

高速数模转换器 AD9779/AD9788 的应用

卫晓娟, 李军红

(凯迈(洛阳)电子有限公司, 河南 洛阳 471000)

摘要: 介绍了高速数模转换器 AD9779 和 AD9788 的使用方法, 重点介绍了其驱动软件设计方法和内部寄存器的配置方式等。文中内容都是实践经验, 具有很强的实用性和参考价值。

关键词: AD9779; AD9788; 数模转换器; 串行外设接口; 锁相环

中图分类号: TP21

文献标识码: B

文章编号: 1674-7720(2011)24-0079-02

The application of high-speed DACs: AD9779 and AD9788

Wei Xiaojuan, Li Junhong

(CAMA(Luoyang) Electronics Co.Ltd, Luoyang 471000, China)

Abstract: This article briefly introduces the features and application of two types of high-speed DACs: AD9779 and AD9788. It includes the design idea of software driver and the configuration for the internal registers. The main content is very practical because of coming from practice.

Key words: AD9779; AD9788; DAC; SPI; PLL

AD9779 是 ADI 公司的一款产品, 是双通道、宽动态范围数模转换器(DAC), 提供 1 GS/s 采样速率, 允许高至奈奎斯特频率的多载波生成^[1]。AD9788 提供 800 MS/s 的采样速率, 其性能和应用方式与 AD9779 非常相似^[2]。AD9779 和 AD9788 虽然不是一个系列, 但芯片管脚是兼容的, 在硬件电路的设计上并没有区别, 两者均使用标准的 SPI 接口来驱动, 因此软件实现上也可以采用统一的驱动方式。但在寄存器的配置上, 两者存在差异, 这在后面会具体介绍。为了描述方便, 如不特殊说明, 下文中使用 AD97xx 来统一表示 AD9779 和 AD9788。

1 AD97xx 的 SPI 规范

AD97xx 采用标准的 4 线 SPI 接口进行通信, 当然也可以配置成 3 线 SPI 接口。

一个寄存器的读写周期(或叫传输周期)包括两部分: 指令部分和数据部分。指令部分是一个 8 bit 的字节, 对应于 8 个时钟的上升沿, 用来控制读写以及随后需要读写的数据字节数。D7 为读/写标志位(其中读为高, 写为低), D6 和 D5 是用来配置指令字节后面需要传输的数据的字节数, 通常配置如表 1 所示(N1 和 N0 对应指令字节中的 D6 和 D5), D4~D0 是 AD97xx 寄存器的地址。

需要特别说明的是, 对于 AD9788, 指令字节的 D6 和

D5 两个位被屏蔽了, 因此这两位不需要配置, 指令字节后面需要传输的数据的字节数是由寄存器本身来决定的。例如对于一个 32 bit 的寄存器, 指令字节后面需要传输 4 B 的数据。

2 AD97xx 的驱动方式

AD97xx 主要通过 SPI 驱动, 实现寄存器的读写和功能的配置。

将 AD97xx 的代码分为两部分设计, 一部分是底层的 SPI 接口驱动, 完成 SPI 接口的初始化、读写以及设备封装等功能; 另一部分是 AD97xx 本身的驱动, 主要通过 SPI 封装的读写接口实现对 AD97xx 芯片的控制。

3 AD97xx 的配置说明

3.1 寄存器长度不固定的问题

AD9779 的寄存器长度是固定的 8 bit, 但 AD9788 的寄存器长度是不固定的, 有 8 bit、16 bit、32 bit, 还有 24 bit。datasheet 中规定一个寄存器的读写传输周期由寄存器本身决定, 包括一个指令字节和实际传输的数据字节数, 例如, 0x00 寄存器传输周期为 2 B, 0x01 寄存器传输周

表 1 指令字节后面传输的数据的字节数

N1	N0	描述
0	0	传送 1 B
0	1	传送 3 B
1	0	传送 2 B
1	1	传送 4 B

期为3 B。因此对于AD9788,不能用统一的读写传输周期,例如5 B(8 bit指令+32 bit数据),而是需要对不同长度的寄存器分开处理,这就意味着需要模拟四种不同的SPI时序。

3.2 个别寄存器回读时应注意

在写寄存器操作时,一般需要回读一下来判断写的值是否正确。回读时要注意以下两种情况,否则会判断错误:

(1)有的寄存器个别位是只读的,与写进去的值没有关系,回读时最好屏蔽掉相关位。例如AD9788的0x04寄存器的高三位(23 bit~21 bit)为只读;AD9779也有类似的寄存器,例如0x00寄存器低三位(2 bit~0 bit)也是只读;

