可选。
- ❑ 12 位 8 通道连续逼近式（SAR）模拟数字转换（analog to digital）模块：最大转换速度可达 200kHz，内建参考电压（1.5V、2.5V）、取样保持电路、自动扫描功能。
- ❑ 2 组 12 位数字模拟转换（digital to analog）电压输出模块：两组 D/A 可由程控同步输出。
- ❑ 一组模拟电压比较器：可直接将输出信号提供给计时模块以便于抓取。
- ❑ 7 组 I/O 端口 P0~P6：共 48pin 脚可供输入/输出。
- ❑ 数字传输周边模块提供两组 USART：可作为异步、同步传输、I2C 接口。
- ❑ 硬件乘法器（hardware multiplier）：其是外部周边模块，使用时只需将操作数放到特定缓存器内，即可在下一个机械周期将结果取回，运算过程不需要处理器的参与。可做无符号相乘、有符号相乘、无符号相乘累加、有符号相乘累加 4 种模式的运算。
- ❑ 3 个内部 DMA（direct memory access，直接存储器访问）控制器：不需要处理器的介入就可将数据在两个内存地址使用。

3. 32 位 MCU——MSP432

MSP432 系列属于低功耗、高性能的微控制器，它是 TI 的 MSP 低功耗微控制器系列中的 32 位 ARM Cortex-M4 内核产品。图 1.13 是 MSP432P401R 功能模块图。

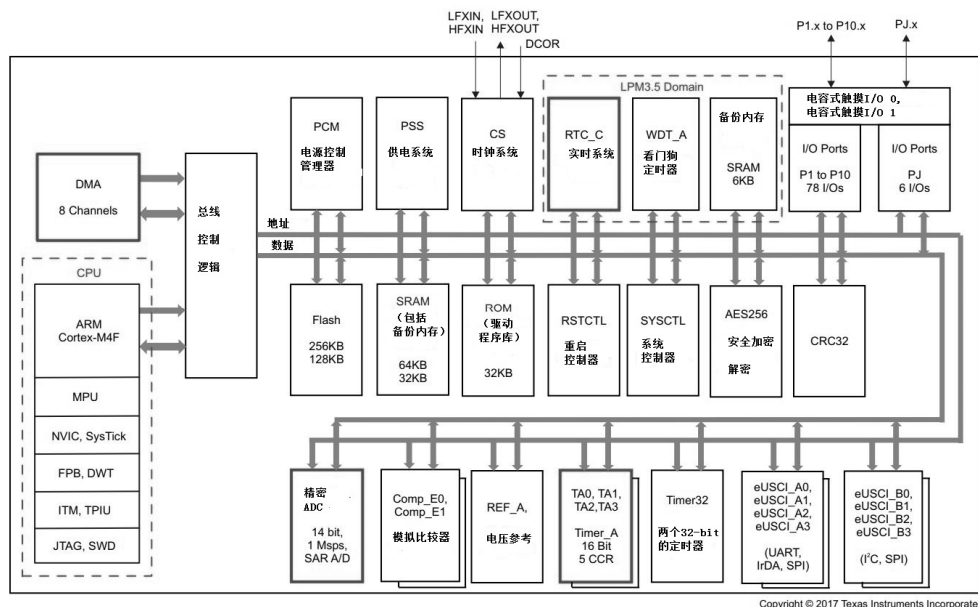


图 1.13 MSP432P401R 功能模块图

MSP432P401R 微控制器可借助 FPU 和 DSP 扩展提供超低功耗性能。其主要特点如下。

- ❑ 带浮点单元和 DSP 加速功能的 48MHz 32 位 ARM Cortex-M4F 内核。
- ❑ 功耗：95 μ A/MHz 工作功耗和 850nA RTC 待机操作功耗。
- ❑ 模拟：24 通道 14 位差动 1MSPS SAR ADC，两个比较器。
- ❑ 数字：高级加密标准 (AES256) 加速器、CRC、DMA、32 位硬件乘法器。
- ❑ 存储器：256KB 闪存、64KB RAM。
- ❑ 计时器：4 个 16 位、2 个 32 位。
- ❑ 通信：多达 4 个 I2C、8 个 SPI、4 个 UART。

