

二进制 BCH 编译码的 DSP 实现

阎渊海,施洪生,亢 凯,胡泽民

(北京交通大学 电气工程学院,北京 100044)

摘 要: 针对 DSP 硬件之间的无线通信,提出了一种基于 DSP 的二进制 BCH(15,7)编译码器.设计采用 TI 公司的 32 位浮点高性能 DSP 处理器,通过 CCS 软件平台,分别编写了相应的编码、译码及纠错程序.文中介绍了二进制 BCH(15,7)码及其纠错码的算法,并且给出了相应的 C 语言程序.通过 DSP 与 PC 之间的串口通信,验证了 BCH 编译码和纠错功能.当 DSP 接收数据出现小于等于两位随机错误时,可以发现错误并给与纠正.同 FPGA 相比,在保证数据传输的准确性的同时,降低了硬件成本,简化了算法实现难度.

关键词: DSP; BCH 码; 纠错; 编译码

中图分类号: TN911

文献标识码: A

文章编号: 1000-7180(2018)08-0064-04

DSP Implementation of Binary BCH Encoding and Decoding

YAN Yuan-hai, SHI Hong-sheng, KANG Kai, HU Ze-min

(School of Electrical Engineering, Beijing Jiaotong University, Beijing 100044, China)

Abstract: Aiming at the wireless communication between DSP hardware, a binary BCH (15,7) codec based on DSP is proposed. Designed by TI's 32-bit floating point high-performance DSP processor, through the CCS software platform, respectively, prepared the corresponding coding, decoding and error correction procedures. In this paper, the algorithm of binary BCH (15,7) code and its error correction code is introduced, and the corresponding C language program is given. Through the serial communication between DSP and PC, BCH coding and decoding and error correction function are verified. When the DSP receive data is less than or equal to two random errors, you can find the error and give correction. Compared with the FPGA, while ensuring the accuracy of data transmission at the same time, reducing the cost of hardware, simplifying the difficulty of the algorithm.

Key words: DSP; BCH codec; error correction; encoding and decoding

1 引言

近年来,数字信号处理器(Digital Signal Processing, DSP)得到广泛应用, TI 公司的 TMS320F28335 在价格、性能、功耗等方面与同类产品相比更为出色^[1]. 对于 DSP 处理器之间的通信,一般采用传统的有线连接,例如 RS485、RS232、CAN 等方式. 可当 DSP 处理器距离较远、或者两者之间有相对运动时,传统有线连接十分不方便,无线通信的出现解决了这一难题. 可无线通信的大力推广,虽然使数据通信变得灵活、简便,但存在着信道易受干扰,误码率高等问题.

为此,本文设计了二进制 BCH(15,7)编译纠错码的 DSP 实现,用来提高数据传输的可靠性. BCH 码是循环码的一个重要子类,是一种常用的线性分组码,纠错能力较强,结合 DSP 的数据处理能力,这里采用 BCH(15,7)进行数据传输,15 是码长,7 是有效数据位^[2]. 例如对 TMS320F28335 的 AD 采集数据进行传输时,由于其具有 12 位的 AD 数据采集精度,可将一次采集数据分为两次传输. 由于编译码算法复杂度较低,传输延时不会太大,能保证正常数据通信. 实验验证中,让 PC 向 DSP 传输数据, DSP 能完成数据编码、译码及纠正 ≤ 2 位错误数据的能力. 纠错编码在 DSP 上的应用,使 DSP 数据传输更

加稳定,扩展了 DSP 的使用范围,同时同 FPGA 相比,DSP 价格更实惠,性价比更高.

2 二进制 BCH(15,7)编译纠错码

BCH 码是一类最重要的循环码,能纠正多个随机错误,它是 1959 年由 Bose、Chaudhuri 及 Hocquenghem 各自独立发现的二元线性循环码,人们用他们的名字字头命名该码^[3].在短中码长情况下,BCH 码的纠错性能很强.二进制 BCH 码是指在数据通信中,传输的数据只有 0 和 1,同多进制 BCH 码相比,二进制 BCH 码译码在纠错时只需要将错误位翻转即可,而多进制码则需要判断错误程度,然后再依据错误程度修改错误.

