

**摘要** 在 Matlab/Simulink 环境下, 用图形化的方式设计 DSP 程序, 可简化程序的设计。利用 Embedded Target for T1 C2000 DSP 工具包, 设计 DSP 的 ADC 转换程序; 利用 Simulink 的数字信号处理工具包, 设计 FIR 滤波器进行滤波处理; 给出在修改生成的 C 语言程序时如何使 DSP 能正确运行。设计的程序在 TMS320LF2407A 处理器上运行正确。

**关键词** 数字信号处理 Matlab Embedded Target for TI C2000 工具箱 FIR 滤波器

TMS320LF2407 是 TI 公司主推的一种高性能、低价格 DSP 处理器, 其处理速度达到 30 MIPS, 片内处理集成 RAM、Flash 及定时器外, 还集成了 A/D 转换器、PWM 控制器及 CAN 总线控制器等模块, 特别适合于电机、电源变换等实时要求高的控制系统。但是通常设计 DSP 程序的方法是, 在 DSP 的集成开发环境 CCS 中用 C 语言设计, 需要花费大量的时间用来编写和输入程序代码。在 Matlab 中用图形化的方式设计 DSP 的程序, 能够缩短产品的开发时间。

### 1 Embedded Target for T1 C2000 DSP 介绍

目前, 新版本的 Matlab 软件(Matlab7.0)已经集成了 TI 公司 C2000、C5000、C6000 系列 DSP 的开发工具包, 可在 Matlab/Simulink 环境中用图形化的方式进行 DSP 的设计及仿真验证。并能将设计的图形文件(.mdl)直接转换成 C 语言程序。其中 C2000 系列的开发工具是 Embedded Target for TI C2000 DSP。该工具包是 TI 公司与 MathWorks 公司共同开发的产品, 在 Matlab/Simulink 中嵌入了 eXpressDSP 工具箱, 支持 C24x 及 C28x 系列的 DSP 处理器。在 C24x 系列 DSP 工具箱中, 包含 DSP 处理器中的模/数转换(ADC)、CAN 发送及接收、PWM 控制等模块。用户可以在 Matlab 中调用这些图形化的功能模块及 Simulink 中的其他模块建立数字信号处理的模型, 并可以对模型进行仿真验证, 然后生成 TMS320C2000 的 C 语言代码及 CCS 的工程项目文件, 在 CCS 中经修改、编译后就可以下载到 DSP 目标板中运行。

### 2 ADC 转换及 FIR 滤波处理程序的设计

以下是用 Embedded target for TI C2000 工具包设计 ADC 转换及 FIR 滤波的步骤。

步骤 1, 在新建的 Simulink 文件(.mdl)中, 放入 C2000 Target Preferences 中的 LF2407 eZdsp 功能块, 用于参数的初始化设置。对话框设置如图 1 所示。其中 DSP 定时器的时钟比例因子(Timer Clock Prescaler), 可以选择 1~128, 则相应的定时器采样时间为:

$$\text{Base Rate Sample Time} = \frac{\text{Timer Period}}{\frac{\text{CPU Clock Speed}}{\text{Timer Clock Prescaler}}}$$

式中 Timer Period 是 DSP 的最大时钟计数周期, LF2407 是 16 位定点处理器, 所以 Timer Period 数是  $2^{16}-1$ 。图 1 中设定的 Timer Clock Prescaler 数值是 2, 当 LF2407 的工作频率(CPU Clock Speed)为 40MHz 时, 由上式计算出的定时器的采样时间是 0.003 2 S。由于数据处理需要占用一定的运行时间, 所以要通过试验选择适当的定时器采样时间。

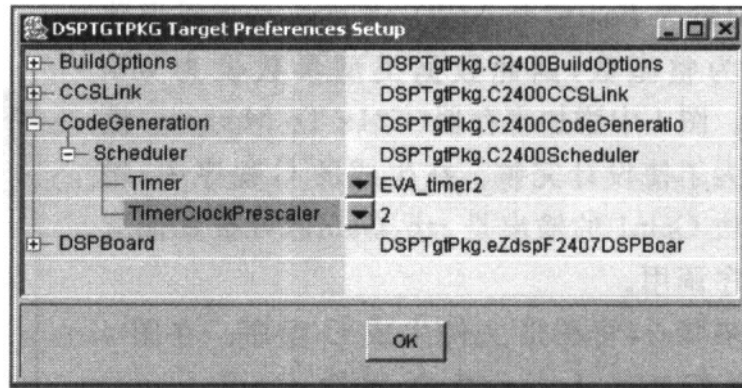


图 1 LF2407 DSP 参数预设置对话框

实现 A/D 转换的功能块是 C24x ADC，其参数设置如图 2 所示。A/D 转换通道可以选择模块 A、B 中的任一个通道，也可以选择多个通道，A/D 转换的采样时间设置为 64/80 000。

