

# 基于 FPGA 的卡尔曼滤波器的设计与实现

仲婷婷, 王长松, 周晓敏, 齐昕

(北京科技大学 机械工程学院机械电子工程系, 北京 100083)

**摘要:** 卡尔曼滤波器在社会各领域有着广泛的应用。采用 FPGA 硬件实现卡尔曼滤波器, 解决了采用 DSP 软件方法实现存在的并行性和速度问题。文章以基于 FPGA 的数据采集系统为硬件平台, 根据模块化设计思想, 采用 VHDL 编程实现 ADS8364 芯片控制模块, 利用 FPGA 的系统级设计工具 DSP Builder 设计卡尔曼滤波器模块, 给出模块的软件仿真结果并完成整个系统的硬件验证。结果证明设计的正确性, 同时表明采用 DSP Builder, 使卡尔曼滤波器的 FPGA 硬件实现更加简单, 速度更快。

**关键词:** FPGA; 卡尔曼滤波器; ADS8364; DSP Builder; Quartus II

## Design and Implementation of Kalman Filter based on FPGA

Zhong tingting, Wang changsong, Zhou xiaomin, Qi xin

(Mechatronic Engineering Department, School of Mechanical Engineering, University of Science and Technology Beijing, Beijing 100083, China)

**Abstract:** The Kalman filter has been widely used in many society fields. The implementation of kalman filter using FPGA solves the problem of parallelism and speed existing in using DSP software. In this paper, with the hardware platform of data collection system based on FPGA and the concepts of modularization, the control module of ADS8364 is realized using VHDL, the Kalman filter model is designed using DSP Builder, results of software simulation of modules are given and hardware verification of the whole system is completed. Results prove the correctness of the design, and that it is simpler and faster to use DSP Builder to implement the Kalman Filter.

**Key words:** FPGA; Kalman filter; ADS8364; DSP Builder; Quartus II

## 1 引言

卡尔曼滤波采用状态空间法在时域内设计滤波器, 是一种最优估计算法。自从被提出以来, 获得了广泛的实际应用, 其应用领域包括控制、通讯、信号处理、石油地震勘探、故障诊断、图像处理等。在应用中, 卡尔曼滤波器通常由 DSP 处理器来实现。使用 DSP 处理器具有设计简单灵活, 可直接采用 C 语言矩阵运算的优点<sup>[1]</sup>, 但由于卡尔曼算法在运算中有大量的矩阵加减乘除运算且 DSP 处理器采用程序顺序执行的 CPU 架构, 在要求较高的场合, 不能满足系统高速、实时的需要。而用 FPGA 实现卡尔曼滤波器, 采用的是硬件并行算法, 能很好的解决速度和实时性的问题, 并且其具有灵活的可配置特性和优良的抗干扰能力<sup>[2]</sup>, 使得 FPGA 构成的数字信号处理系统非常易于修改、测试及硬件升级。

一般, 数字滤波器的 FPGA 实现是用 VHDL 或 Verilog HDL 等硬件描述语言通过编写底层代码实现。这种编程方式效率低, 难度大。利用 Altera 公司 FPGA 的 DSP 开发工具 DSP Builder 设计卡尔曼滤波器, 比基于硬件描述语言的设计, 周期更短, 设计更容易。

本文以基于现场可编程逻辑门阵列 FPGA 器件和模数转换器设计的数据采集系统为硬件平台, 进行算法设计。基于模块化设计思想, 设计了时钟分频模块, AD 转换芯片的 FPGA 控制模块和卡尔曼滤波模块。卡尔曼滤波模块采用 DSP Builder 设计, 转换成硬件描述语言 VHDL 后, 应用软件 Modelsim、QuartusII 进行仿真并完成硬件验证。

## 2 卡尔曼滤波原理

卡尔曼滤波, 是一种线性最小方差估计, 特点是考虑了系统的模型误差和测量噪声的统计特性<sup>[3, 4]</sup>。卡尔曼滤波的一般方程在许多文献中有详细讨论, 本文直接引用其结论。