2.1 二进制 BCH(15,7)编码

BCH(15,7)的编码输出格式如表 1,这里为了满足 7 位有效位的格式,统一将第 0 位设置成 1.由表可知,要想得到输出编码,前提是得到 BCH 校验码,即 BCH 生成多项式. BCH 生成多项式如式(1)所示,

$$g(\chi) = \text{LCM}[m_1(\chi), m_2(\chi) \cdots m_{\lfloor \frac{15}{2} \rfloor}(\chi)]$$

(1)

式中,LCM 指最小公倍式; t 表示纠错个数; $m_i(\chi)$ 为最小多项式;由该式可得到生成矩阵 G .以 BCH(15,7)为例,其生成多项式为式(2),转换成二进制码为 111010001.

$$g(\chi) = m_1(\chi) \cdot m_3(\chi) = \chi^8 + \chi^7 + \chi^6 + \chi^4 + 1$$

(2)

将 7 位数据位后补 8 位 0,之后与生成多项式的二进制码进行模 2 除法运算,即可得到 BCH(15,7)的校验码^[4].

表 1 BCH(15,7)编码格式

0 位	1~6 位	7~14 位
1	数据	BCH 校验码

2.2 二进制 BCH(15,7)译码及纠错

BCH 码的译码相对于编码要复杂一些,可以分为以下三步:(1)由接收到的数据 r 计算错误图样 s ;(2)由错误图样 s 计算错误序列 e ;(3)通过错误序列 e 找到数据 r 的错误位并改正,二进制 BCH 相当于把该位取反^[5-6].常用的译码算法有彼得森算法(Peterson)、伯利坎普算法(Berlekamp)、欧几里得算法(Euclidean)等^[7].由于本文设计的 BCH 码长较短,故采用直接由错误图样 s 得到错误位置,减少算法的复杂程度.

在编写 BCH 译码算法之前,需得到 BCH(15,7)的监督矩阵 H ,其格式为 $\{Q^T, I_8\}$. Q 由生成矩阵 G 得到, I_8 为 8 阶单位矩阵.由此可得到 BCH(15,7)的监督矩阵为

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

之后,将接收到的数据 r 与 H 矩阵的转置矩阵做乘法运算,即可得到错误图样 s ,这时,如果 s 中元素全为 0,则数据传输无误,否则,通过 s 与 H 矩阵的列对比,可得出错误位置,将该位翻转即可.

3 BCH(15,7)码的 DSP 实现

3.1 DSP 编码实现

DSP 编码主要实现的功能包括:数据提取、BCH 校验码生成、数据打包成帧.这里假设 DSP 要提取的是 16 位数据中前 7 位,且首项为 1,则其具体流程图如图 1.

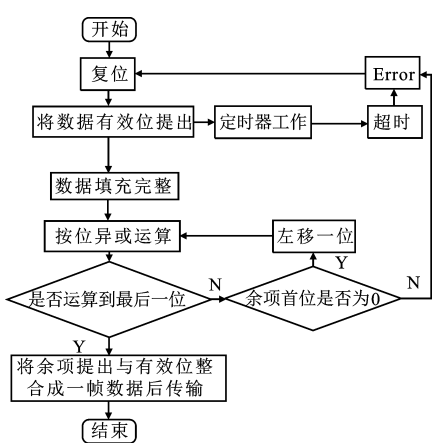


图 1 BCH(15,7)编码流程图

程序运行开始后先进行 DSP 复位,接着有效位提出,定时器开始工作,目的是防止数据出错后陷入死循环.全部按位异或完的余项即为 BCH 校验码,余项的首次产生是由数据与生成多项式码按位异或得到,之后每次余项与生成多项式码参与运算^[8].其 CCS 编程如下:

```
for(i=0;i<BCH_LENGTH_GENERATPLOY;i++)
{
    cverchar[i]=ReceivedChar[i]*GeneraPloy[a++];
}
```

```
} //数据与生成多项式码首次产生余项
for(i=0;i<BCH_LENGTH_GENERATPLOY;i++)
{
    cverchar[i]=cverchar[i]^GeneraPlay[i];
}
```