4. 32 位 MCU——CH2201

CH2201 物联网安全 MCU 是平头哥基于无剑 MCU 平台的生态芯片，其内置平头哥 32 位可信 CPU 玄铁 E802 (TEE 安全)，支持 AliOS Things 操作系统、物联网接入套件、可信引导、安全接入、可靠存储、差分升级等功能。通过 CH2201，用户可体验平头哥软硬一体芯片生态的开发优势。图 1.14 是 CH2201 的系统结构框图。

CH2201 的主要特点如下。

- ❑ 丰富的入云通道支持：开源 MQTT、CoAP、LwIP 等常用网络组件，支持阿里云、天翼云、OneNet 等接入。
- ❑ 安全可靠：在 Flash 中存储固件加扰，可信引导，启动进行镜像验签。

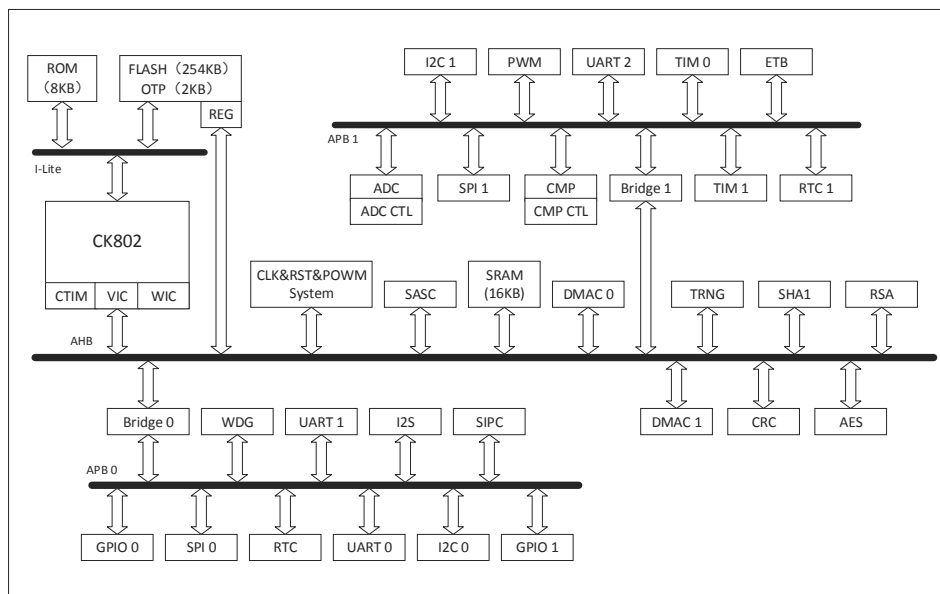


图 1.14 CH2201 系统结构框图

- ❑ 模拟：16 通道 12 位可单独配置的 ADC，支持差分输入，拥有高达 1000MSPS^① 的转换速率。
- ❑ 存储器：256KB 闪存、80KB RAM。
- ❑ 计时器：2 个独立可编程 32 位定时器。
- ❑ 通信：2 个 I2C、2 个 SPI、3 个 UART。

5. 32 位 MCU——STM32

STM32 系列从内核上可分为 ARM Cortex®-M0、M0+、M3、M4 和 M7，是意法半导体（ST）专为要求高性能、低成本、低功耗的嵌入式应用而设计，是目前非常流行的 32 位 MCU，也有一些国产的 STM32 兼容产品。按内核架构和应用适应性可以分为主流产品（如 STM32F0、STM32F1、STM32F3）、超低功耗产品（如 STM32L0、STM32L1、STM32L4、STM32L4+）和高性能产品（如 STM32F2、STM32F4、STM32F7、STM32H7）。图 1.15 是 STM32F303 的系统结构框图。

STM32 系列 MCU 型号很多，主频、存储器容量、外设、封装差异较大，其主要特征如下。

- ❑ 基于 ARM Cortex-M 内核的 32 位 MCU。
- ❑ 超前的体系结构：高性能、低电压、低功耗、创新的内核以及丰富的外设。
- ❑ 芯片型号众多、应用覆盖面广。
- ❑ 简单易用、低风险。
- ❑ 提供强大的软件支持、全面丰富的技术文档和软件包。