设随机线性离散系统的状态方程为： $x_{k+1} = \Phi_{k+1|k}x_k + \Gamma_k u_k + G_k \omega_k$

观测方程为： $y_k = c_k x_k + v_k$

式中  $x_k$  为系统的状态向量， $y_k$  为系统的观测序列， $\omega_k$  为系统的过程噪声序列， $v_k$  为观测噪声序列， $u_k$  为系统控制输入， $\Phi_{k+1|k}$  为状态转移矩阵， $\Gamma_k, G_k$  为系数矩阵， $c_k$  为观测矩阵。

1)  $\omega_k$ 、 $v_k$  为零均值白噪声，满足

$$\begin{cases} E[\omega_k] = 0 \\ Cov[\omega_k, \omega_j] = E[\omega_k, \omega_j^T] = Q_k \delta_{kj} \end{cases}$$

$$\begin{cases} E[v_k] = 0 \\ Cov[v_k, v_j] = E[v_k, v_j^T] = R_k \delta_{kj} \end{cases}$$

其中  $Q_k$  为系统过程噪声  $\omega_k$  的对称非负定方差矩阵， $R_k$  为系统观测噪声  $v_k$  的对称正定方差阵，均为可知的，而

$$\delta_{kj} = \begin{cases} 1 & k = j \\ 0 & k \neq j \end{cases}$$

2)  $\omega_k$  和  $v_k$  不相关，即  $Cov[\omega_k, v_j] = E[\omega_k, v_j^T] = 0$

因为控制规律  $u_k$  是有规律的信号，可以先不考虑，假设  $u_k = 0$ 。设  $k$  时刻的状态估计值为： $\hat{x}_k = \hat{x}_{k|k-1} + K_k (y_k - c_k \hat{x}_{k|k-1})$ ，卡尔曼滤波的递推步骤如下所示：

$$\text{状态一步预测值： } \hat{x}_{k|k-1} = \Phi_{k|k-1} \hat{x}_{k-1} \quad (1)$$

$$\text{预测误差方差： } P_{k|k-1} = \Phi_{k|k-1} P_{k-1} \Phi_{k|k-1}^T + G_{k-1} Q_{k-1} G_{k-1}^T \quad (2)$$

$$\text{滤波增益： } K_k = P_{k|k-1} c_k^T (c_k P_{k|k-1} c_k^T + R_k)^{-1} \quad (3)$$

$$\text{最佳滤波值： } \hat{x}_k = \hat{x}_{k|k-1} + K_k (y_k - c_k \hat{x}_{k|k-1}) \quad (4)$$

$$\text{滤波误差方差： } P_k = (I - K_k c_k) P_{k|k-1} \quad (5)$$

以上即为集中式卡尔曼滤波的基本方程。只要给定滤波初值  $\hat{x}_0$  和  $P_0$ ，根据  $k$  时刻的观测  $y_k$  就可递推计算得此时刻的状态估计值  $\hat{x}_k$ 。

在一个滤波周期内，从卡尔曼滤波在使用系统信息和观测信息的先后次序来看，卡尔曼

滤波具有两个明显的信息更新过程：时间更新过程和观测更新过程。式(1)说明了根据k-1时刻的状态估计预测k时刻状态的方法，式(2)对这种预测的质量优劣做了定量描述。这两式的计算中，仅使用了与系统的动态特性有关的信息，如状态一步转移矩阵、噪声输入阵、过程噪声方差阵。从时间的推移过程来看，这两式将时间从k-1时刻推进至k时刻，描述了卡尔曼滤波的时间更新过程。其余各式用来计算对时间更新值的修正量，该修正量由时间更新的质量优劣( $P_{k|k-1}$ )、观测信息的质量优劣( $R_k$ )、观测与状态的关系( $c_k$ )以及具体的观测信息  $y_k$  所确定，这一过程描述了卡尔曼滤波的观测更新过程<sup>[5]</sup>。把以上方程改用方框图表示更能清晰的看出卡尔曼滤波的过程，如下图所示。

