

# 论 RTX51 单片机实时操作系统

徐 哲

(齐齐哈尔工程学院 信息工程系, 黑龙江 齐齐哈尔 161005)

**摘 要:** 很多单片机的应用中都需要同时执行很多任务。对于这样的应用, 可以利用实时操作系统来灵活地安排系统资源。RTX51 就是一种应用于 MCS51 系列单片机的实时多任务操作系统, 它可以工作在所有 8051 单片机以及派生家族中, 简化了复杂的软件设计, 缩短了项目周期, 在实践中用 RTX51 来开发单片机软件, 具有诸多优越性。

**关键词:** RTX51; 8051 单片机; 操作系统

**中图分类号:** TP332.3 **文献标识码:** A **文章编号:** 2302011 (2012) 03-0022-03

## 0 引言

近年来, 利用嵌入式实时操作系统来开发嵌入式系统软件已是大势所趋。这是因为利用嵌入式实时操作系统, 程序开发者可以将复杂的系统分成多个相对独立的任务, 从而分而治之, 降低了系统的复杂性; 并通过对不同的任务赋以不同的优先级, 从而能够保证系统的实时性; 同时, 由于软件系统有多个任务组成, 所以其模块化程度好, 提高了系统的可维护性。

## 1 RTX51 简介

RTX51 是美国 Keil 公司开发的一个用于 8051 系列单片机处理器的多任务实时操作系统, 它可以简化那些复杂而且时间要求严格的工程的软件设计工作。RTX51 有两个模式, 即 RTX51 完全模式 (Full) 和最小模式 (Tiny)。RTX51 Full 有 4 个任务优先级, 可以和中断函数并行处理, 各个任务之间通过使用“邮箱”系统来进行信号和消息的传递, 可以从内存池中申请和释放内存; 同时, 可以强制一个任务停止执行, 等待一个中断, 或者是其他中断传来的信号量或者消

息。RTX51 Tiny 是 RTX51 Full 的一个子集, 可以很容易地运行在 8051 单片机系统上, 而不需要外部 RAM (XDATA)。目前, 使用较多的是 RTX51 Tiny, 因为 RTX51 Tiny 不仅是免费的, 而且功能强大, 完全满足系统软件开发的需要, 它可以灵活的分配硬件系统资源 (CPU、存储器等) 给各个任务, 从而大大的缩短了程序开发的时间并增强了软件工作的稳定性, 对嵌入式操作系统在工业控制领域的应用具有先导作用。

## 2 RTX51 系统的特点

RTX51 可以在所有的 8051 系列芯片上运行, 用户只需要用标准的 C 语言编写 RTX51 程序, 然后用 C51 编译器编译即可生成代码。其中, 仅有少数内容和标准 C 语言有差异, 这些内容是为了实现任务标识和优先级而设置的。RTX51 程序设计需要包含实时运行头文件和必要的库文件, 并且要用 BL51 连接/定位器来实现连接。在 keil 中, 你只需要在目标选项的 Target 标签中的 Operating 中选择 RTX51 Tiny, 在你的头文件中加上 #include<rtx51tiny.h>即可。

收稿日期: 2012-08-22

作者简介: 徐 哲, 男, 齐齐哈尔工程学院信息工程系教师, 研究方向: 应用电子技术。

### 3 RTX51 事件

(1) 超时: 挂起运行的任务指定数量的时钟周期。

(2) 间隔: 类似于超时, 但是软件定时器没有复位, 典型应用是产生时钟。

(3) 信号: 用于任务内部同步协调。

(4) 消息: 适用于 RTX51 Full, 用于信息的交换。我们可以把一个消息发送到一个特定的邮箱。消息由 2 字节组成, 可以是用户按照自己的需求定义的数据, 也可以是指向数据的指针。如果邮箱的消息列表已满, 而且是中断发送消息, 这个消息将会丢失; 如果是任务发送消息, 那么任务将会进入到等待状态, 直到邮箱重新有了位置可以接收这一条信息。邮箱是按照 FIFO 的原则来管理消息的, 如果几个任务都在等待接收消息, 那么最先进入等待接收队列的将接收消息。一个邮箱最多可以存储 8 条消息。当邮箱满的时候, 最多只能有 16 个等待任务。

(5) 中断: 适用于 RTX51 Full, 一个任务可以等待 8051 硬件中断。

(6) 信号量: 适用于 RTX51 Full, 信号量用于管理共享的系统资源。通过使用“令牌”, 允许在同一时刻只有一个任务使用某些资源。如果几个任务申请访问同一个资源, 那么首先提出申请的将允许访问, 其他的任务进入等待队列, 直到第 1 个任务操作完毕, 下一个任务才能继续。

### 4 RTX51 任务和任务状态

#### 4.1 RTX51 任务

RTX51 区分两类任务, 即快速任务和标准任务。快速任务有很快的响应速度, 每个快速任务使用 8051 一个单独的寄存器组, 并且有自己的堆栈区域。RTX51 支持最大同时有 3 个快速任务。标准任务需要多一点的时间来进行任务切换, 因此使用的内部 RAM 相对快速任务要少, 所有的标准任务共用 1 个寄存器组和堆栈。当任务切换的时候, 当前任务的寄存器状态和堆栈内容转移到外部存储器中。RTX51 支持最大 16 个标准任务。