^① SPS 全称 samples per second，表示采样一次每秒，是转化速率的单位。MSPS 表示每秒采样百万次。

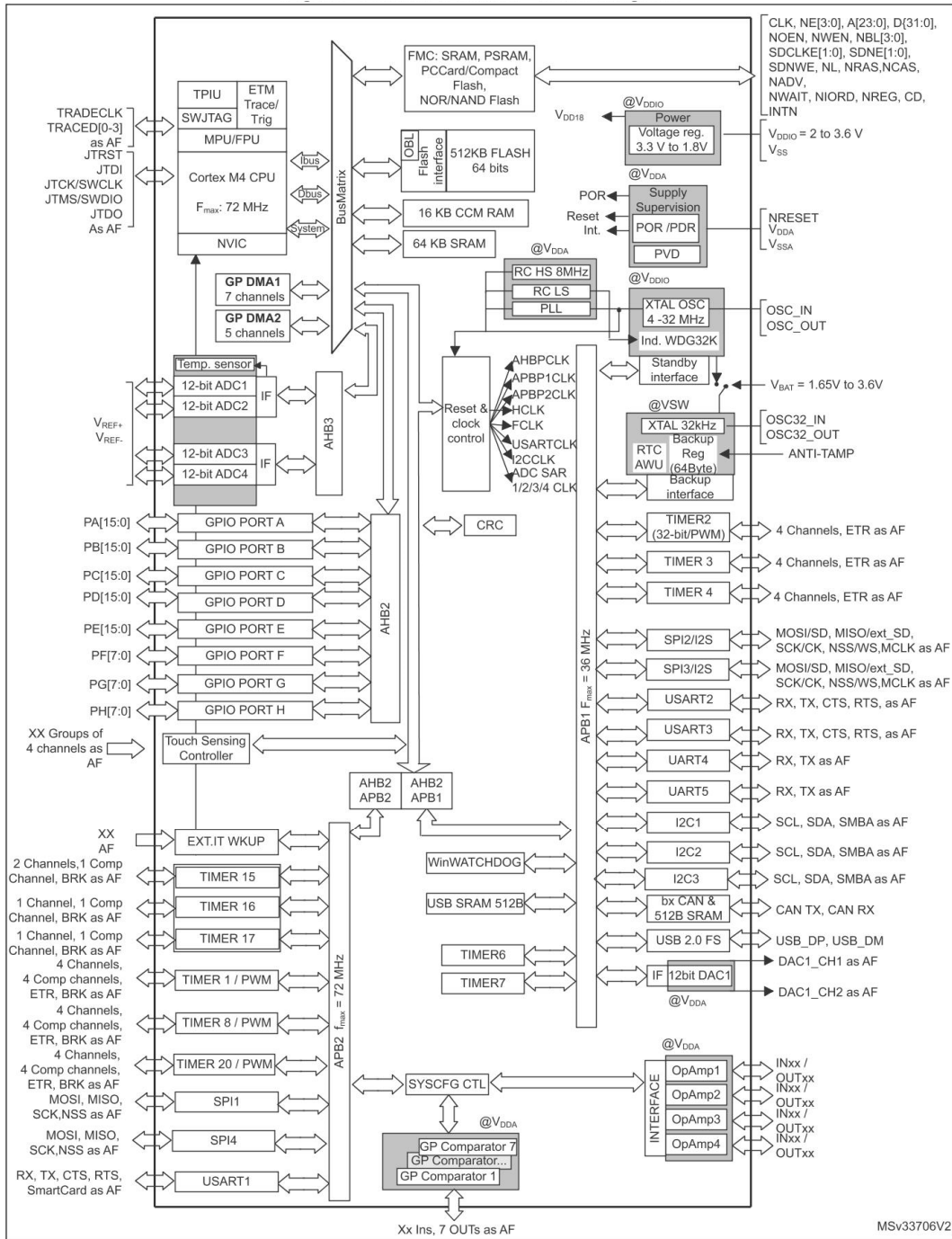


图 1.15 STM32F303 系统结构框图

6. 32 位 MCU——PIC32

PIC32 是美国微芯科技公司 (Microchip) 推出的 32 位 MCU 系列, 工作频率为 80MHz,

在零等待状态闪存访问时的性能为 1.56DMIPS/MHz^①，MCU 的工作电压为 2.3~3.6V，具有丰富的外设，有两个编程和调试接口，多个 16 路 10 位 ADC 和两个模拟比较器。图 1.16 所示为 PIC32 的系统结构框图。