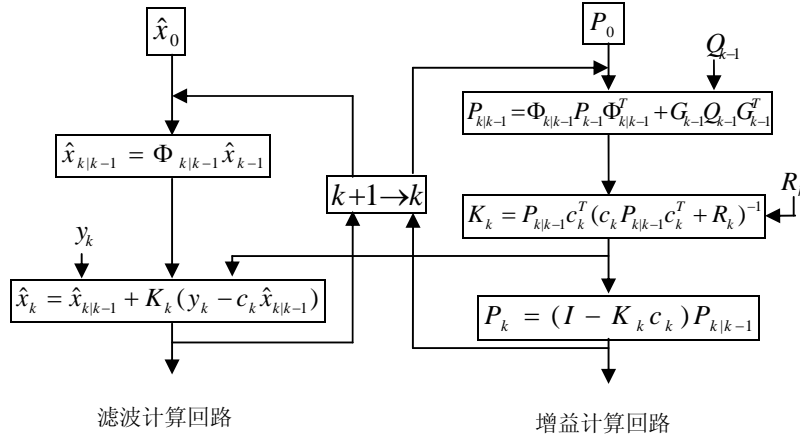


图1 卡尔曼滤波原理方框图

由图可得，卡尔曼滤波具有两个计算回路:增益计算回路和滤波计算回路。增益计算回路是独立计算的，滤波计算回路依赖于增益计算回路。卡尔曼滤波的两个回路相互联系。

本设计应用在基于高频信号注入法的无传感器永磁同步电机的转子位置检测的系统中，根据电机运动方程:

$$\sum_i T_i = J \frac{d\omega}{dt}$$

经离散化后可得状态方程:  $x_{k+1} = \Phi_{k+1|k} x_k + \Gamma_k u_k + G_k \omega_k$

式中,

$$x_k = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_k, \Phi_{k+1|k} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}, \Gamma_k = \begin{bmatrix} T^2 \\ 2 \\ T \end{bmatrix}, G_k = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, c_k = [1 \quad 0], Q_k = 2, R_k = 4.$$

系统量测方程为:  $y_k = c_k x_k + v_k$

$x_1$ 、 $x_2$  分别代表电机的转子位置角和位置角变化率。T为采样周期。先不考虑控制信号的作用。噪声  $v_k$  包括量测误差、A/D变换器产生的误差等等。在实时控制中，当外界突然扰动，如突加负载引起速度位置发生变化时，可能会给出错误的结果。不过，在电机运行过程中，这种状态持续时间相对较短，总的来看，不会造成太大影响。

### 3 FPGA 设计

尽管卡尔曼滤波原理的有良好的滤波效果，但由于其计算量大，当采样率高时，一个采样周期内难以完成计算，且计算机的字长有限，使计算中舍入误差和截断误差积累、传递，造成数值不稳定，因此用MCU和DSP难以实现。FPGA实现了并行计算，即有多个乘法器和累加器并行处理数据，采用FPGA实现卡尔曼滤波器，输入和输出数据的计算同时进行，可以大大提高滤波速度<sup>[5]</sup>。

本设计选用的FPGA是Altera公司Cyclone II 系列的EP2C20Q240C8N，其内核采用1.5V电源供电，功耗小，端口工作电压为3.3V。FPGA中I/O端口可自由定义，电路设计方便，编程灵活，不易受外部干扰。系统编译环境采用Quartus II，顶层设计为图形化方式，易实现模块化。FPGA的主要任务是控制AD转换芯片进行数据转换，并从AD芯片中读取转换输出数据，进行卡尔曼滤波处理并输出。芯片模块的划分如下图所示，分频模块和AD控制模块采用VHDL语言编程实现，卡尔曼滤波模块用DSP Builder软件进行设计。

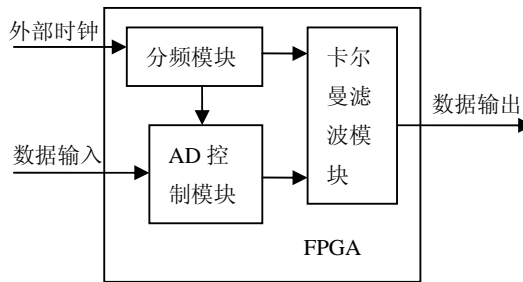


图2 芯片模块结构划分示意图

### 4 单元功能模块设计

#### 4.1 分频模块的设计

分频模块是将外部时钟进行分频设定，得到系统内部AD控制模块和卡尔曼滤波模块所需要的时钟，并为AD芯片提供时钟信号基准。此模块采用VHDL语言编程实现。实现较简单，这里直接给出其10分频时序仿真图。分频模块的外部时钟输入频率为50MHz，产生的时钟频率为5MHz。

