

# 面向物联网领域的嵌入式系统研究

ZYZ

## 嵌入式系统

嵌入式系统被定义为：以应用为中心、以计算机技术为基础、软件硬件可裁剪、适应应用系统对功能、可靠性、成本、体积、功耗严格要求的专用计算机系统。嵌入式系统是将先进的计算机技术、半导体技术和电子技术和各个行业的具体应用相结合后的产物,这一点就决定了它必然是一个技术密集、资金密集、高度分散、不断创新的知识集成系统。

嵌入式系统的最大特点就是技术密集化,能够将一些先进的技术完美的呈现出来,它同普通的计算机技术相比较有着自己的优势,首先,在硬件方面,嵌入式系统的可封装、实用性、专用性更高;其次,在软件方面,嵌入式系统相比较于其它的计算机技术,其传输速度更快、所需代码更小,大大的节省了时间、空间,更有利于信息、数据的传输。随着信息时代的不断发展,人们对于信息的要求也越来越高,而嵌入式系统正是在这个阶段发展而来,它的诞生在一定程度上满足了人们对于信息技术的一些高端要求,它也不断地被应用到各种各样的领域中去,推动信息技术的不断发展,使得社会不断进步。

## 物联网

物联网是伴随着计算机技术和通信网络的迅速发展而出现的一种新技术,它的主要功能是利用智能识别设备通过感知、计算和分析来获取外部信息,继而通过互联网技术实现信息的交换和共享,从而将提供信息的对象互联。

在互联网发展的基础上,为实现物品的智能化,建立物与物之间的连接,物联网应运而生。物联网是可以通过射频识别、红外感应器、全球定位系统、激光扫描器等信息传感设备,按约定的协议,把任何物品与互联网相连接,进行信息交换和通信,以实现物品的智能化识别、定位、跟踪、监控和管理的一种网络。

物联网在各个领域的蓬勃发展，极大地推动了智能交通、智能家居、智慧医疗以及智慧物流等行业的快速发展，成为社会发展的重要动力，其广阔的应用领域可以让人们享受到更加安全轻松便捷的生活。根据物联网的特点，其层次可划分为感知层、网络层和应用层。感知层用于感知和采集物体的信息数据，并将物理信息传送给网络层；网络层将物体连接到网络中，实现信息交互；应用层则对感知信息和数据进行相关分析和处理，以实现智能化决策和控制。

## 物联网与嵌入式系统的联系

物联网的主要功能是信息的传送和共享，而嵌入式系统主要是一种微型计算机系统，它是根据嵌入对象所需要的功能而设计的。二者在功能上的主要区别是嵌入式系统具有针对性，它可以根据不同的设备而改变，应用于不同的地方就会具有不同的功能；而物联网主要是通过前端的感知设备获得信息，进而通过网络的传输实现信息的共享。嵌入式系统通过一定的设计就可以实现物联网的功能，因此，物联网是包含在嵌入式系统之中的，随着目前嵌入式系统技术的不断发展，二者之间出现了互相融合的趋势，即在嵌入式系统中应用物联网技术。

物联网就是将物和物之间用网络连接起来的互联网，物联网中的“物”通常是指物理设备，包括汽车、家用电器和其他的电子设备以及其中的传感器。在这些物联网终端的电子器件中，大多数都应用到了嵌入式系统，比如，汽车中使用的内嵌 GPS 模块以实现车辆的定位和行驶过程中的导航；在家用电器，如空调、冰箱等也通过嵌入式系统实现了网络化。因此，在物联网之中相互连接，用来感知信息的物体大多数都应用到了嵌入式系统设备以实现各种功能。随着互联网技术和传感器技术的迅速发展，嵌入式系统已经越来越广泛地应用到物联网中，从而提高物联网前端设备对信息的获取和感知能力。

互联网的发展和普及为实现物联网的应用提供了前提条件，而近年来传感器技术和射频识别技术的广泛应用使得嵌入式系统得以在物联网领域大展身手。物联网和嵌入式系统的融合将是必然趋势，它必将为人们的生活带来极大的便利。