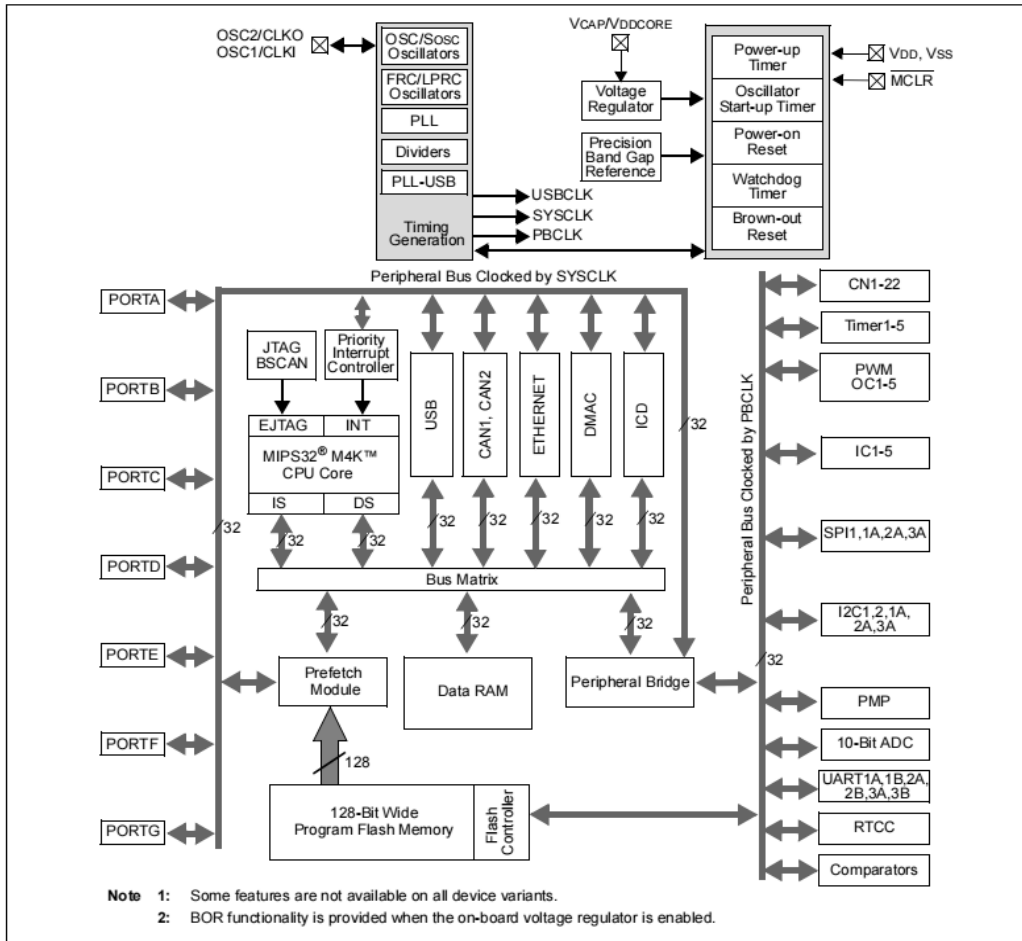


图 1.16 PIC32 系统结构框图

PIC32 的主要特点如下。

- ❑ 具有 5 级流水线的 MIPS32® M4K® 32 位内核。
- ❑ 最高频率为 80MHz。
- ❑ 预取缓存模块，以加快闪存读取。
- ❑ 单周期乘法和高性能除法单元。
- ❑ 具有单独可编程优先级的多个中断向量。
- ❑ 多达 6 个 UART 模块、4 个 SPI 模块、5 个 I2C 模块、5 个 16 位的定时器。

^① DMIPS 是一个测量 CPU 运行一个整数运算测试程序时表现出的相对性能高低的单位。D 全称 dhrystone，是一种整数运算测试程序；MIPS 指每秒执行百万条指令。

