

## 单片机开发中应掌握的基本技巧

在单片机应用开发中，代码的使用效率问题、单片机抗干扰性和可靠性等问题仍困扰着工程师。为帮助工程师解决单片机设计上的难题，现根据论坛中的讨论归纳出单片机开发中应掌握的几个基本技巧。

### 一、 如何提高 C 语言编程代码的效率

邓宏杰指出，用 C 语言进行单片机程序设计是单片机开发与应用的必然趋势。“如果使用 C 编程时，要达到最高的效率，最好熟悉所使用的 C 编译器。先试验一下每条 C 语言编译以后对应的汇编语言的语句行数，这样就可以很明确的知道效率。在今后编程的时候，使用编译效率最高的语句。”

他指出，各家的 C 编译器都会有一定的差异，故编译效率也会有所不同，优秀的嵌入式系统 C 编译器代码长度和执行时间仅比以汇编语言编写的同样功能程度长 5-20%。“对于复杂而开发时间紧的项目时，可以采用 C 语言，但前提是要求你对该 MCU 系统的 C 语言和 C 编译器非常熟悉，特别要注意该 C 编译系统所能支持的数据类型和算法。

虽然 C 语言是最普遍的一种高级语言，但由于不同的 MCU 厂家其 C 语言编译系统是有所差别的，特别是在一些特殊功能模块的操作上。所以如果对这些特性不了解，那么调试起来问题就会很多，反而导致执行效率低于汇编语言。”

### 二、 如何减少程序中的 bug?

对于如何减少程序的 bug，邓宏杰给出了一些建议，他指出系统运行中应考虑的范围管理参数有：

1. 物理参数。这些参数主要是系统的输入参数，它包括激励参数、采集处理中的运行参数和处理结束的结果参数。合理设定这些边界，将超出边界的参数都视为非正常激励或非正常回应进行出错处理。

2. 资源参数。这些参数主要是系统中的电路、器件、功能单元的资源，如记忆体容量、存储单元长度、堆叠深度。在程式设计中，对资源参数不允许超范围使用。

3. 应用参数。这些应用参数常表现为一些单片机、功能单元的应用条件。如 E2PROM 的擦写次数与资料存储时间等应用参数界限。

4. 过程参数。指系统运行中的有序变化的参数。

### 三、 如何解决单片机的抗干扰性问题

邓宏杰指出：防止干扰最有效的方法是去除干扰源、隔断干扰路径，但往往很难做到，所以只能看单片机抗干扰能力够不够强了。单片机干扰最常见的现象就是复位；至于程序跑飞，其实也可以用软件陷阱和看门狗将程序拉回到复位状态；所以单片机软件抗干扰最重要的是处理好复位状态。

一般单片机都会有一些标志寄存器，可以用来判断复位原因；另外你也可以自己在 RAM 中埋一些标志。在每次程序复位时，通过判断这些标志，可以判断出不同的复位原因；还可以根据不同的标志直接跳到相应的程序。这样可以使程序运行有连续性，用户在使用时也不会察觉到程序被重新复位过。

#### 四、 如何测试单片机系统的可靠性

有读者希望了解用用什么方法来测试单片机系统的可靠性，邓宏杰指出：“当一个单片机系统设计完成，对于不同的单片机系统产品会有不同的测试项目和方法，但是有一些是必须测试的：

1. 测试单片机软件功能的完善性。这是针对所有单片机系统功能的测试，测试软件是否写的正确完整。
2. 上电、掉电测试。在使用中用户必然会遇到上电和掉电的情况，可以进行多次开关电源，测试单片机系统的可靠性。
3. 老化测试。测试长时间工作情况下，单片机系统的可靠性。必要的话可以放置在高温，高压以及强电磁干扰的环境下测试。
4. ESD 和 EFT 等测试。可以使用各种干扰模拟器来测试单片机系统的可靠性。

例如使用静电模拟器测试单片机系统的抗静电 ESD 能力；使用突波杂讯模拟器进行快速脉冲抗干扰 EFT 测试等等。邓宏杰强调：“还可以模拟人为使用中，可能发生的破坏情况。人体或者衣服织物故意摩擦单片机系统的接触端口，由此测试抗静电的能力。用大功率电钻靠近单片机系统工作，由此测试抗电磁干扰能力等。”