

# 医用电子直线加速器控制软件的测试方法

涂施<sup>1</sup>, 蔡建平<sup>1</sup>, 谢会东<sup>2</sup>

(1. 北京工业大学 软件学院, 北京 100124;

2. 北京大基康明医疗设备有限公司, 北京 100176)

**摘要:** 为了实现对嵌入式控制软件的测试, 利用开发工具 Realview MDK 提供的仿真测试技术进行前期测试, 并设计一种基于 XML 脚本语言描述测试用例的自动化测试方法完成功能测试, 从而较完善地满足了对该软件的测试需求。

**关键词:** 电子直线加速器; 嵌入式软件测试; 控制软件

中图分类号: TP311.54

文献标识码: A

文章编号: 1674-7720(2013)12-0004-03

The method of testing medical electronic linear accelerator's controlling software

Tu Shi<sup>1</sup>, Cai Jianping<sup>1</sup>, Xie Huidong<sup>2</sup>

(1. School of Software Engineering, Beijing University of Technology, Beijing 100124, China;

2. Beijing Top Grade Healthcare Co., Ltd., Beijing 100176, China)

**Abstract:** In order to test the controlling software, we can design a kind of automation testing method based on XML script language description test cases, combining with development environment Realview MDK which provides the simulation testing technology, so as to better meet the requirements of the software testing.

**Key words:** medical electronic linear accelerator; embedded software test; controlling software

医用电子直线加速器控制软件是一种使用在医疗设备上的嵌入式应用软件。目前市面上对嵌入式软件测试的工具一般都价格昂贵, 而且不同的嵌入式软件不完全适合医用电子直线加速器控制软件的测试; 其次, 嵌入式软件都与硬件相结合, 测试起来更有难度。为解决这一难题, 设计一套较为通用的测试方法来满足对该软件的功能测试需求, 具有很好的现实意义。

## 1 医用电子直线加速器控制软件的硬件系统

放射治疗在对恶性肿瘤的治疗中起着重要的作用, 而放射治疗技术的发展与加速器技术的发展紧密相关<sup>[1]</sup>。当前, 医用加速器的代表是医用直线加速器, 因而对其软件的可靠性和功能性的要求更高于其他领域。

本文中的医用电子直线加速器控制软件采用 CE9200-3 ARM 开发板, 为开发提供基本的底层硬件、系统和驱动等资源。其主要配置包括: CPU 为 AT91RM9200, 7 片 16 路输入的 ADC 板子, 3 片数字输入输出板。对于该软件的测试, 与一般软件的很多测试方法是通用的, 但是由于它是嵌入式软件, 而且涉及到硬件, 需要采用一些特殊的测试方法来区别对待。

## 2 基于 MDK 的软件仿真测试

在一般的嵌入式系统开发中, 软件的开发和测试都是在目标硬件平台初步开发完成之后进行的。Keil 公司开发的 ARM 开发工具 MDK 是用来开发基于 ARM 核的系列微控制器的嵌入式应用程序<sup>[2]</sup>。RealView MDK 可以使嵌入式系统的软硬件协同开发、同步进行。

### 2.1 MDK 软件仿真器

大多开发工具仅局限于对内核指令集的仿真, 而 Realview MDK 的软件仿真器不但可以仿真 ARM 的指令集, 还能仿真许多外围设备, 包括串口、外部 I/O、中断以及片上外设在内的整个目标硬件<sup>[3]</sup>。不仅如此, 其强大之处还体现在它的各种调试工具上, 通过与这些调试工具进行交互, MDK 的软件仿真器可以精确地仿真各种片上外围接口以及这些接口与外部芯片通信的过程。AT91RM9200 软件仿真器包括以下仿真模块: PWM、UART、I<sup>2</sup>C、SPI、DMA、ADC、NAN Flash 控制器、看门狗定时器<sup>[4]</sup>。其中, 每一个仿真模块都对应一个对话框, 该对话框包含了相应接口的所有寄存器。

当使用 AT91RM9200 软件仿真器调试程序时, 可以

《微型机与应用》2013 年 第 32 卷 第 12 期

在这些外围接口对话框中观察到每一个寄存器值的状态及其改变的情况;还可以通过修改这些寄存器的值来达到控制程序运行的目的。软件仿真器的每一个接口模块都提供了引脚寄存器 VTREG,用于记录 CPU 引脚的状态。这些引脚寄存器的值可以通过逻辑分析仪、调试脚本显示和修改,使得嵌入式软件的调试变得更加简单易行。图 1 为通过调试脚本向 AT91RM9200 软件仿真器的 ADC 模块的引脚 ADC2 输入正弦信号,然后通过逻辑分析仪将这个引脚寄存器的状态显示出来的仿真图。