7. 32 位 MCU——CH2601

CH2601 是基于平头哥 32 位玄铁 CPU E906 的 RISC-V 生态芯片,它配置 512KB Flash、256KB SRAM 及丰富的片上外设,最高主频为 220MHz,支持 AliOS Things 物联网操作系统、平头哥 YoC 软件平台及平头哥剑池开发工具(CDK)。其主要特点如下。

- 支持 Wi-Fi 通信、音频播放等应用实例,支持多媒体组件、含 Codec 解码器。
- 内置丰富的基础组件,包括 KV、AT、CLI、LittleVGL 等。
- 内置 AliOS Things 物联网操作系统。
- 适配 CSI 驱动接口,开放便捷。
- 32 位 RISC-V: RV32IMACX 指令集、5 级流水线、3.1CoreMark/MHz、220MHz 主频。
- 12 位 ADC,支持单次采样、连续采样及外部信号触发采样模式。
- 存储器: 512KB/4MB Flash、256KB SRAM。
- 通信: 2 个 SPI、1 个 I2C、2 个 UART。
- 复位电路: 外部复位电路、看门狗复位、上电复位、掉电复位、CPU 软复位。

虽然 MCU 品种繁多、应用也不尽相同,但其主要结构和功能特点是相似的。现代 MCU 的主要特点如下。

- 集成度高: MCU 集成了一个微型计算机系统所需的资源,包括处理器、存储器、常用外设、电源管理、复位等,堪称“麻雀虽小,五脏俱全”。
- 性能较高: 目前很多 MCU 内核(如 ARM Cortex-M4)都是 32 位字长,且带浮点处理单元(FPU),工作主频超过 200MHz,相当于当时 Intel 80486 的计算能力。MCU 不仅适用于简单的逻辑控制,而且适用数字滤波、电机控制、音频处理等运算密集的应用。
- 外设丰富: MCU 面向数据采集、控制类应用,一般都集成了丰富的外设,包括定时器、UART、I2C、USB 等。MCU 功能强大、使用方便,通常单个芯片即可满足应用需求。
- 低功耗: 这是 MCU 相对于其他微处理器最具优势的特点。很多 MCU 都可以做到 50~100 μ A/MHz 的运行功耗,在休眠模式耗电可低至微安级。MCU 具有多种电源管理和低功耗模式,通过与软件结合,可以获得最佳的功耗性能,大大增加了电池的续航能力。
- 产品线广: 每种 MCU 都有一个系列产品,存储器资源、外设种类、引脚数、封装形式等都有不同的选择,用户可根据自己的应用需求,选择最合适的 MCU 芯片,设计出性价比最好的产品。

1.3 本章小结

嵌入式系统是服务于特定应用的计算机系统,其硬件和软件具有专用性。针对不同的

应用，嵌入式系统中硬件和软件的复杂程度有很大差异，对实时性、可靠性、体积、功耗、成本等有不同的要求。嵌入式系统一般需要专用的软硬件开发调试环境和工具。嵌入式处理器一般可分为嵌入式微处理器（MPU）、嵌入式微控制器（MCU）、数字信号处理器（DSP）、专用集成电路（ASIC）等，它们的性能、应用侧重点各不相同。目前广泛使用的嵌入式处理器是各种嵌入式微控制器 MCU。常用的嵌入式微控制器有多种内核，包括主流的 32 位 ARM 处理器及 RISC-V 处理器。物联网时代给嵌入式系统带来了新的巨大发展空间。

1.4 习 题

1. 什么是嵌入式系统？
2. 简述嵌入式系统的发展过程。
3. 从硬件系统来看，嵌入式系统由哪些部分组成？
4. 从软件系统来看，嵌入式系统由哪些部分组成？
5. 嵌入式系统有哪些特点？
6. 在嵌入式系统的开发过程中，调试嵌入式系统的方法和手段主要有哪几种？
7. 举出几个嵌入式系统应用的例子，通过查资料和独立思考，说明这些嵌入式系统产品主要由哪几部分组成，每个组成部分完成什么功能。
提示：智能家电、仪器仪表、工业控制类产品的例子等。
8. 通过查阅资料，列举几个嵌入式 MPU、MCU 的制造商及其具体产品型号，并说明其主要特点。
9. 通过查阅资料，试分析物联网应用对嵌入式处理器功能、性能的需求。
10. 通过查阅资料，简述嵌入式微控制器（MCU）的发展史。