

基于 LabVIEW 的 CRC - 16 程序分析与实现

陈公兴, 谷 刚

(广东工业大学 自动化学院, 广东 广州 510090)

摘要: 在论述循环冗余校验的原理和特点的基础上, 采用 LabVIEW 技术, 分析和研究了 CRC - 16 程序算法和程序实现的问题, 验证了它的可靠性和高效性. 说明基于图形化编程语言的 LabVIEW 技术, 可以更好更直观地用于研究高阶 CRC 软件和硬件实现的问题.

关键词: 循环冗余校验; CRC-16; LabVIEW 技术

中图分类号: TN919

文献标识码: A

文章编号: 1007-7162 (2008) 03-0073-03

循环冗余校验 (Cyclic Redundancy Check, CRC) 是一种最常用的、最有效的差错检测编码, 是基于除法与余数的原理进行错误检测的, 是对传送数据块进行检验的一种高效的差错控制方法. 在实际应用时, 发送端计算出 CRC 值并随数据一同发送给接收端; 接收端对收到的数据重新计算 CRC 并与收到的 CRC 相比较, 若两个 CRC 值不同, 则说明数据通信出现错误^[1]. 最通用的 CRC 码是 CRC - 16 (冗余块校验序列). CRC - 16 块校验字符是二进制除法过程的余数. 数据消息多项式 $G(x)$ 将除以唯一的生成器多项式函数 $P(x)$, 商被丢弃, 余数被截短为 16 位, 然后作为 BCC 附加到消息上. 生成器多项式必须是一个质数, CRC - 16 将检测所有单位错误、除数至少包含 3 个逻辑 1 时的所有双位错误、除数包含因数 11 时的所有奇数数量的位错误、所有 16 位或不到 16 位的错误串以及 99.9% 的多于 16 位的错误串^[2]. CRC 校验具有典型特征: 编码简单、误判概率低、检错能力强、数学结构严密; 允许高阶纠错码的设计; 易于用编码器及检测电路实现. 且它在串行通信和前向差错控制 (FEC) 系统中得到广泛应用^[3]. 而 LabVIEW 图形化编程显得更直观、简单和易学. 因此, 基于 LabVIEW 技术的 CRC 研究和开发在测控和通信领域具有重要的意义. 目前, 已经利用 FPGA、DSP 和单片机技术研究 CRC 程序实现和硬件设计.

1 循环冗余校验原理

1.1 循环码的构成

循环码叫做 CRC 码, 是一种典型的分组线性

码, 是一种高效能的检错和纠错码, 码组的构成如图 1 所示, 由 k 个信息码元和 $(n - k)$ 个检验码元构成.

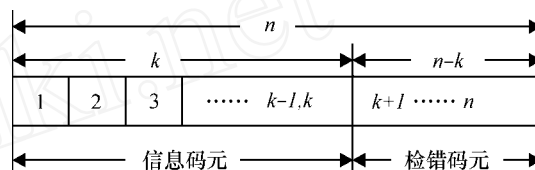


图 1 循环码组的构成

1.2 循环冗余校验的原理

通常循环冗余校验方法的基本原理如下:

- 1) 设要发送的数据对应的多项式为 $p(x)$;
- 2) 发送方和接收方约定一个生成多项式 $G(x)$, 设该生成多项式的最高次幂为 r ;
- 3) 在数据块的末尾添加 r 个零, 则其相对应的多项式为 $M(x) = p(x) * x^r$;
- 4) 用 $M(x)$ 除以 $G(x)$, 获得商 $Q(x)$ 和余式 $R(x)$, 则 $M(x) = Q(x) * G(x) + R(x)$;
- 5) 令 $T(x) = M(x) + R(x)$, 采用模 2 运算 (无进借位的二进制运算, 也相当异或运算), $T(x)$ 所对应的数据是在原数据块的末尾加上余式所对应的数据得到的;
- 6) 发送 $T(x)$ 所对应数据;
- 7) 设接收端接收到的数据对应的多项式为 $N(x)$, 将 $N(x)$ 除以 $G(x)$, 若余式为零, 则表明传输的数据帧正确. 否则该数据通信出现错误. 即若 $N(x) = T(x)$, 则

$$\frac{N(x)}{G(x)} = \frac{Q(x) * G(x) + R(x) + R(x)}{G(x)} =$$

收稿日期: 2008-01-09

作者简介: 陈公兴 (1981-), 男, 硕士研究生, 主要研究方向为虚拟仪器技术.

$$\frac{Q(x) * G(x)}{G(x)} = Q(x) \text{ (模 2 运算)}$$

余式为零. 若 $N(x)$ 不等于 $T(x)$, 余式不为零.