将程序烧写入 DSP 后,通过串口转 USB 把 DSP 与 PC 连接起来,PC 向 DSP 发送数据,DSP 对数据进行 BCH 编码后再返给 PC,并在串口调试助手上显示^[9]. 假定 PC 向 DSP 发送的数据为 111100001111000010,程序识别的是前七位数据位,即 11110000,运行程序后,串口调试助手显示如图 2, 10100110 即为程序自动生成的 BCH 校验码.



图 2 PC 显示 DSP 返回帧

3.2 DSP 译码纠错实现

相比于译码程序,纠错程序相对简单,只需将错误位取反即可, w 为错误位,CCS 程序如下:

```
void RevBit( int w)
{
    sendchar[w]^=1;
}

//在进行译码之前,需要将错误图像 s 求出来,即数据 r 与的 H 转置矩阵相乘. 结果为一行八列的矩阵. CCS 中程序如下:
int r; //矩阵的列索引
int c; //矩阵的行索引
sendchar[1]=0;
for(r=0;r<LENGTH-8;r++)
{
    for(c=0;c<LENGTH-1;c++)
        check[r]^=(sendchar[c]&h[c][r]);
}
for(c=0;c<LENGTH-8;c++)
    if(check[c]!=0)
```

return FALSE;
return TRUE;
这里的 FALSE 和 TRUE 作为返回值,分别表示有错误发生和没有错误发生的情况.

有错误发生可以有三钟情况,第一是当接收到的数据只有一位错误时;第二是当接收到的数据发生两位错误时;最后是当接收到的数据错误位数大于两位时.

3.2.1 有一位错误发生

当只有一位数据发生错误时,只需将错误图样 s 与 H 的转置矩阵一一对比,便可找到错误. 如第三行与 s 完全相同,则第三位为错误位. CCS 程序如下:

```
for(c=0;c<LENGTH-1;c++)
{
    b=0;
    for(r=0;r<LENGTH-8;r++)
    {
        if(h[c][r]==check[r])
            b++;
        if(b==8)
        {
            RevBit(c);
            break;
        }
    }
```

b 为标志位,当 b 等于 8 时,表示某一行元素与错误图样 s 完全一样. 在 PC 与 DSP 通信中添加译码程序,PC 给 DSP 发送 BCH 编码后的数据 1111000010100110,进行 BCH 译码纠错,PC 显示 BCH 译码纠错后的数据. 这里我们人为给数据添加错误,将数据第二位由 1 变为 0,PC 上返回如图 3.



图 3 一位数据纠错后返回帧

3.2.2 有两位错误发生

当有两位错误发生时,这时候错误图样不能直接和 H 的转置矩阵对比,而应该把 H 转置矩阵行与行之间做异或运算,之后再与 s 做对比. 如第三行与第五行异或完后,与 s 对比完全一样,则接收数据的第三位和第五位出错. CCS 程序如下:

```
for(r=c+1;r<LENGTH-1;r++)
```

```
{d=0;
for(e=0;e<LENGTH-8;e++)
{if(check[e] == (h[c][e]^h[r][e]))
{d++;}
if(d==8)
{RevBit(c);
RevBit(r);}}}
```

在程序中人为加入错误,把数据 111000010100110 的第三位和第十一位取反,造成 2 位错误数据,PC 向 DSP 传输数据为 110000010110110,显示返回数据帧如图 4 所示。