# 面向物联网领域主流的 SoC

## 蓝牙芯片

随着无线通信技术的飞速发展，物联网技术也越来越成熟，人们对物联网应用的关注越来越多。同样随着集成电路技术的不断发展，系统芯片已经成为芯片的主流。此外，芯片的功耗很大程度上会影响芯片的性能和应用范围，尤其是面向物联网应用的芯片，其影响更大，所以芯片的低功耗设计也成为 IC 设计者越来越不可忽视的因素。由于蓝牙是一种重要的短距离无线通信的技术，尤其是低功耗蓝牙，在物联网应用中具有非常重要的地位。

同时随着集成电路技术的不断发展，系统级芯片设计已经成为当今集成电路设计的主流方向，而且基于平台的设计方法是目前最主流的设计方法。基于平台的设计方法，是指在已有 SoC 平台上进行新的 SoC 的设计，是一种系统级的复用技术，除了对 IP 核进行复用，还能对整个系统架构进行复用，从而实现设计的正确性和高效性。

目前，美国的博通公司、高通公司、德州仪器公司和英国的 CSR 公司走在蓝牙芯片技术研发的前列，而且市面上已经有许多国外公司的蓝牙 4.X 标准的蓝牙芯片，比如德州仪器公司开发的 CC2541、CC2640 等面向低功耗蓝牙应用的无线 MCU；ST 公司的低功耗蓝牙 SoC 芯片 BlueNRG；CSR 公司的 CSR1000 系列的低功耗蓝牙芯片等。同样，国内也有一些芯片设计公司也拥有自己的蓝牙芯片，比如作为国内芯片行业的佼佼者，华为海思于 2014 年推出的四合一芯片 Hi1101，该芯片支持 WLAN、蓝牙 4.0、FM 收发和卫星定位技术；展讯 2015 年推出的可用于手机、平板电脑的综合型 SoC 芯片 SC9830A，同样支持蓝牙 4.0 协议。而对于 2016 年发布的蓝牙 5.0 技术标准，市面上的蓝牙 5.0 芯片还相对较少，但仍有一些公司已经推出。比如联发科技 2017 年推出的支持蓝牙 5.0 规格的 SoCMT7622 等。

## 窄带物联网（NB-IoT）芯片

由于物联网领域极为庞杂，因此不可能用一款芯片覆盖正在由不断扩大的应用和市场组成的物联网。物联网芯片不仅包括集成在传感器/模组中的基带芯片、射频芯片、定位芯片等，也包括嵌入在终端中的系统级芯片——嵌入式微处理器（MCU/SoC 片上系统等），且这些领域的物联网芯片需求规模巨大。因此，对于芯片产业而言，物联网给行业带来了新的机遇。

聚焦物联网产业中最热门的窄带物联网（NB-IoT）技术来看，为赢得与国际芯片企业一争高下的机会，中国大陆和台湾地区前几大芯片设计企业（华为海思、紫光展锐、联发科、中兴微电子）早已投入研发中，并已面向市场推出了具有竞争力的 NB-IoT 芯片产品。通过技术实力、频谱范围、芯片设计、运营商支持、商用程度等方面综合考量，紫光展锐、华为海思和联发科等企业已位居领先地位。

### 春藤 8908

春藤 8908 是紫光展锐的代表芯片。紫光展锐在物联网领域拥有目前最全面的产品线，覆盖无线广域网以及短距离无线网络。以蜂窝通信为例，包含 2G/3G/4G/NB-IOT/eMTC 等成熟技术与产品，在短距离无线连接方面，还有 BT5.0、BT MCU、WiFi MCU 等芯片。在 NB-IoT 技术方面，紫光展锐的 NB-IoT 芯片可完整支持 3GPP Release 13 的几乎所有功能，覆盖性强，功耗低，不仅可满足中国市场需求，同时也瞄准欧美及海外市场。