2 CRC - 16 程序分析及 LabV IEW 仿真实现

2.1 CRC - 16 生成电路^[5]

CRC 码中位的数量等于生成多项式的最高指数. 指数标识包括逻辑 1 的生成多项式中的位位置.

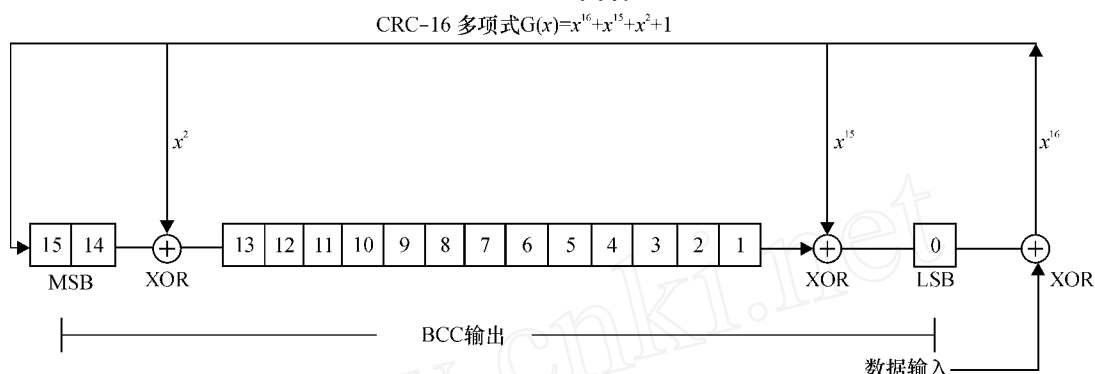


图 2 CRC - 16 生成电路

2.2 16 位 CRC 的程序算法、程序框图及其 LabV IEW 编程语言.

其算法如下:

- 1) 设置一个 16 位寄存器为全 1, 称之为 CRC 寄存器;
- 2) 把数据帧中的第一个字节的 8 位与 CRC 寄存器中的低字节进行异或运算, 结果存回 CRC 寄存器;
- 3) 将 CRC 寄存器向右移一位, 最高位填充 0, 最低位移出并检测;
- 4) 如果最低位为 0: 重复第 3) 步 (下一次移位); 如果最低位为 1: 将 CRC 寄存器与一个预设的固定值 (0xA001H) 进行异或运算;
- 5) 重复第 3) 步和第 4) 步直到 8 次移位, 这样处理完了一个完整的 8 位;
- 6) 重复第 2) 步至第 5) 步来处理下一个 8 位, 直到所有的字节处理结束;
- 7) 最终 CRC 寄存器的值就是 CRC 的值.

其程序框图如图 3 所示.

其对应的 LabV IEW 编程语言程序:

因此, 对于 CRC - 16, 其标准的生成器多项式 $G(x)$ 等于 $x^{16} + x^{15} + x^2 + x^0$. 图 2 给出了生成 CRC - 16 的块特性校验 (Block Character Check, BCC) 电路方块图. CRC 生成电路需要给 BCC 中的每个位一个寄存器. 图中有 16 个移位寄存器. 另外除了 x^0 之外, 对于包含逻辑 1 的生成多项式的每个位位置, 都在移位寄存器的输出处放置了一个 XOR 与或门. BCC 是整条消息通过 CRC 生成电路之后 16 个寄存器的内容.

