

Java能扮演嵌入式应用开发的主角吗？PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/252/2021_2022_Java_E8_83_BD_E6_89_AE_c104_252525.htm Java自从推出以来一直备受关注，不过在嵌入式系统设计师看来，其性能并不能令人满意。运行Java虚拟机（JVM）解释Java字节码，这种方式对大多数嵌入式应用来说占用空间过多，运行速度过慢。不过Sun的Java 2 Micro Edition（J2ME）改变了这种状况。对嵌入式系统设计师来说，Java有许多优点。作为一门编程语言，Java允许面向对象编程，又没有C中存在的严重问题。Java运行时环境还提供了有用属性。Java提供的内存管理功能使得编程人员不必分配及释放内存。运行时环境甚至可以通过整合核心类库来简化程序分配。但大多数嵌入式应用面临Java没有处理好的两大约束：没有足够的空间和时间。那么，Java作为一种在C基础上改进了的面向对象的语言，在嵌入式应用开发方面能挑大梁吗？能否为它自己撑起一片蔚蓝的天空呢？

一、为什么会是Java？对于嵌入式系统来说，Java技术比C语言和汇编语言具有很明显的优越性。最显著的特点是减少了系统的开发和维护，增强了代码的重利用能力，提高了Java代码与系统原有代码的可整合性。

1. 提高开发效率和可维护性 在项目的整个生命周期中，Java环境大大的简化了开发和维护。由于目标环境是建立在一个虚拟机上，代码可以很容易的编写、调试、分析、更改、维护。加上将来要连接的外接设备，未来的系统可能比目前的嵌入式系统复杂很多。升级手册也许不会在整个项目的生命周期中都能起到作用。取而代之的是，硬件设备的可连接性使得能够远程管理模块，这样就

保证了开发人员能在产品上增加新的性能，同时解决了在产品生产后软件升级和维护的问题。

2. 重复利用代码 由于嵌入式系统有特殊的需求，以及不同的专门硬件要协同工作，嵌入式软件开发者通常使用非常原始的方法来开发，有时每一个新的项目都要从头再来一遍。现在，随着嵌入式技术的成熟以及系统本身变得更大更优化，很多人开始对于把一个产品的模块甚至是全部的应用程序用到另一个产品感兴趣。这种可重新利用性使得"一次开发，多次利用"成为了可能。Java环境使得一个模块可以只要做很少的工作就可以适应多个项目和平台。甚至包括有时客户需要一个新的目标板，或者采用新的硬件（CPU或外设）和软件，或者使用不同的Linux都可以进行移植。

3. 集成Java代码和源代码 使用源代码明显是指应用程序的多可用性以及代码的重利用能力。在Java应用中，一个设计很好的界面，或者虚拟机，或者是底层的硬件都可以很好的兼容到嵌入式系统中。尽管无法移植，对于很多功能和硬件界面来说，在本地环境下开发的代码也许仍然是最好的解决方案。在C、C 或者汇编语言中，加入标准的通信、接口模块、用户界面、安全特性会花费很多时间与金钱。与之相比较，Java的基本库本身就提供了这些东西甚至还更多，这样就可以加速开发。

二、Java碎片真的会有影响吗？

在使用JavaME CLDC进行移动电话开发时，人们经常会碰到碎片这个词。Java强调“一次开发，多次利用”，但碎片出现，却打破了这种传奇。于是，这就导致应用开发人员不得不在许多不同的设备进行应用程序的测试，甚至于不得不在应用程序中对某些特殊的设备进行一步客户化的工作。对程序开发人员来说，碎片真是个恶梦，因为碎片平白无故的增添了

代码量和测试工作量。当然，对移动电话持有者来说也不是什么好事，因为碎片消耗了设备的空间。不管怎么说，碎片对每个人来说都是件很讨厌的事情。但对于嵌入开发者而言，碎片又意味着什么呢？首先来看看碎片产生的根源。移动电话行业标准本来给不同的产品预留了一定的自由空间，这初衷是好的。但事实上，这种预留的空间，却导致了不同产品之间的冲突，不能进行很好的兼容。这就是碎片产生的最根本原因。于是这种不兼容性进而升级到了Java实现的程序里。这正是Java想花大力气创建一个统一Java实现的原因所在，如JSR248，MSA（Mobile Service Architecture）的建立。从嵌入式开发人员的角度来看，也许并没有这么糟糕。其实碎片并不会影响到嵌入式开发人员，因为已经可以确定设备之间的硬件是完全兼容的。如果使用的是原始语言像C/C++的话，嵌入式开发人员可以在任何地方来编写代码，并在不同的设备上进行代码的重用。100Test下载频道开通，各类考试题目直接下载。详细请访问www.100test.com