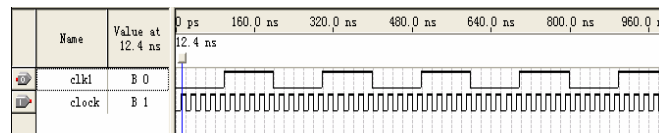


图3 10分频电路时序仿真图

#### 4.2 AD控制模块

##### 4.2.1 ADS8364 的简介与工作原理

采用ADS8364芯片对输入信号数据采样。ADS8364是TI公司的一款高速、低功耗、六路模拟输入、16位并行输出的模数转换器<sup>[6]</sup>。六路模拟输入分为三组(A、B和C)，每个输入端都有一个保持信号来实现所有通道的同时采样与转换功能，适合于多路采集系统的需要。ADS8364的时钟信号由外部提供，转换时间为20个时钟周期，最高频率为5MHz，ADS8364的六个通道可以同时进行采样/转换。ADS8364采用+5V工作电压，并带有80DB共模抑制的全差分输入通道以及六个4μs连续近似的模数转换器、六个差分采样放大器。另外，在REFin和REFout引脚内部还带有+2.5V参考电压。ADS8364的差分输入可在-VREF到+VREF之间变化。ADS8364与FPGA的接口连接如下图。

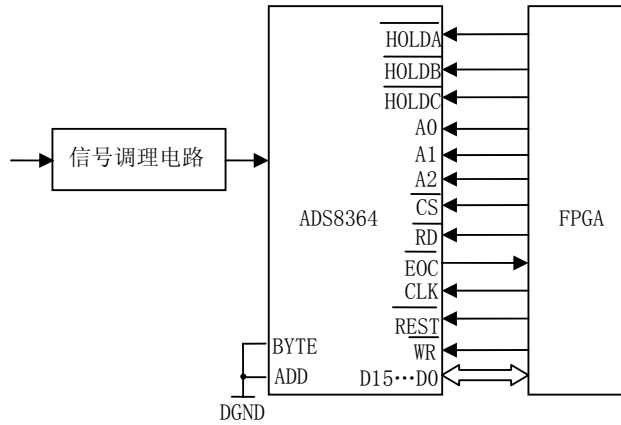


图4 ADS8364与FPGA连接示意图

三个保持信号(HOLDA,HOLDB,HOLDC)启动指定通道的转换。当三个保持信号同时被选通时,其转换结果将保存在六个寄存器中。A/D转换结果为16位,最高位为符号位,数据输出方式很灵活,分别由BYTE,ADD与地址线A2、A1、A0的组合控制。在ADS8364的HOLDX保持至少20ns的低电平时,转换开始。这个低电平可使各个通道的采样保持放大器同时处于保持状态从而使每个通道同时开始转换。当转换结果被存入输出寄存器后,引脚EOC的输出将保持半个时钟周期的低电平。通过置RD和CS为低电平可使数据读出到并行输出总线。

#### 4.2.2 AD采样控制模块

结合ADS8364的工作时序,如下图,在FPGA内部设计AD采样控制模块,为ADS8364采样提供所需的驱动信号,并从ADS8364中读取转换输出数据,同时为后续的电提供相应的控制信号。