#### 4.2 任务状态

(1) 运行 (RUNNING) ——当前正在运行的

任务处于 RUNNING 状态, 同一时间只有 1 个任务可以运行。

(2) 就绪 (READY) ——等待运行的任务处于 READY 状态, 在当前运行的任务退出运行状态后, 就绪队列中优先级最高的任务进入到运行状态。

(3) 阻塞 (BLOCKED) ——等待一个事件的任务处于 BLOCKED 状态, 如果事件发生且优先级比正在运行的任务高, 此任务进入运行状态; 如果优先级比正在运行的任务低, 此任务进入 READY 状态。

(4) 删除 (DELETED) ——没有开始的任务处于删除状态。

### 5 RTX51 中断处理

RTX51 完全模式提供两种方法来处理中断, 即一种是 C51 的中断函数, 另一种是 RTX51 的任务中断。它又可分为快速任务中断和标准任务中断。对于中断函数这种方法, 它同时也可以在不使用 RTX51 的情况下使用, 当中断发生的时候, 程序就跳到了相应的中断函数, 它和正在运行的任务是相互独立的, 中断的处理是在 RTX51 系统之外, 和任务切换规则没有关联。对于任务中断的方法, 不管使用快速还是标准任务来处理中断, 如果中断发生, 等待中断的任务就从“等待”状态进入到就绪状态, 并按照任务切换规则进行切换。这种中断处理是完全集成在 RTX51 的内部, 硬件中断事件的处理和信号、消息的处理是完全相同的。在系统响应时间上中断函数是最快的。RTX51 必须完全控制中断使能寄存器, 这样才能遵守任务的切换规则, 并保证中断程序的无误进行。必须注意中断使能寄存器是由 RTX51 完全控制的, 禁止用户手动修改。

### 6 RTX51 应用中应注意的问题

原则上讲 RTX51 可以应用到基于 MCS51 系列单片机组成的任何系统中, 但由于 RTX51 在运行时需要占用 CPU 的部分硬件资源, 如通常占用定时/计数器 T0, 且对堆栈深度要求较高, 因此, 在使用时要注意 RTX51 对硬件配置的要求。

#### 6.1 RTX51 对硬件的要求

RTX51 分为完全模式和最小模式, 对于

RTX51 完全模式, 要求系统要有外部的 RAM, 同时还应合理分配 CPU 的硬件资源, 留出定时器/计数器 T0 作为系统多任务执行的时间片发生器; 对于最小模式, 由于没有外部的 RAM, 定时/计数器 T0 不能使用, 而是留给系统作多任务切换的时间片发生器。

## 6.2 合理分配任务的时间片和任务的优先级

在执行多任务时, 由于各任务处理时的时间不同, 所以应合理的分配各个任务的时间片数, 尽量使任务在规定时间内完成, 减少任务间的消息传递。此外, 在 RTX51 完全模式下, 应注意各个任务的优先级配置, 而在最小模式下, 各个任务是没有优先级规定的。

## 7 RTX51 TINY 的特性和系统函数

### 7.1 RTX51 TINY 的特性

- (1) 可以在没有外部 RAM 的系统中运行;
- (2) 只支持循环方式和信号方式的任务切换;
- (3) 不支持优先级;
- (4) 不包含信息程序;
- (5) 没有存储池分配机制;
- (6) 使用 TIMER0 做系统时钟;
- (7) 默认地使用了第 1 组寄存器, 不过这可以通过其配置文件进行重新配置;
- (8) 支持最多 16 个任务;
- (9) 每一个系统时钟中断大约用了 100~700 个机器周期;
- (10) 把所有的任务都分配到第 0 组寄存器,

所有任务的函数必须使用 C51 的默认设置进行编译。中断则不受这个限制。

### 7.2 RTX51 TINY 的系统函数

- (1) isr\_send\_signal 从一个中断发送一个信号到一个任务
- (2) os\_clear\_signal 删除一个发送的信号
- (3) os\_create\_task 移动一个任务到运行队列
- (4) os\_delete\_task 从运行队列中删除一个任务
- (5) os\_running\_task\_id 返回当前运行的任务的标识符 (task ID)
- (6) os\_send\_signal 从一个任务发送一个信号到另一个任务
- (7) os\_wait 等待一个事件
- (8) os\_wait1 等待一个事件
- (9) os\_wait1 os\_wait2 等待一个事件

## 8 结语

实践表明, RTX51 多任务实时操作系统既能保证对外界的信息以足够快的速度进行处理, 又能并行运行多个任务, 具有实时性和并行性的特点, 因此能很好地完成对多个信息的实时测量、处理和进行相应的多个实时控制。此外, 使用 RTX51 实时操作系统, 降低了软件编写的复杂度, 提高了产品的开发效率, 方便了系统的维护和功能扩展, 在硬件资源足够的情况下, 效果更加明显。

### 参考文献:

- [1] 阳艳, 蒋礼, 杨科灵, 罗少轩. 嵌入式操作系统 RTX51 Tiny 的分析及应用 [J]. 计算机技术与发展, 2006 (06).
- [2] 李仕涌, 谭南林. 多任务操作系统在嵌入式系统开发中的应用 [J]. 北方交通大学学报, 2002 (04).