紫光展锐在 2017 第四季度同时推出的 NB-IoT 单模的春藤 8908、NB-IoT/GSM 双模的春藤 8909 两颗芯片，并在 2018 上半年成功流片 NB-IoT/eMTC/GSM 三模的春藤 8915。其中春藤 8908 是业内唯一一款 NB+GSM 双模、集成度最高的物联网单芯片。硬件方面，春藤系列芯片具备高集成度、高可靠性、高性价比等优点，适配全球运营商网络频段。同时配套紫光展锐的射频外

围芯片，降低方案设计难度，可提供完整的芯片打包方案。在软件方面，展锐可提供全套的软件开发包（SDK），帮助客户基于 NB-IoT 通信芯片进行二次开发，以应对物联网应用碎片化的特点。

在 2018 年中国联通的 NB-IoT 通信模块招标项目中，拿到最大中标份额的深圳市有方科技和大唐通信 NB-IoT 模组均采用的紫光展锐春藤 8908 解决方案，这也意味着紫光展锐芯片在中国联通物联网 NB 通信模块项目中占据 55% 的份额。

## Boudica 系列

Boudica 系列是华为海思的代表芯片。海思在手机芯片设计领域的实力是有目共睹的，而在 NB-IoT 芯片领域，海思也早早地投入研发中。2017 年 9 月，华为海思推出了全球首款正式商用的 NB-IoT 芯片——Boudica120，接着又推出了第二代芯片 Boudica 150。这两款 NB-IoT 芯片的主要特点是高集成、低功耗、可内置轻量级的 Huawei LiteOS 物联网操作系统。

海思的 LiteOS 物联网操作系统是华为面向物联网领域构建的轻量级、开源的“统一物联网操作系统和软件中间件”平台。此外，海思的物联网平台 Ocean Connect 提供 CMP（连接管理）、DMP（设备管理平台）、AEP（应用使能平台）三大能力，通过开放的 APIs 系列化 Agent、丰富开发工具，降低了物联网开发门槛，帮助运营商和企业客户实现多行业终端的快速接入，以及丰富应用的快速集成。海思的 NB-IoT 芯片主要定位于轻量级芯片，目前只支持 NB-IoT 单模，网络协议栈也非常简单，也没有集成 TCP 等上层网络协议，这将给模组厂家集成带来了一定困难。

华为 Boudica 系列产品具有先发优势。在应用方面，华为与深圳、鹰潭、福州水务展开合作。华为则联合电信与深圳燃气和北京燃气开展试点。但是从产品定义的角度，华为芯片只支持单模，未来在更多应用场景支持方面，还存在一些

局限性。

## MT2625

MT2625 是联发科的代表芯片。针对 NB-IoT 市场，联发科的布局也十分积极。2017 年 6 月底，联发科推出了旗下首款 NB-IoT 系统单芯片（SoC）——MT2625，并打造业界尺寸最小（16mm X 18mm）的 NB-IoT 通用模组，以超高集成度为海量物联网设备提供兼具低功耗及成本效益的解决方案。该方案支持 3GPP NB-IoT（R13 NB1, R14 NB2）的 450MHz-2.1GHz 全频段运作，适合全球范围内智能家居、物流跟踪、智能抄表等静态或移动型物联网应用。

MT2625 芯片延续了联发科在整体功耗控制方面的一贯优势，通过在 MCU 和 PSRAM 等组件上加载联发科特有的低功耗技术，使得采用该方案的物联网设备，能够搭配免充电电池达到长时间待机，符合低功耗广域物联网的要求。此前中国电信发布的《NB-IoT 芯片评测报告》指出，联发科 MT2625 芯片续航可达 10 年。

## 面向物联网领域主流的嵌入式 OS

过去十年我国嵌入式系统产业发展迅速，嵌入式应用逐渐告别单片机时代，迈入集成电路、计算机、通信、电子技术等多学科交叉融合的嵌入式系统时代。如今嵌入式系统又将迎来物联网应用的高潮。

物联网系统中有大量的嵌入式设备，与传统的嵌入式设备相比，物联网感知层的设备更小、功耗更低，而且需要安全可靠和具备组网能力，物联网通信层需要支持各种通信协议和协议之间的转换，应用层则需要具备云计算能力。在软件方面，支撑物联网设备的软件比传统的嵌入式设备软件更加复杂，这也对嵌入式操作系统提出了更高的要求。为了应对这种要求，一种面向物联网设备和应用的软件系统——物联网操作系统（也称为面向 IoT 的 OS）应运而生。