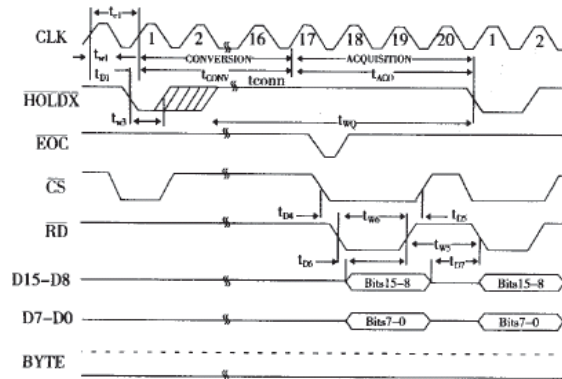


图5 ADS8364工作时序图

虽然理论上可以在模数转换的20个CLK周期进行数据读取,但这样数字接口部分可能会对模拟电路产生不良影响降低转换精度,所以最好是等模数转换过程完全结束后再读数据。另外为了确保数据读取的可靠性,需要在A2,A1和A0等控制信号的状态完全稳定后才进行数据读取<sup>[7]</sup>。本设计以一通道的数据采样为例,状态机的状态定为22个(20个AD转换状态,2个读数据状态),状态机的输出控制位为16位。采用VHDL语言编程实现状态机后,生成图元文件。图6所示为AD采样控制模块与分频模块的图元文件的连接。ADS8364的时钟输入由50MHz系统时钟10分频得到(AD芯片与其控制模块使用相同的5MHz时钟信号)。编译后进行时序仿真,仿真波形如图7所示。

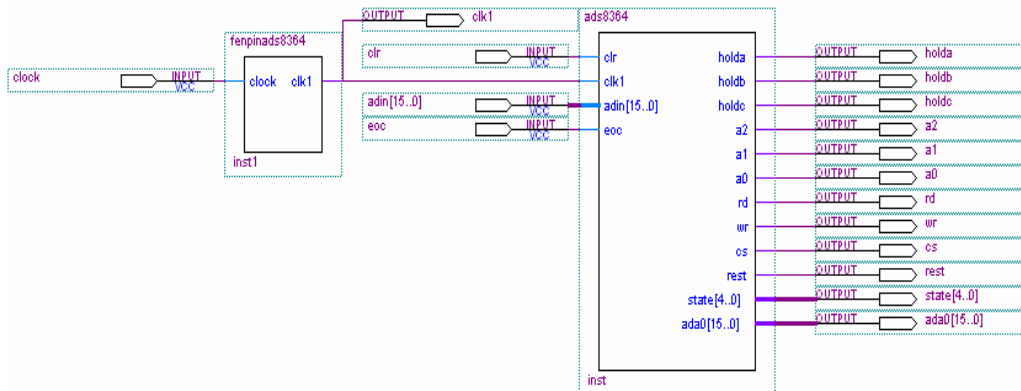


图 6 AD 采样控制模块与分频模块的图元文件连接图

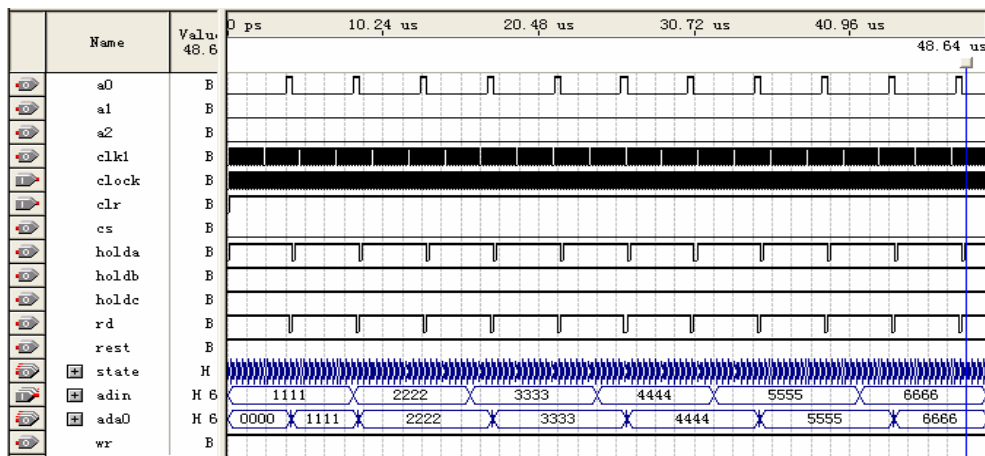


图 7 AD 采样控制模块时序仿真波形

### 4.3 卡尔曼滤波模块

#### 4.3.1 DSP Builder 简介

DSP Builder 是 Altera 公司推出的 FPGA 的系统级设计工具<sup>[8]</sup>。它将 Matlab 和 Simulink 系统级设计工具的算法开发、仿真和验证功能与 QuartusII 的基于 Verilog HDL 及 VHDL 语言的设计流程整合在一起,实现了这些工具的集成,为用户提供了一个从软件到硬件的完整的 DSP 开发平台。

