

第7章 Micrium公司介绍

7.1 Micrium公司介绍

介绍前，先让大家看看 Micrium 的工作环境，非常不错。



以前讲 emWin 教程的时候，也贴过 SEGGER 公司的工作环境，效果也不错。从某种意义上来说一个好的工作环境还是很重要的。

Micrium 公司在嵌入式软件组件方面处于领导地位，该公司的旗产品 $\mu\text{C}/\text{OS}$ 系列具有无可比拟的可靠性，稳定性，完美的源代码，以及大量的文档。此外，Micrium 公司的一些组件包含符合行业标准，包括医疗电子，航空电子设备和工业产品所要求的严格的安全关键标准认证。

Jean J. Labrosse 作为 Micrium 公司的创建者和 Christian L  gar   (Micrium 公司的执行副总裁兼首席技术官) 都是广为人知和受尊敬的行业专家。他们都编写和出版过很多关于嵌入式系统方面的书籍和文章。在世界知名的嵌入式会议上也都有他们的身影。他们出版的书籍中尤其以 Jean J. Labrosse 编写的 $\mu\text{C}/\text{OS}$ 系列书籍最为出名 ,也正是这本书让成千上万的嵌入式爱好者知道了 $\mu\text{C}/\text{OS}$ 系统。国内关于 $\mu\text{C}/\text{OS}$ 的资料更是铺天盖地，特别是第 II 版。

7.2 μ C/OS的发展历程

下面的这些叙述来自 Jean J. Labrosse 本人，以第一人称叙述的（[值得一看](#)）。

1989 年，在 μ C/OS 故事开始的时候，我在佛罗里达州罗德岱堡加盟 Dynalco 控制，并开始致力于为大型工业往复式发动机设计一个新的基于微处理器的点火控制系统。

我相信一个操作系统将使得这个项目大获裨益。我最初考虑的核心是一个在过去使我受益匪浅的系统。然而代价有点高，而我的预算有点微薄。另一种选择是，我以前从来没有使用一个内核，但它是五分之一我原来选择的期望价格。最终，我决定使用不熟悉的操作系统的经济利益超过了它的高价位对应的潜在优势。



Jean J. Labrosse

但我很快就意识到，我会为看似便宜的操作系统付出我的时间代价。在接收内核后的两个月内，我在与技术支持不断接触，试图确定为什么即使是最简单的应用程序将无法运行。我终于发现，我是购买该操作系统的第一批客户之一，本质上是一个不知情的 beta 测试。

这个系统的很多问题令我沮丧，我转向我最初拒绝的昂贵系统，看来已经晚了，因为到了项目后期经费也是问题。两天后能运行简单的应用了，内核问题暂时告一段落了。

没过多久，又陷入了另一个问题中。我的一位工程师报告说，新的操作系统似乎包含一个 bug。本以为软件供应商会感兴趣，我及时向他们传达工程师的调查结果。然而我被告知的不是软件问题，而是 90 天的保修期已经到期。除非

我购买维护合同，错误就会得到解决，感觉这是在敲诈我。软件提供者不以为然，我不得不提供维护费。

令人难以置信的是，软件供应商历时半年才解决了这个 bug。总而言之，从我开始使用这个系统一年后才完成我的点火系统。显然，我需要一个更好的解决方案。

我开始开发自己的内核。我天真的认为，所有内核只是保存和恢复 CPU 寄存器，写一个不应该是特别具有挑战性。这个过程使我很忙，晚上和周末都用上了，并且证明比我预期的要困难得多。大约一年后，我的第一个操作系统，命名为 μ C/OS 正式完成

μC/OS

μC/OS 是根据我创建并发布在 Dynalco 的严格标准进行设计的。操作系统的源代码功能自由，注意间距，字斟句酌的注释，并一致的命名。虽然 μC/OS 同其它的内核一样暂时只支持少数特定的

的处理器功能，上面的那些特性使得 μC/OS 与其它的内核可以分明显的区分开。工程师可以用 μC/OS 很容易的适应新的 CPU 架构。

不幸的是，我是唯一——一个了解 μC/OS 的人。渴望把我的新软件介绍给别人，我写了一个深入的论文，解释 μC/OS 的内部运作。

嵌入式系统编程发表了修剪后的论文，并将其分为了两部分，它产生了强烈的反响。工程师们感激一个高品质的内核的内部工作被开源，他们下载了 μC/OS 的源代码。不久后，我的文章出现在嵌入式系统编程的研发出版物，C 用户杂志出版商与我联系。他们感兴趣的打印整个 μC/OS 的书。

在 1992 年年底，我的第一本，名为 μC / OS 的实时内核书正式发布。这本书有 250 页，并且提供纸质版。

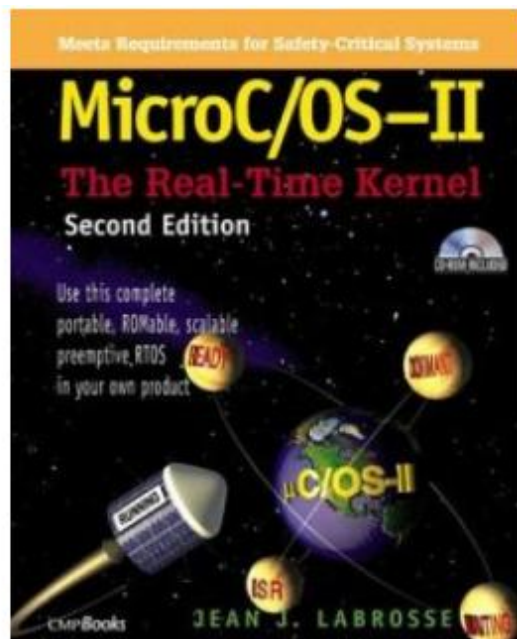
虽然初期销售量有点让人失望，每个月可以在 C 用户杂志发布我研发的 μC/OS 实时内核。与此同时，我开始获得核心专家的关注。在 1993 年的春天，我应邀参加在佐治亚州亚特兰大举行的嵌入式系统大会（ESC），在这里我研发的操作系统的获得了超过 70 多位嵌入式爱好者的支持。在几年之内，我是一个 ESC 装置，提供我的内核讲座，每次会议都会有数百名工程师。