互联网为物联网系统搭建了无处不在的互联通道，云计算和大数据的兴起为物联网数据处理和分析提供了技术支撑。在嵌入式设备端，32 位 MCU 技术已经成熟，价格与 8/16 位 MCU 接近，不仅在网关设备上使用，也在传感和执行单元上普遍使用。在 MCU 市场，ARM Cortex M 系列（M0/M3/M4）的 MCU 占有最大的份额。在过去 10 年中，ARM 已经建立了完善的生态环境，这大大帮助了包括物联网 OS 在内的嵌入式软件的发展。

最早具备物联网 OS 特征的是无线传感网（WSN）的 OS，例如加州大学伯克利分校（UC Berkeley）的 TinyOS 和瑞士计算科学学院（Swedish Institute of Computer Science）网络系统小组 Adam Dunkels 开发的 Contiki，它们是无线传感网 OS 的典型代表。2010 年之后欧洲有了 RIOT，相比前面两个 OS，RIOT 更加接近一个完整的 RTOS（实时多任务操作系统），具备实时性和模块化结构，支持标准的 C 和 C++ 编程接口。RIOT 不仅可以运行在小型 MCU 上，还可以支持 MPU。在资源允许的条件下，可以运行最新的互联网和物联网协议栈，并完成协议转换工作。随着 32 位 MCU 的价格下降，Linux 又无法支持没有 MMU 的 MCU，RTOS 理所当然地成为运行在 MCU 的物联网 OS 的首选，原因是基于 RTOS 的设计运行更灵活和可扩展的软件运行在这些系统中。

2014 年 1 月，微软嵌入式事业部总监 Bob Breynaert 透露，微软计划推出物联网版本的 Windows Embedded。2014 年 8 月，微软开始向所有 Windows 物联网开发者配套英特尔 Galileo 主板的 Windows 物联网 OS 预览版。与 Arduino 兼容的 Galileo 主板采用了英特尔 Quark 系统芯片，具有单核 400 MHz 32 位 CPU，尺寸比一张信用卡大不了多少，提供 10/100M 以太网卡、PCI Express、JTAG 和 USB 端口，以及一个 SD 插槽和 RS232 串行端口，英特尔将其定位在物联网和可

穿戴设备平台。Windows 物联网 OS 是一个 Windows8.1 的非商业版本。显然预览版的推出是微软进军物联网计划中的一个步骤，让制造商和开发人员产生新的想法，并提供反馈，以帮助微软继续发展 Windows 物联网 OS。

2014 年 2 月在德国纽伦堡的嵌入式世界大会上，风河发布了其基于 VxWorks 7 的物联网 OS。最近风河在其官方网站上给出了这个产品的介绍和白皮书。微软和风河这两家操作系统大公司的加入，使得物联网 OS 已经呼之欲出。与此同时，传统的 RTOS 的公司也纷纷有所动作，比如因开发了  $\mu\text{C}/\text{OSII}$  和  $\mu\text{C}/\text{OSIII}$  而闻名的 Micrium 公司 2014 年 9 月推出了 Micrium Spectrum，这是一个针对物联网设备，集成了嵌入式软件、协议栈和云服务的端到端的解决方案。除此之外，Express Logic 公司推出基于 ARM 物联网设备的 XWave 平台，该公司的 RTOS 内核是 ThreadX。

国内关于物联网 OS 的研究和开发才刚刚开始，多数还停留在学习、移植和应用阶段，自主研发的物联网 OS 很少，高校物联网专业教学上有的使用开源的 TinyOS。中国科学院软件研究所与无锡中科物联网基础软件研发中心发布过一套物联网软件平台。北京凯思昊鹏的 SENHopen OS，可以运行于无线传感网微节点之上，是国内比较早的商用物联网（无线传感网）OS。中兴公司自主研发的嵌入式操作系统在物联网设备上也有应用。2014 年 7 月，上海庆科联合阿里智能云发布了物联网 OS——MICO，MICO 目前是基于庆科的 WiFi 模块，未来庆科希望开发与硬件无关的物联网 OS。