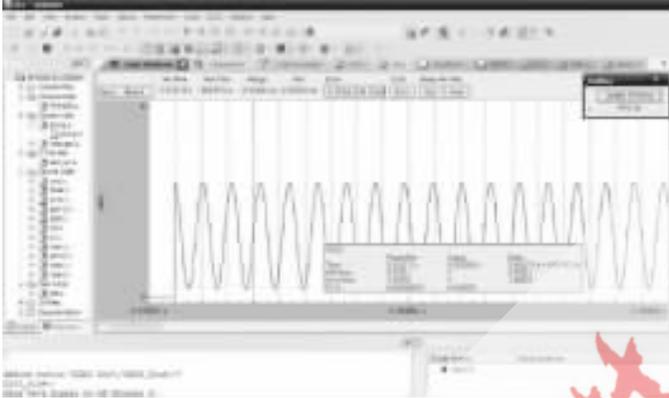


图 1 ADC2 的逻辑分析仪仿真图

## 2.2 UART 通信仿真

很多微处理器都提供了 UART 或 RS232 串行接口<sup>[5]</sup>。μVision4 调试器可仿真微处理器 UART 接口,并提供了 3 个像操作终端那样方便的串行接口。通过 Peripherals 菜单很容易查看并设置通信的波特率和其他参数,当然也可以在程序中编写通信设置的参数。通过模拟超级终端,软件就可以在没有硬件支持的情况下得到如图 2 所示结果。



图 2 UART1 模拟仿真

## 2.3 GPIO 仿真

GPIO 的仿真功能可以通过以下的小程序实现:依次闪烁每个 LED,先从前向后,再由后至前,如此反复连续。其功能函数代码如下:

```
int main (void) {
    unsigned int i;                /*LED var */
    IODIR1 = 0x00FF0000;
                                /* P1.16..23 defined as Outputs */
    while (1) {                    /*Loop forever */
        for (i = 1<<16; i < 1<<23; i <= 1){
```

```
                                /* Blink LED 0,1,2,3,4,5,6 */
        IOSET1 = i;                /*Turn on LED */
        wait ();                    /*call wait function*/
        IOCLR1 = i; }              /*Turn off LED*/
    for (i = 1<<23; i>1<<16; i>=1){
                                /* Blink LED 7,6,5,4,3,2,1 */
        IOSET1 = i;                /* Turn on LED */
        wait ();                    /* call wait function */
        IOCLR1 = i; }              /* Turn off LED */
    }
}
```

仿真调试的片上外围设备状态如图 3 所示,GPIO Port1 外设对话框中的 Output 的第 22 位被置位,表示对应的 LED 灯被点亮。从图中 GPIO 的逻辑分析仪信号中可以清晰看出 LED 闪烁的全过程,并直接查看 IO 引脚的值。



图 3 GPIO1 输出的时序逻辑

## 2.4 性能测试

RealView 作为对医用电子直线加速器控制软件仿真测试还能做到对其代码覆盖率进行测试,这在开发和测试过程中都比较实用。μVision4 仿真器的执行剖析器能够显示已知地址区域执行统计的信息。对没有调试信息的地址区域,显示列表中不会显示这块区域的执行情况。μVision4 的性能分析仪则可用于显示整个模块的执行时间及各个模块被调用的次数。μVision4 的仿真器可以记录整个程序代码的执行时间及函数调用情况。

当需要优化代码时,可以修改后直接进行仿真,从执行的时间测试出代码优化程度,很方便直观。

## 2.5 覆盖统计

μVision4 提供了一个代码统计执行情况的功能,该功能以代码统计对话框的形式表示出来。在调试框中,已执行的代码行在左侧以绿色标出。当测试嵌入式应用程序时,可用此功能查看哪些程序还没有被执行。代码统计框提供了程序中各个模块及函数的执行情况,如图 4 所示。

在 Current Module 下拉列表框中列出了程序中的所有模块,在下面的列表框中显示了相应模块中指令的执行情况,即每个模块或函数的指令执行的百分比,只要执行了的部分均以绿色标出,在 Output Window 的 Command 页中可以用 Coverage 调试命令将此信息输出到输



图4 代码覆盖率

出窗口中。当某个函数的执行率覆盖不足90%以上,可以查看没有被绿色覆盖的代码,然后通过改写代码或是加上宏定义的测试函数执行未执行的代码和路径,这也是一种测试覆盖率的好方法。

### 3 基于XML脚本的自动化测试

医用电子直线加速器控制软件的大部分测试用例都是从上位机发送给下位机的操作指令,从而控制下位机的治疗床运动,因此测试部分的指令是测试该控制软件的核心。由于指令部分的测试用例多,而且需要反复测试,单纯依靠手工测试非常耗时,因而需要考虑自动化测试来提高测试效率。