(2)寄存器在配置完后就发生了变化,这种寄存器一般是指寄存器,例如AD9788的0x09寄存器,有一个锁定指示位,配置完后,若PLL锁定,这一位会改变,读出的值和写的值就会不一样。

3.3 AD97xx PLL的配置

PLL的配置是AD97xx配置中的一个重点,AD9779与AD9788的PLL结构是相同的,不同的是AD9788比AD9779多了一个NCO,用于频点搬移。本节以AD9788为例讲述PLL相关寄存器的配置。

在AD9788的数据手册中,共有15个配置寄存器,其中与PLL相关的寄存器主要有0x01、0x04两个寄存器。不过由于要进行频点搬移,所以还要对0x0a寄存器按照系统需要进行配置。这样一来,只需关注0x01、0x04和0x0a这三个配置寄存器就可以了。

3.3.1 DCTL寄存器

DCTL寄存器(DIGITAL CONTROL REGISTER)参数为:地址:0x01h;宽度:2 B;默认值:0x3100;定义值:0x31c0。在此寄存器中,主要关注bit[7:6]的设置,如果系统要求DAC实现8倍内插,就要配置bit[7:6]=11,其他位按默认值即可。

3.3.2 PLLCTL寄存器

PLLCTL寄存器(PLL CONTROL REGISTER)参数为:地址:0x04h;宽度:3 B;默认值:0x3837cf;定义值:0x2fb387。

在对PLL的CTL寄存器进行设置时,主要关心bit[15:11]和bit[7:2]值的设置,bit[15]默认是PLL无效,所以要配置bit[15]=1;bit[14:13]和bit[12:11]要根据具体系统需求配置,本例中配置为bit[14:13]=01,bit[12:11]=10,bit[7:2]的值通过式(1)计算并查表2后决定。

$$f_{vco} = f_{refclk} \times N_1 \times N_2 \quad (1)$$

经过查表2可知,bit[7:2]可以取100001或者100000,为了使1474.56 MHz尽量接近于PLL Lock选择范围的中间位置,所以选用bit[7:2]=100001。对于其他位,按器件手册上推荐的最佳值选取即可,这里要注意bit[23:21]的只读属性。

表2 AD9788 PLL部分基带选择表

温度为-40℃~+85℃时,锁相环工作区间		
锁相环波段选择	VCO频率范围/MHz	
	最低频率	最高频率
100011(35)	1480	1553
100010(34)	1475	1529
100001(33)	1439	1505
100000(32)	1435	1489

3.3.3 FTW寄存器

FTW寄存器(FREQUENCY TUNING WORD REGISTER)参数为:地址:0x0ah;宽度:4 B;默认值:0x00000000;定义值:0x3078E38E。

载波信号的频率由FTW寄存器(0x0a)来配置,计算方法如下:

$$f_{CENTER} = (FTW)(f_{DACCLK})/2^{32} \quad 0 \leq FTW \leq 2^{31} \quad (2)$$

$$f_{CENTER} = f_{DACCLK} \times [1 - (FTW/2^{32})] \quad 2^{31} \leq FTW \leq 2^{31} - 1 \quad (3)$$

其中, f_{CENTER} 为NCO搬移到的中心频率(例如139.6 MHz), f_{DACCLK} 为DAC的采样时钟,FTW为FTW寄存器的值。 f_{DACCLK} 的计算公式如下:

$$f_{DACCLK} = f_{REFCLK} \times N_2 \quad (4)$$

$$FTW = f_{CENTER} \times 2^{32} / f_{DACCLK} \quad (5)$$

通过以上的原理和公式,就可求出特定中心频率下的FTW寄存器的值。例如 $f_{CENTER} = 139.6$ MHz, $f_{REFCLK} = 92.16$ MHz, $N_1 = 2$, $N_2 = 8$ 时,则 $f_{DACCLK} = 92.16$ MHz $\times 8 = 737.28$ MHz。

由式(5)即可求出FTW寄存器的值,如下:

$$FTW = 139.6 \text{ MHz} \times 2^{32} / 737.28 \text{ MHz} \approx 813228942 \\ = 0X3078E38E$$

驱动工作完成以后,可以通过一些简单的测试方法来验证AD97xx芯片是否正常工作。当确认芯片和驱动程序能够正常工作后,驱动程序就可以投入使用了。

参考文献

- [1] Analog Devices Inc. AD9776/AD9778/AD9779 datasheet[S]. 2007.
- [2] Analog Devices Inc. AD9785/AD9787/AD9788 datasheet[S]. 2007.

(收稿日期:2011-07-20)

作者简介:

卫晓娟,女,1984年生,学士,助理工程师,主要研究方向:电子工程。

李军红,女,1981年生,学士,助理工程师,主要研究方向:电子工程。