

&1、TreeOS 是一个怎样的的操作系统？

TreeOS1.0 物联网实时操作系统是一套称之为 MCUSDS 的“单片机标准化系统”。它是一个真正适用于 51 单片机的实时操作系统；它无核，含有大量驱动库，可以帮助你完成 60%~90%软件开发任务的实时操作系统；一个适合初学者学习、使用的单片机操作系统。TreeOS1.0 填补了目前世界上中低档微控制器无适用操作系统的重大难题。

&2、TreeOS 与其它嵌入式实时操作系统有什么比有什么优势？

- 1、TreeOS 系统占用资源小，可适用于低端单片机；
- 2、TreeOS 含有大量驱动的库，节省开发时间；
- 3、TreeOS 采用无核设计，调试简洁明了；
- 4、TreeOS 采用构件化设计，可以通过灵活的配置裁剪出紧凑高效的应用；
- 5、TreeOS 需要的运行 RAM 更小，对 MCU 的要求相对较低。

&3、TreeOS 与其他操作系统有无内核区别？

单片机内存容量，程序空间等资源非常有限，操作系统内核需占用相当一部分内存和程序空间，这就制约了系统的性能；运行内核程序需要占用处理器时间，因此降低了系统实时能力，不利于降低功耗；操作系统内核程序非常复杂，学习较为困难；这些单片机操作系统所具有的传统内核，对于需要大量设计驱动程序，常用程序库等。

而 TreeOS1.0 操作系统因其“无核”的设计思想占用系统资源(内存、程序空间、MCU 运行时间)的问题,使 MCS51 单片机等中低挡单片机也能用上操作系统,TreeOS1.0 最大的好处就是提供了可复用的软件体系架构以及可复用的软件构件库,简化了软件开发过程,缩短了开发周期。

&4、TreeOS 系统的可靠性与稳定性如何?

该系统由专家编写,已经获得专利,并应用于大量的项目,可靠性与稳定性毋庸置疑。

&5、TreeOS 使用起来是否困难?

TreeOS 有着标准化软件、标准化电路、可靠性设计经验。一些标准的驱动程序和常用程序库,可以直接拿来就用,大大减少工作量。使用时会有说明文档,帮助工程师们快速使用,开发到自己的项目中。

&6、TreeOS 对初学者的基础要求如何?

有简单的 C 语言基础,简单的单片机基础和电子技术基础基本上就可以使用我们的 TreeOS 操作系统。

&7、TreeOS 操作系统有什么作用?

(1)TreeOS 操作系统提供了一些标准的驱动程序和常用程序库,可以直接拿来就用,大大减少工作量;

(2)操作系统提供了完善的编程思路,使大量的任务可以有条不紊地在一起工作,减少了出错的可能性;

(3)对于初学者来说,可以大大缩短自身的积累过程,助其快速

成长;

&8、TreeOS 操作系统适用于哪些行业?

物联网、工业 4.0、人工智能、航天航空、军事电子、工控设备、智能仪表、机器人、汽车电子、智能家居、信息家电、办公自动化设备、通讯设备等行业。

&9、使用 TreeOS 系统开发项目的效果?

开发人员从底层做起，需要做大量的重复性劳动。使用 TreeOS 之类的操作系统，则只需要编写用户程序，很多现成程序可以拿来就用，也不必将所有任务运行的各种情况记在心中，这就简化了开发的过程，减少了工作量。能完成 60~90%软件开发任务。

&10、TreeOS 适用于什么类型的单片机?

中低档微控制器，单片机的内存容量、程序空间等资源就非常有限，而操作系统内核需要占用相当一部分的内存和程序空间，这就制约了系统的性能。然而 TreeOS 操作系统没有内核，不占用内存，则中低档的单片机比较适用。如目前支持 STC51，MSP430，AVR，STM32 这些主流 MCU，未来我们会添加更多的 MCU 支持。

&11、TreeOS 驱动是否容易移植?