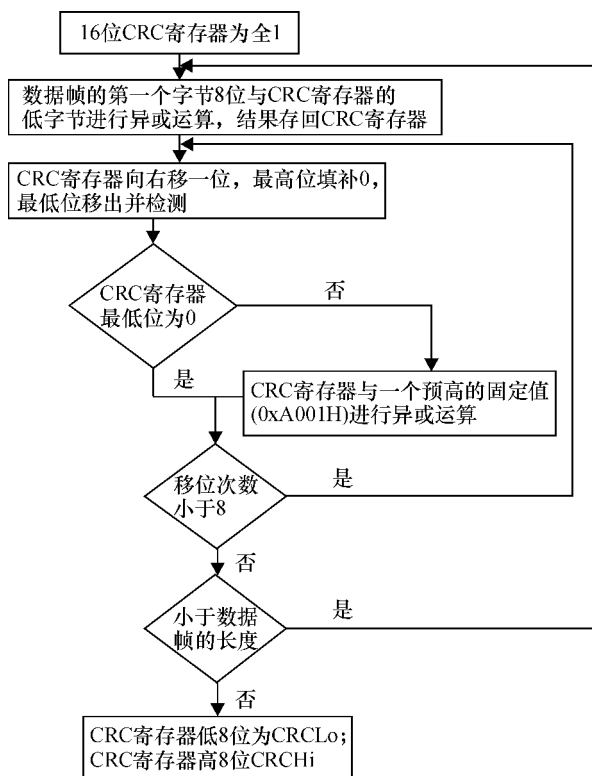


图 3 CRC - 16 算法的流程框图

```
uInt16 i, j, tmp, CRC;
uInt16 buf[ i];
uInt8 CRCHi, CRCLo;
CRC = 0xFFFF;
for(i=0; i < length; i + +)
{
    CRC = buf[ i]^CRC;
    for(j=0; j < 8; j + +)
    {
        tmp = CRC&0x0001;
        CRC = CRC > > 1;
        if( tmp)
            CRC = CRC^0xA001;
    }
}
CRCLo = CRC&0xFF;
CRCHi = CRC > > 8;
```

2 3 LabVIEW 程序仿真实现

图 4为 LabVIEW 算法仿真实现的方框图.

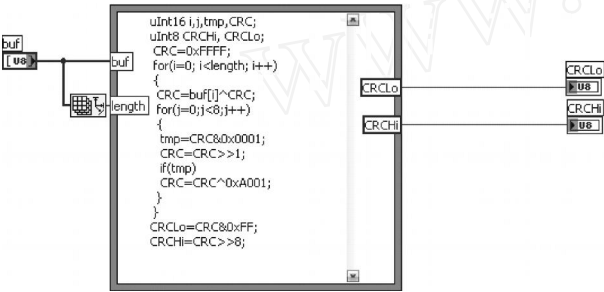


图 4 基于 LabVIEW 的 CRC - 16 算法实现的程序框图

3 结束语

基于 LabVIEW 技术 (虚拟仪器技术)能够简单地、形象地编写 CRC - 16 程序及其实现.因此,一个 16 位的 CRC 校验码生成程序能被广泛地应用于数据传输校验,因为它能高效率地检测数据传输的正确性.目前,磁盘驱动器的读写已经采用了 CRC - 16,通用的图像存储格式 GIF、TIFF 等也都采用 CRC 作为检错手段.进一步采用 LabVIEW 技术,可以推导和分析高阶 CRC 程序和硬件实现的问题.

参考文献:

[1] 全国高校嵌入式系统教学研讨会编委会,中国电子学会嵌入式系统专家委员会联合主编. 2006 年第四届全国高校嵌入式系统教学研讨会、第一届全国嵌入式系统学术交流会议论文集 [C]. 北京:北京航空航天大学出版社, 2006

[2] Wayne Tomasi 数据通信与联网技术 [M]. 北京:清华大学出版社, 2006

[3] 李朝青. PC 机与单片机 & DSP 数据通信技术选编 [G]. 北京:张宝生,孙岩等,译.北京:北京航空航天大学出版社, 2004

[4] 于锋. 计算机网络与数据通信 [M]. 北京:中国水利水电出版社, 2003: 38-39.

[5] 戴鹏飞,王胜开,王格芳,等. 测试工程与 LabVIEW 应用 [M]. 北京:电子工业出版社, 2006

Programme Analysis and Realization of CRC-16 Based on LabVIEW

Chen Gong-xing, Gu Gang

(Faculty of Automation, Guangdong University of Technology, Guangzhou, Guangdong 510090, China)

Abstract: Based on the principles and the characteristics of Cyclic Redundancy Check, the problems with CRC-16's programme algorithm and programme realization have been analysed, and its reliability and validity have been testified by using LabVIEW technology. It shows that the problems of the software and hardware realization of high-order CRC based on LabVIEW technology, in figure and programme language can be coped with well and intuitively

Key words: Cyclic Redundancy Check; CRC-16; LabVIEW technology