随着作为演讲人的知名度增加，大家对我的书的兴趣也有所回升。经过努力，μC/OS 书籍的销量也超过 15000 份。

几年来，μC/ OS 的变化很小。然而，当 R & D 问我写第二版时，我决定更新书籍和系统。更新的操作系统为 μC/OS- II 。

μC/OS- II 还提供了许多功能，包括堆栈检查功能，钩子函数，和一个安全的方式来动态分配内存。

正如软件的最新版本，书籍也获得了新的名字，新版本为 MicroC / OS-II (*"Micro" was used in place of "μ" because titles incorporating Greek letters posed problems for many book retailers*) 不像我的第一个版本，这本新书是一本精装版。



MicroC / OS-II 该书发布于 1998 年。这本书伴随着源代码一起发布，我想再次会有成千上万的开发者测试内核，并提供宝贵的意见。

许多大学教授也被这本书吸引了，并开始设计围绕 μ C / OS-II 的全部课程。不久的大学生毕业生，其内核的培训侧重于操作系统的，在那里他们继续使用 μ C / OS-II。

2000 年 7 月，当 DO - 178B A 级认证被授予了使用 μ C / OS-II 的航空电子产品，这就为操作系统的可靠性提供了确切证据。此认证，通过美国联邦航空电子管理局(FAA)认可，被授予软件视可以足够安全的在飞机上使用。其它的认证包括食品及药物管理局(FDA)，pre-market notification (510(k))，pre-market approval (PMA) for

medical devices，和 IEC- 61508 的工业控制。

Micrium[®]

我成立自己的公司，并于 1999 年 9 月，Micrium 公司正式应运而生。(Micrium comes from the word 'Micro' (for

microprocessors or microcontrollers) and 'ium' (which means the Universe of) and thus, Micrium means the Universe of Microprocessors (as seen through the eyes of software).)

在 Micrium 公司合并前的几个月里，我开始了 μ C / OS-II 书籍的第二版，并于 1999 年 11 月发布，伴随着内核的新版本。添加了两个主要的功能到操作系统中：事件标志和互斥信号量。



Christian Légaré

Micrium 在不断的发展壮大。聘请工程师移植 μ C / OS-II 到新的硬件平台和开发示例项目以及应用笔记。2002 年，我的一位老朋友，Christian Légaré 加入 Micrium 公司担任副总裁。他的大量的企业和技术专长进一步加速了公司的快速成长。由于他的加入，公司从单一产品扩大到有 15 种产品组合。

作为 Micrium 的总裁，我仍然致力于编写世界一流的内核代码，比如最近的 μ C / OS-III。经过无数个小时的细致编程和产品测试，这款强大的操作系统根源于 μ C / OS-II，却又是一个全新的内核。

公司理念：**I am highly circumspect of fads and**

unproven technology as I write new software.

Although I like to keep abreast of the latest

developments in the high-tech world, the focus is on solving engineers' problems and

providing a solid and complete infrastructure, rather than on how to prematurely exploit emerging trends.

这种理念已经取得了相当的成功。Micrium 公司到现在已经有十四年了，是一位德高望重的嵌入式软件提供商。行业调查一致显示操作系统在嵌入式领域十分流行。我的目标一直是 **provide effective solutions for the same types of problems that I confronted at Dynalco, and that millions of embedded systems developers continue to face today.**

7.4 如何选择可靠的RTOS

Micrium 公司曾经发过一片文章，详细的讲解了如何选择可靠的 RTOS，这篇文章值得一读，我把这篇文章的相关内容整理了一下，初学者有时间可以读一读。

● 如何选择可靠的 RTOS

- 现在的 RTOS 供应商很少提及他们的 OS 在安全性方面做的努力，对于这些方面也是支支吾吾，所以有些时候对于开发人员来说选择 RTOS 成一个赌博，所以很多时候大家都会选择有安全认证的。

● RTOS 的评估

- 源码的重要性，有源码才能很好的评估这个 RTOS，远比文档里面吹嘘我们的 RTOS 多好多好强很多。
- 现在的一些 RTOS 厂家会提供在 PC 机上面运行的环境，这个用于了解 API 就行，不能用于测试任务实际的执行。
- 特别是一些复杂的应用，测试 RTOS 的安全性非常麻烦，而且耗时间。

● 历史和声誉

- 一个 RTOS 发展的过程当中，会有一个 Release Notes 记录着这个 OS 所修改的以前的 BUG 和已经新增加的功能，我觉得这个很重要，有些 OS 看着升级很快，光升级一些新的功能，隐藏的 BUG 一点没改。
- RTOS 供应商好的口碑也非常重要，大家肯定都喜欢用口碑好的。

● 认证

- 关于认证，这个文章里面讲了很多认证的过程以及需要做的事情，感觉比较的详细，有兴趣的可以看看。