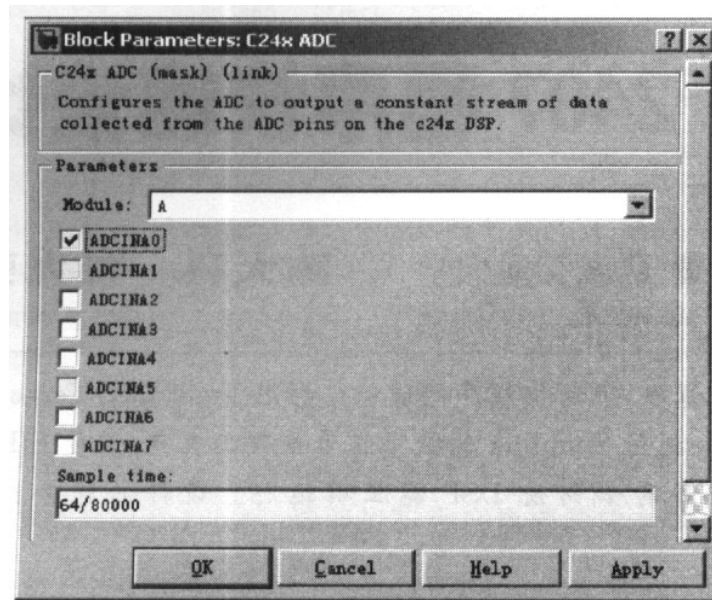


图 2 ADC 参数设置对话框

步骤 2，设计 FIR 滤波器。在 Simulink 的信号处理工具箱(Signal Processing Bloeks et)中，将滤波器设计专用工具(FDATool)放入文件中，双击图符，弹出图 3 所示的滤波器设计对话框。

在图 3 中，选择滤波器类型为 FIR 低通滤波器，采样频率为 6kHz，低通频率为 1kHz，截至频率为 2 kHz。先点击对话框中的 Design Filter，然后再点击图 3 中实现模型(Realize Model)图标进入模型实现对话框，选择 Over-write generated“filter”block，则在设计框图中生成一个名称为“Filter”的 FIR 滤波器的功能框图，再将原先放入的 FDATool 图标删除，将 Filter 连接到图中，完成的设计如图 4 所示。