2014 年 10 月，ARM 公布了专门为物联网设计的软件及系统平台，以加速物联网设备的开发及应用。该软件专门为基于 ARM CortexM 架构的 MCU 而设计，包括了设备端的嵌入式 mbed OS 操作系统、软件工具包 mbed 和云端的 mbed Device Server 三大部分。ARM 公司称“能够以安全的方式为连接和管理设



备提供所需的服务器端技术”，ARM 在借 mbed 基础软件为物联网设备构建“基石”，希望物联网设备商能够专注于为其产品增加更多新功能，使产品尽快上市并具有竞争力。

mbed OS 的设计理念有别于传统的 RTOS 和 Linux，它不追求最大的灵活性，而强调物联网应用专业性——低功耗和高效率。在 mbed OS 的设备端，连接和安全是两大亮点。mbed OS 支持多种互联标准，包括 3G/4G/LTE、Bluetooth Smart（蓝牙 4.0）、WiFi 以及 6LoWPAN（基于 IEEE 802.15.4 实现 IPv6 通信协议）。在安全方面，mbed OS 提供了通信和设备安全两种机制（在一个称为 CryptoBox 的子系统里），提供安全的 API，支持安全标示、固件升级、认证和安全存储等功能，这个功能大大简化了产品的安全设计。mbed OS 强调代码的可重用性，面向对象和模块化，开发者可用 C++ 语言编写自己的应用。

mbed Device Server 是 mbed OS 软件平台的另一大亮点，这是由 ARM 自己开发的云端软件，支持 IPSO Web 目标管理和 OMA 设备管理标准，支持 CoAP、6LoWPAN 和 HTTP 协议，提供设备开发者可以免费使用的 SDK 开发包。在应用上，mbed OS 目前侧重在智能家居、智慧城市（智能照明）和可穿戴设备 3 个方向。

据 ARM 介绍，mbed OS 将对开发者和设备制造商免费提供，该操作系统源码一部分基于 ARM 收购的 Sensinode 公司的技术构建，另一部分由内部开发。mbed OS 基于 Apache 2.0 许可证，操作系统大部分为开源，但部分组件只对 ARM 合作商开放。ARM 在 2014 年向合作商提供 alpha 版本的 mbed OS，预计 2015 年 10 月发布第一个正式的版本。目前业界对 mbed OS 持观望态度，毕竟一个操作系统需要一段很长的时间才能让用户和市场认可，ARM 虽不能说是从零开始，但在这个领域还是一个新兵。因为 ARM 在嵌入式系统技术和产业链方

面有 20 多年的成功经验，所以我们有理由期待 mbed OS 的成功，另外开源技术也是 ARM 切入物联网市场的一大利器。

物联网产业正处在发展初期，碎片化特点必将导致物联网 OS 的多样性。一种物联网 OS 很难支持物联网系统中的所有设备。从技术层面看，传统嵌入式 Linux 和 RTOS 将持续在物联网设备中应用，但它们都将面临来自技术和商业层面的挑战。

## 面向物联网领域主流的系统架构

物联网系统是由系统内核，外围功能组件，物联网协同框架，通用智能引擎，集成开发环境等几个大的子系统组成。这些子系统之间相互配合，共同组成一个完整的面向各种各样物联网应用场景的软件基础平台。这些子系统之间有一定的层次依赖关系，比如外围功能组件需要依赖于物联网系统内核，物联网协同框架需要依赖于外围功能组件，而公共智能引擎，需要依赖于下层的内核，外围功能组件，甚至是物联网协同框架等。目前主流的物联网操作系统，比如 Google 的 Brillo，Linux 开放基金会的 Ostro 项目，以及 HelloX 项目，都遵循这样一种框架。

## 物联网操作系统内核

内核是任何操作系统都有的核心组件，操作系统的核心功能和核心机制，都是在内核中实现的。比如最核心的线程/任务管理，内存管理，内核安全和同步等机制。虽然从功能上说，大部分操作系统的内核都相差不大，但是在这些具体功能的实现上，面向不同领域的操作系统，其实现目标和实现技术都是不同的。