使用 DSP Builder 设计数字信号处理系统的流程为:

- (1)在 Matlab/Simulink 中建立一个\*.mdl 模型文件,用图形方式调用 Altera DSP Builder 和 Simulink 其他库中的图形模块进行数学模型设计;
- (2)利用 Simulink 图形化仿真分析功能对此设计模型进行纯数学上的仿真、验证及修改;
- (3)通过 Altera DSP Builder 中的 SignalCompiler 模块把 Matlab/Simulink 的设计文件(.mdl)转成相应的硬件描述语言 VHDL 设计文件(.vhd),以及用于控制综合与编译的 TCL 脚本;
- (4)在 SignalCompiler 里用选定的工具自动进行综合和编译,或用 QuartusII 调用上述 VHDL 文件,选择器件型号锁定引脚重新编译;
- (5)配置芯片(program device),即把 SRAM 对象文件(\*.sof)下载到 FPGA 芯片。

这种设计方式将已经成熟的在 MATLAB 中进行 DSP 算法设计优化的过程,与先进的 FPGA 硬件实现技术有机地融合到一起,实现软件设计与硬件验证的直接握手,大大简化设计过程,提高设计效率,它的出现进一步推动了 FPGA 在数字信号处理领域的应用。

#### 4.3.1 卡尔曼滤波算法矩阵运算的分解

卡尔曼滤波算法涉及到矩阵运算，把矩阵运算分解成一系列加减乘除运算单元，利用 DSP Builder 的加减乘除等模块来建模实现。利用平行的硬件技术来实现算法，可以提高执行速度<sup>[9, 10]</sup>。在本设计中，卡尔曼滤波算法的 5 个方程可以分解成如下的形式。

(1) 状态一步预测值  $\hat{x}_{k|k-1}$  分解为：

$$\hat{x}_{k|k-1} = \Phi_{k|k-1} \hat{x}_{k-1} = [xl_1 \quad xl_2]_k^T$$

其中：  $xl_1 = x_1 + Tx_2$

$$xl_2 = x_2$$

(2) 预测误差方差  $P_{k|k-1}$  分解为：

$$P_{k|k-1} = \Phi_{k|k-1} P_{k-1} \Phi_{k|k-1}^T + G_{k-1} Q_{k-1} G_{k-1}^T = \begin{bmatrix} pl_{11} & pl_{12} \\ pl_{21} & pl_{22} \end{bmatrix}$$

其中：  $pl_{11} = p_{11} + Tp_{12} + Tp_{21} + T^2 p_{22}$

$$pl_{12} = p_{12} + Tp_{22}$$

$$pl_{21} = p_{22} + Tp_{22}$$

$$pl_{22} = p_{22} + 2$$

(3) 滤波增益  $K_k$  可分解为：

$$K_k = P_{k|k-1} c_k^T (c_k P_{k|k-1} c_k^T + R_k)^{-1} = [K_1 \quad K_2]_k^T$$

其中：  $K_1 = pl_{11} / (pl_{11} + 4)$

$$K_2 = pl_{11} / (pl_{21} + 4)$$

(4) 最佳滤波值  $\hat{x}_k$  可分解为：

$$\hat{x}_k = \hat{x}_{k|k-1} + K_k (y_k - c_k \hat{x}_{k|k-1}) = [x_1 \quad x_2]_k^T$$