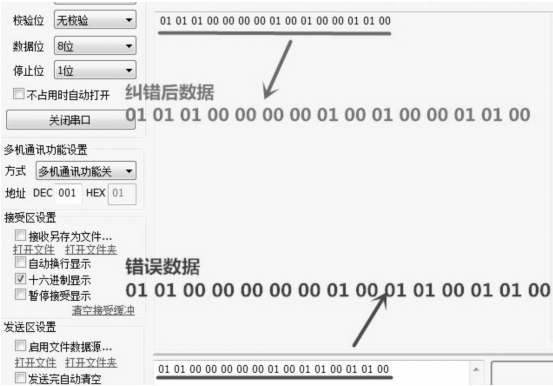


图 4 两位数据纠错后返回帧

3.2.3 有大于两位错误发生

这时 BCH(15,7)已经无法对错误数据进行完整修改,为了保证数据传输正确性,数据只能重新发送。

综上三种情况,BCH(15,7)的译码结构流程图如图 5 所示。

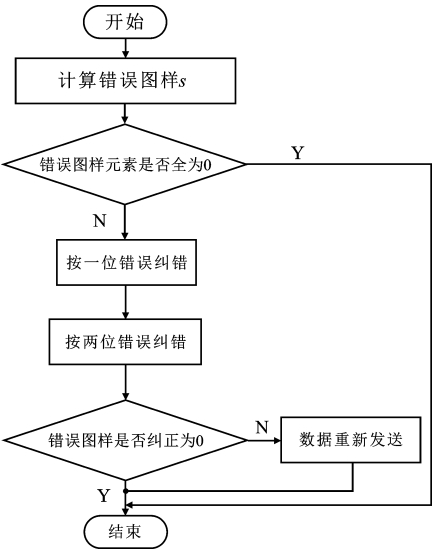


图 5 BCH(15,7)译码纠错流程图

4 结束语

(1)通过 DSP 实现 BCH(15,7)编译纠错码算法,在保证处理速度、准确度的同时,还可充分利用其外设资源,增加功能任务和算法难度。

(2)BCH(15,7)码的码长较短,采用该文中的译码方法,可减少算法难度,提高查找效率,并且在码长较短的情况下,其实际纠错能力接近理论值^[10]。

(3)结合 DSP 的使用,算法只满足了(15,7)一种编译纠错码情况,应用范围受限,之后可采用复杂程度更高的 BCH 编译码,满足多种需求。

(4)在保证数据传输正确性时,采用纠错算法对数据进行修正,是一种比较被动的做法,当与降低噪声、屏蔽干扰等方法结合使用时,效果会更好。

参考文献:

[1] 张卿杰,徐友,左楠,等.手把手教你学 DSP[M].北京:北京航空航天大学出版社,2015.

[2] 傅祖芸,赵建中.信息论与编码[M].2版.北京:电子工业出版社,2014.

[3] 刘旭.基于(15,5,7)BCH 码的信息协调协议[D].重庆:西南大学,2015.

[4] 赵淑明,王卫东,李彦强.BCH 编译码的 DSP 实现[J].山西电子技术,2008,35(2):25-26.

[5] 李璐,周海燕.一种含 BCH 编解码器的 SLC/MLC NAND FLASH 控制器的 VLSI 设计[J].现代电子技术,2009,32(7):167-170.

[6] 宁楠,鲍慧,宋文妙.一种基于 FPGA 的纠错编译器的设计与实现[J].通信技术,2008,41(8):95-100.

[7] 张彦,李署坚,崔金.一种 BCH 码编译码器的设计与实现[J].通信技术,2010,43(12):24-26.

[8] 张雪梅,郭春波.BCH(15,7,5)纠错译码码的 MATLAB 实现[J].微型电脑应用,2004,20(7):44-46.

[9] 李冰.基于 DSP 的 GSM 基站信息监测设备设计与实现[D].成都:电子科技大学,2015.

[10] 余绍迪,赵利,刘小康.应用于遥控命令提取的快速 BCH 译码方法[J].电视技术,2017,41(4/5):177-181.

作者简介:

阎渊海 男,(1992-),硕士研究生.研究方向为无线数据测试及传输. E-mail:16121565@bjtu.edu.cn.

施洪生 男,(1975-),博士,副教授.研究方向为风机齿轮箱测试.

亢凯 男,(1994-),硕士研究生.研究方向为风机磨粒油液检测.