#### 3.1 测试用例的XML设计

作为W3C的一个数据交换推荐标准,XML已经被广泛应用于很多领域。用XML作为测试脚本的统一格式,可以降低测试脚本与测试工具之间的耦合度,提高测试的可扩展性<sup>[6]</sup>。XML允许定义自己的标记集,“形式良好”的XML文件才能被正确地解释处理。在XML中一个标签用于描述一段数据;另一个标签可分为开始标签和结束标签。在开始标签和结束标签之间,又可以使用其他标签描述其他数据,以此来实现数据关系的描述。XML与HTML的差别在于,HTML是用来显示的,而XML是用于数据的传输网络传输数据、存储数据的。利用这一点,可以用XML来描述测试用例,然后通过串口或网口对下位机传送指令并进行测试。其XML的格式为:

```
<commands>
  <type>HELP</type>
  <type>RESET<set></set></type>
  <type>INPUT<set>SET</set></type>
  <type>INPUT<set>SET BIN</set></type>
  <type>OUTPUT<set>SET WINDOW</set></type>
  <type>OUTPUT <set>A </set><para>GANTRY HEAD
    FLOLON </para></type>
</commands>
```

#### 3.2 测试平台的设计

为将测试得到的信息显示出来,并进行下一步的自动化测试,需要搭建一个测试平台,模拟上位机给下位机发送指令。本测试平台采用宿主机与目标机交联测试的方式,程序的编译、测试分析运行在宿主主机上,程序运

行后得到的测试信息在目标机上产生,通过物理/逻辑连接传输到宿主主机上,由显示模块执行。通常物理连接为串口或局域网络,逻辑连接为串口通信协议或TCP/IP网络协议,本文研究的测试平台宿主机与目标机之间的通信采用串口连接。

#### 3.3 自动化测试的实现

将控制软件每个模块的XML脚本文件作为测试用例独立存放在一个文件夹中,然后把宿主机与目标机通过串口连接起来,并在目标机上下载好所需要测试的软件。本测试平台由用户选择需要导入的XML测试脚本,采用DOM解析编写的XML脚本,文档对象模型DOM(Document Object Model)采用与平台和语言无关的方式表示XML文档的W3C标准。通俗地说,就是通过解析XML文档,为XML文档在逻辑上建立一个树模型,树的节点是一个个对象。通过存取这些对象就能够存取XML文档的内容,然后执行向串口定时发送经过解析的测试用例,查看从串口得到的反馈,可以判断测试用例是否通过。将所有的测试用例文档(即XML文件)管理起来,有利于之后的回归测试,大大提高测试的效率。

#### 3.4 测试结果的显示

由于测试用例很多,所以随机选取50个对治疗床的控制的测试用例来进行测试,查看执行结果与执行效率,图5为测试用例的通过情况,并通过图6的饼状图



图5 测试报告

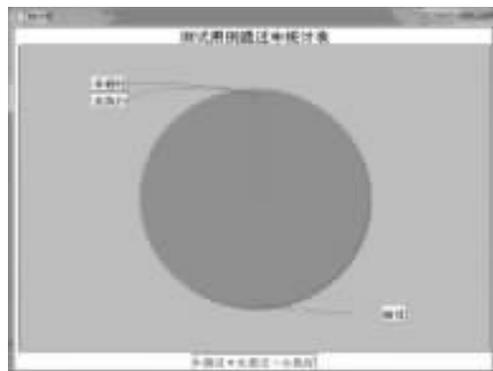


图6 测试用例通过率统计表

直观显示测试用例的通过率。

在软件开发阶段中模拟仿真得出正确的结果,可以减少下载到硬件调试的时间,大大提高效率<sup>[7]</sup>,本文中实例均在开发板上调试通过。利用 RealView MDK 的软件仿真功能可以大大提高项目的前期开发效能,帮助缩短开发周期;需要上位机发送指令控制下位机的运动时,透过医用电子直线加速器控制软件特有的性质,可以实现测试用例的自动化执行;减少人工投入,而且便于软件更新或是软件升级的回归测试,这种测试方法同样比较适合类似于医用电子的控制软件的测试。

#### 参考文献

- [1] 程敬海. 放射治疗设备[D]. 上海: 上海理工大学医疗器械学院, 2009.
- [2] ARM 公司. RealView MDK—引领 ARM 开发工具新潮流[J]. 电子产品世界, 2007(11): 125-127.
- [3] 库少平, 徐良平. RealView MDK 软件仿真器的测试技术研究[J]. 单片机与嵌入式系统应用, 2008(8): 18-21.
- [4] 李宁. ARM 开发工具 RealView MDK 使用入门[M]. 北京:

北京航空航天大学出版社, 2008.

- [5] 邵荣防, 罗克露. 嵌入式系统仿真测试平台的体系结构研究[J]. 福建电脑, 2004(7): 4-5.
- [6] 朱经纬. XML 技术在软件测试自动化中的应用[J]. 计算机工程, 2005, 31(02): 94-95.
- [7] 蔡建平. 嵌入式软件测试实用技术[M]. 北京: 清华大学出版社, 2010.

(收稿日期: 2013-02-26)

#### 作者简介:

涂施, 女, 1988 年生, 硕士研究生, 主要研究方向: 软件工程。

蔡建平, 男, 1960 年生, 教授, 主要研究方向: 嵌入式软件工程、数字媒体技术、多媒体技术。

谢会东, 男, 1980 年生, 工程师, 主要研究方向: 嵌入式软件工程。