其中：  $x_1 = xl_1 + K_1 (y - xl_1)$

$$x_2 = xl_2 + K_2 (y - xl_1)$$

(5) 滤波误差方差  $P_k$  分解为：

$$P_k = (I - K_k c_k) P_{k|k-1} = \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix}$$

$$\text{其中: } p_{11} = pl_{11} - K_1 pl_{11}$$

$$p_{12} = pl_{12} - K_1 pl_{12}$$

$$p_{21} = pl_{21} - K_2 pl_{11}$$

$$p_{22} = pl_{22} - K_2 pl_{12}$$

输出预测  $\hat{y}_k$  可分解为:  $\hat{y}_k = c_k \hat{x}_{k|k-1} = xl_1$

由于卡尔曼滤波的基本方程是时间域内的递推形式, 其计算过程是一个不断的“预测—修正”过程, 在求解时不要求存储大量的数据, 并且一旦观测到了新的数据, 随时可以算得新的滤波值, 因此这种滤波方程非常便于实时处理。

#### 4.3.1 卡尔曼滤波模型的建立

在 Matlab/Simulink 中建立一个\*.mdl 模型文件, 根据以上的等式, 利用 DSP Builder 库和 Simulink 库中的图形模块进行设计输入。本设计采用层次化设计, 依据卡尔曼滤波算法分解的等式, 利用 DSP Builder 库中的加减乘除模块以及端口模块来实现建模。建立的顶层系统模型如下图所示。

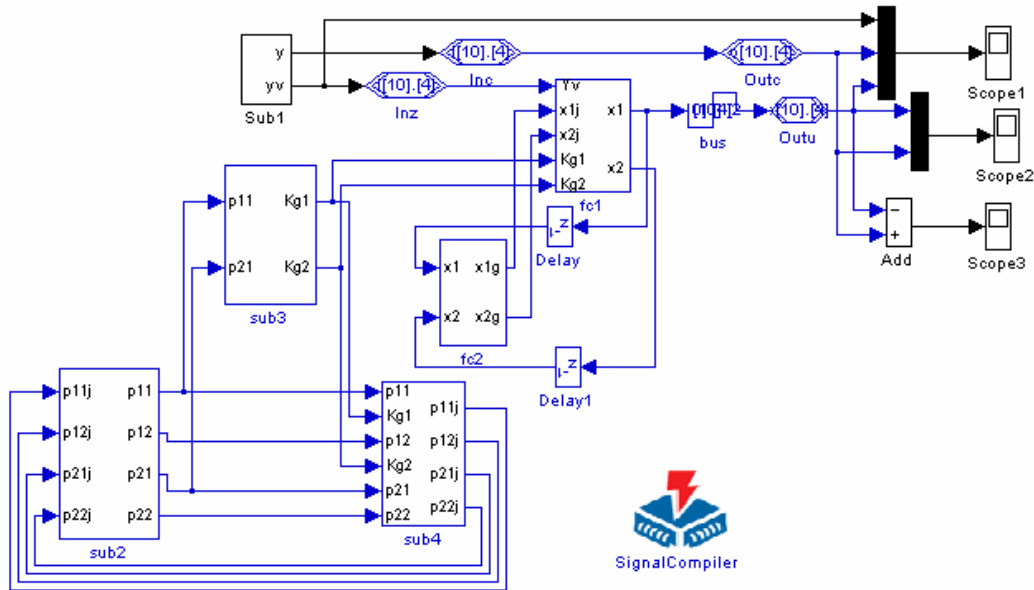


图8 卡尔曼滤波系统模型

上图中, sub1 为来自 Simulink 库中的输入信号模块, 信号  $y$  为未叠加噪声的原信号,  $yv$  为叠加高斯白噪声的待滤波的信号。fc1,fc2,sub2,sub3,sub4 为卡尔曼滤波的五个子系统模

块。其中, fc2 为依据最佳滤波值:  $\hat{x}_k = \hat{x}_{k|k-1} + K_k (y_k - c_k \hat{x}_{k|k-1})$  方程分解的等式所建立的子系统模型, 如下图所示。同理根据其他方程分解的等式, 利用 DSP Builder 库中的加减乘除图形模块以及端口模块来实现建模, 这里不详细给出。