所有的驱动程序都不会有比较复杂的语句，都是简单易用的语句，SPI 接口和 IIC 接口都是采用模拟方式，易于移植到其他的单片机。

&12、TreeOS 的关键性技术？

- (1) 采用近年来兴起的软件构件化技术
- (2) 采用无核设计，解决操作系统挤占系统资源问题
- (3) 业内首创“面向场景”编程，把场景按树形结构组织起来同时提供了涉及应用层的解决方案；
- (4) 任务网格化，软件结构清晰明了。

&13、我们应该怎么样来学习 TreeOS 操作系统？

在学习 TreeOS 操作系统时，首先得有简单的单片机基础、C 语言基础、电子技术基础。然后看说明文档，学会调用驱动函数，来实现你的项目。

&14、什么是 TreeOS ComLib 软件构件库？

TreeOS ComLib 软件构件库是 TreeOS 1.0 构件化操作系统的主要组成部分。

TreeOS ComLib 软件构件库主要包含以下四方面的内容：

- 1) 各种 MCU 的初始化模板、中断服务程序模板；
- 2) 各种周边设备的驱动程序；
- 3) 常用的库程序；
- 4) 可通用的场景库程序或模板。

&15、TreeOS ComLib 有哪些特点？

- 1) 符合 TreeOS 1.0 软件架构规范
- 2) 可方便移植到不同的单片机平台

- 3) 专家编写, 经过实践检验
- 4) 不断升级、扩展

&16、如何使用 TreeOS ComLib 软件构件库?

1) 复制: 根据项目需要, 将.c 和.h 文件复制到项目文件夹中, 并把 C 文件调入工程。

2) 配置: 一般配置都在.h 文件中进行, 包括条件编译选择、属性、IO 口定义、特殊语句等。需要配置之处, 带有“//M/”标志;

3) 裁剪: 有些子程序用不着, 必须把它注释掉, 以免占用内存和 ROM 空间。一般通过.h 文件中的“条件编译选择”进行裁剪。需要裁剪之处, 带有“//M/”标志;

4) 引用: 包括调用子程序和引用“特殊语句”。最常见的引用“特殊语句”是放在 T0 中断中进行倒计时, 例如 TreeOS_keyboard 中的 iSACN_KEY_TIME;

5) 修改与补充: 有些子程序需要根据实际情况进行编写, 例如 TreeOS_keyboard 中的 ReadPress(void)。需要修改与补充程序之处, 带有“//M/”标志;

6) 模板: 有些文件只是一个模板, 一般只提供架构, 不提供具体内容“有时提供例程”可以按照该架构来编写程序, 同样带有“//M/”提示标志。

&17、当初在什么样的情况下才想做一套这样的系统?

TreeOS 物联网实时操作系统的诞生源于项目, 在做过的众多

项目中，有很多通用的地方，能不能做一个系统将项目间相同功能存放起来，为以后的项目开发缩短周期，所以 TreeOS 的诞生并非横空出世，而是一个真正的“从实践中来，到实践中去”的产品，经历了一百多个单片机项目的提炼和验证，以及理论上的升华，版本不断升级，于 2012 年形成了专用于单片机的 TreeOS1.0 版本，真正实现了从量变到质变的过程。可以使嵌入式软件的开发避免了大量的冗余操作，从而使软件的开发更稳定、更高效！

&18、TreeOS 未来发展方向？

TreeOS 将会不断升级，主要沿两条线路：一是不断扩充构件库以及构建专用的构件库，这个任务量很大；二是对任务的管理可以考虑加上内核或微内核，以适用更复杂的嵌入式应用。

&19、Treeos 是面向物联网的，和华为 liteos, Tinyos, Contiki 等物联网操作系统有什么优势呢？

相比之下，我们的 Treeos 有一个构件库提供丰富的驱动程序，加快开发的进度。

&20、使用 TreeOS 出现 bug 容易调试吗？

像树形一样的架构的编程思路，很容易定位、解决问题。

&21、TreeOS 有什么缺点吗？

整个循环的执行时间不是一个常数，随着任务数的增加，任务等待执行的时间也随之增加。

&22、我们的 Treeos 系统是主打物联网的，对吧，那么它也一定支持许多物联网相关的标准了，现在支持

哪些？

现在支持的标准包括 ZigBee，蓝牙，Lora，wifi，像比较新的 NB-IOT，我们也在做。

&23、关于 Treeos 配套的构件库，具体包含哪些内容，然后它要做到哪种规模？

主要是 MCU 相关的外围电路的，相关的驱动程序，像 SD 卡读写，CAN 总线，485 总线，RF 通信，TCP/IP 协议栈等等

&24、在未来 TreeOS 使用人数激增的情况下，是否开源？

仍然开源，只有在开源的情况下，才有可能促使 TreeOS 不断发展，才有使用人数激增的情况，吸引数十百万的物联网工程师。

www.treeos.com