对于面向嵌入式领域的嵌入式操作系统，则更加关注对中断的响应时间，更加关注线程或任务的调度算法，以使得整个系统能够在可预知的时间内，完成对外部事件的响应。

物联网操作系统的内核最主要的特点是其伸缩性。物联网操作系统的内核应该能够适应各种配置的硬件环境。同时，物联网操作系统的内核应该足够节能，

确保在一些能源受限的应用下，能够持续足够长的时间。除此之外，物联网操作系统的内核还提供面向物联网应用的常用连接功能，比如对蓝牙的支持，对 Zigbee 的支持，对 WiFi 的支持，等等。各类领域应用可以直接利用物联网操作系统内核的这些连接功能，实现最基本的通信需求。

## 外围功能组件

物联网操作系统内核只是提供最基本的操作系统功能，供物联网应用程序调用。但只有物联网操作系统内核是远远不够的，在很多情况下，还需要很多其它功能模块的支持，比如文件系统，TCP/IP 网络协议栈，数据库等。

之所以把这些功能组件称为“外围”，是因为在很多情况下，这些功能组件都不是必须的。而且在实际的物联网应用中，这些外围组件也不会全部被用到，其它的功能组件必须裁剪掉。因为在物联网应用中，很多情况下的系统硬件资源非常有限，如果保留没有用到的功能组件，会浪费掉很多资源。这些外围功能组件都是针对物联网操作系统进行定制和开发的，与物联网操作系统内核之间的接口非常清晰，具备高度的可裁剪性。

## 物联网协同框架

物联网协同框架是实现物联网“协同”功能性需求的关键功能系统。物联网操作系统的内核和外围功能组件，仅仅实现了物联网设备之间的“连接”功能。但是我们知道，仅仅实现物联网设备的连接上网，是远远不够的。物联网的精髓在于，物联网设备之间能够相互交互和协同，使得物联网设备能够“充分合作”，相互协调一致，以达到单一物联网设备无法完成的功能。而物联网协同框架，就是为物联网设备之间的协同提供了技术基础。

一般情况下，物联网协同框架是一组软件的集合，由许多个功能相互独立，但是又相互依赖的软件模块组成。比如，Google 的 Weave 物联网协同框架，是由云平台组件 Weave Cloud，面向设备端的 LibWeave，以及面向智能手机客户端的 Weave Client 等组件组成。Weave Cloud 是整个框架的“中心管理器”，所有

基于 Weave 的物联网设备，首先都连接到 Weave Cloud 上，接受 Weave Cloud 下发的指令，并向 Weave Cloud 上报相关数据。Weave Client 则也需通过 Weave Cloud 来管理和控制基于 Weave 的物联网设备，等等。

一般来说，物联网协同框架至少包括如下功能：物联网设备发现机制、物联网设备的初始化与配置管理、物联网设备之间的协同交互、云端服务等功能。

## 公共智能引擎

通过物联网协同框架，可以使得物联网设备之间建立关联，充分协作，完成单一物联网设备无法完成的功能。但是这种协同的功能，还是局限于事先定义好的逻辑上。

物联网智能引擎，就是指包含了诸如语音与语义识别，机器学习等功能模块，以使得物联网能够超出“事先定义好”的活动规则，能够具备像人一样具备“智慧”的能力。在物联网智能引擎内的功能模块，都是基础能力，可以供各种物联网应用所调用。比较典型的例子就是，在物联网设备中加入语音识别功能，人们通过自然语言，与物联网设备直接对话，来达到下达指令的目的。

## 集成开发环境

集成开发环境是任何一个完备的操作系统所必需提供的功能组件，程序员通过集成开发环境的辅助，完成具体应用的开发，这些应用最终运行在目标操作系统上。在物联网操作系统本身的开发中，会采用不同的相对专业的开发工具。在操作系统发布之后，也要提供一套完整的开发工具，方便物联网领域的程序员开发物联网应用。