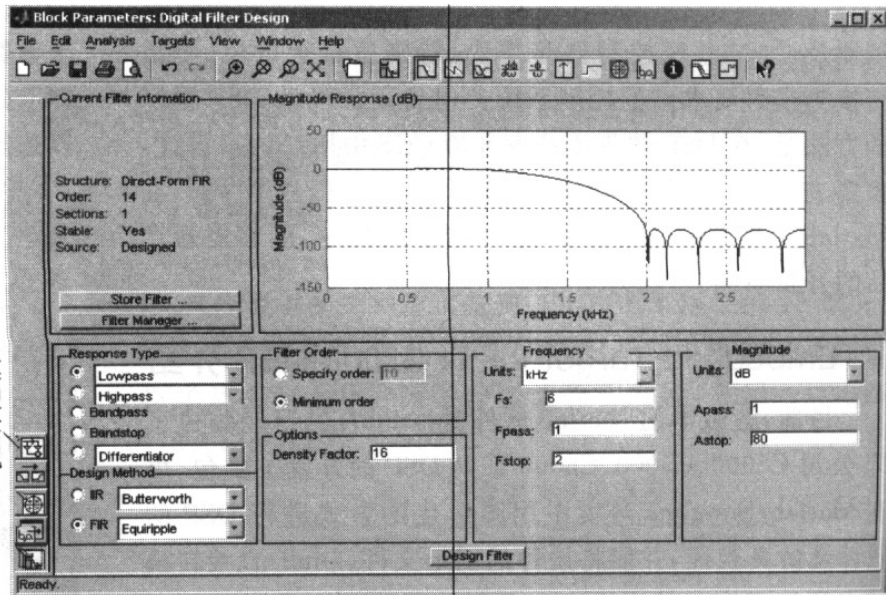


图3 数字滤波器设计对话框

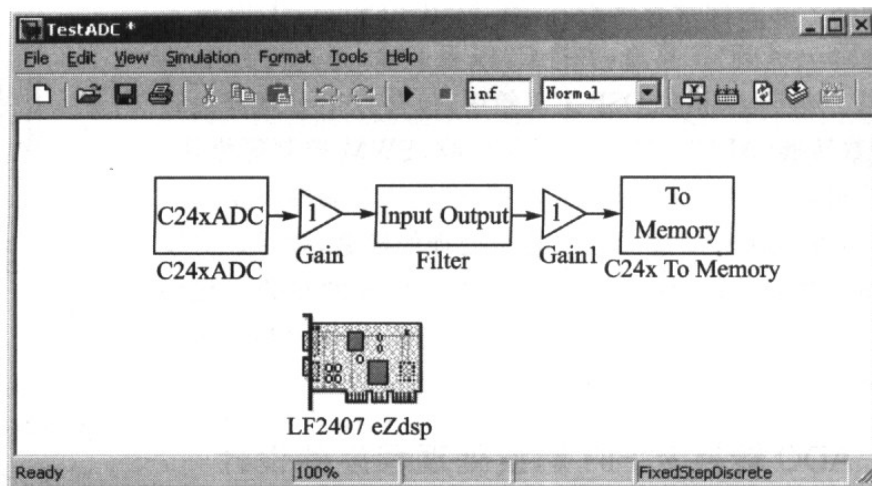


图4 ADC转换及FIR滤波 Simulink设计

图4中添加的增益模块(Gain)是为了实现数据类型的转换。由于ADC转换输出的是16位整型数据，而滤波器的输入需要双精度浮点数据，因此Gain的数据类型参数(Signal datatypes)设为float(“double”)。Gain1是将浮点数转换为整型数，因此数据类型参数设为uint(16)。图4中添加寄存器(C24x To Memory)模块是为了使设计完整。在生成的C程序中，增益模块Gain1的输出是rth-Gain1，可被其他应用程序调用。

步骤3，将图形文件生成C程序。在图4中，运行Simulation菜单下的Configuration Parameters项，弹出配置对话框，选择其中的Real-Time WorkShop项，点击Build按钮，则将图4的框图转化为TI C2000DSP的C语言代码，并自动调用CCS2软件编译运行该程序。生成的C语言程序包括：中断向量文件vectors.asm、中断服务程序MW\_c24xx\_csl.c、ADC转换控制testADC.c、主程序TestADC\_main.c等。

### 3 修改及完善程序

由Matlab直接生成的程序能够实现ADC转换及数字滤波功能，但是由于程序中使用

用了许多缺省设置，在运行过程中还存在一些问题。用上述方法生成的程序中，中断处理程序中只对定时器中断进行处理。当由于干扰信号引起其他中断时，会造成 DSP 停机，因此要在中断向量定义无效中断(`_nothing`)，并在中断响应程序中添加中断服务程序。修改后的程序中，斜体部分是修改的代码。修改程序如下：

中断向量文件 `vectors.asm`:

```

        .ref _c_int0, _schedulerTimer_ISR, wd_disable, _nothing
        .sect      "vectors"
rset:   B          wd_disable          00h reset
int1:   B          _nothing            ;02h INT1
int2:   B          _nothing            ;04h INT2
int3:   B          _schedulerTimer_ISR ;06h INT3
int4:   B 08h INT4nothing
int5:   B 0Ah INT5nothing
int6:   B 0Ch INT6nothing

```

中断服务程序 `MW_c24xx_csl.c`:

```

interrupt void schedulerTimer_ISR(void) {
    pendingInterrupt = 1;
    MMREGS[EVAIFRB] = MMREGS[EVAIFRB] & 0x0001;
    /* 清除 T2 计数器中断标志 */
}

interrupt void nothing(void) {
    enable_interrupts();          /* 清除其他中断并返回 */
    return;
}

```

在 ADC 转换处理程序中，每次 ADC 转换结束后没有复位 DSP 的排序器指针。虽然程序中设定的是只进行一个通道的 A/D 转换，但结果却是 16 个通道的循环转换，造成数字滤波器的输入数据不正确，因此程序要作修改，在 `testadc_c` 中添加：

```

TestADC_B.C24xADC=MMREGS[RESULT0]>>6; /*A/D 转换结果右移 6 位*/
MMREGS[ArN2TRL2]=MMREGS[ADCTRL2]|0X4242; /*复位排序器指针并清除 A/
D 中断标志*/

```

如果需要通过 t/o 端口输出结果，则可以在 `testadc_c` 中定义一个 I/o(portI00)，然后将 FIR 滤波后的数据 `rtb-GainI` 输出，程序如下：

```
ioport unsigned port100;          /* 定义 I/O 端口地址 */  
rtb_Gain1 = (uint16_T)_fixptlowering0;  
                                     /* FIR 滤波结果存入变量 rtb_Gain1 */  
port100=rtb_Gain1;                /* 通过 I/O 端口输出数据 */
```

#### 4 试验结果及结论

由上述方法设计的程序在 TMS3201LF2407A 处理器上能正常运行。试验结果证明，采用 FIR 滤波后，ADC 转换的抗干扰能力有了明显的提高。用 Matlab 快速建立 TI DSP 模型及直接生成 C 语言程序的方法，简化了 DSP 软件的开发。在 Embedded target for TI C2000 工具箱中还包含其他工具，如 PWM 控制、CAN 总线控制及通用 I/O 控制等。利用这些工具与 Simulink 中的其他工具相结合，能够完成复杂功能的 DSP 程序设计，并且只需进行少量的修改，就可以实现功能正确的 C 语言程序的设计，节省程序的编写及输入时间。