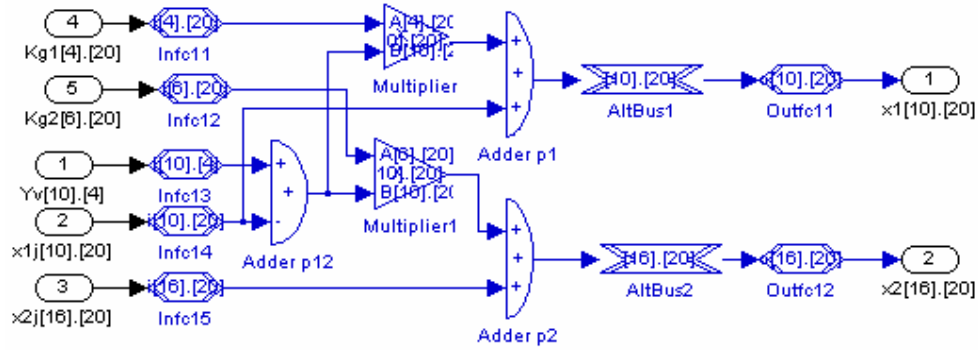


图9 fc2子系统模型

滤波增益方程  $K_k = P_{k|k-1} c_k^T (c_k P_{k|k-1} c_k^T + R_k)^{-1}$  对应的子系统模块用到了除法模块。由于 DSP Builder 库中除法模块只能给出商结果的整数部分和余数，所以在运行除法之前，给被除数乘以适当的增益以保证结果的精度（可以采用移位运算以提高运算速度）。设计中通过设定小数点在数中的不同位置，表示不同大小和精度的小数。

卡尔曼滤波整个过程中用到多个加法、乘法运算和除法运算，这些运算混合在一起，卡尔曼滤波比较复杂，计算量大。误差方差  $P_k$  存在稳态值，从任意初始方差阵  $P_0$  算起，当算步  $k \rightarrow \infty$  时， $P_k \rightarrow P_w$ （稳态值），由递推公式可知， $P_k$  稳定后增益矩阵  $K_k$  也随之稳定了，因此若用  $K_k$  的稳态值  $K_w$  来代替  $K_k$ ，则只是前面若干步有误差，随着递推次数的增多，估计就越来越精确，这样可以离线将  $K_w$  事先算好，作为常系数储存起来，从而省去了每一步对  $K_k$  的递推运算，使整个运算得到简化。相应的，利用 DSP Builder 软件建立 kalman 模块也得到简化，节省硬件资源，如下图所示。

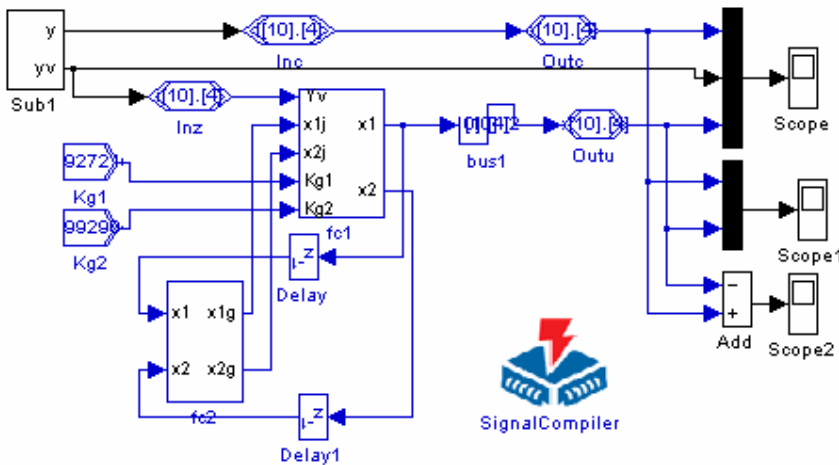


图10 离线卡尔曼滤波系统模型

#### 4.3.1 卡尔曼滤波模型的仿真结果

##### (1) Simulink 仿真结果

采用叠加高斯白噪声的正弦信号作为输入信号，对建立的滤波器模型进行仿真。从以下仿真结果看出，叠加噪声的输入信号通过滤波后，与未叠加噪声的正弦信号相吻合，待信号达到稳定后，误差在 0.5 度以内。

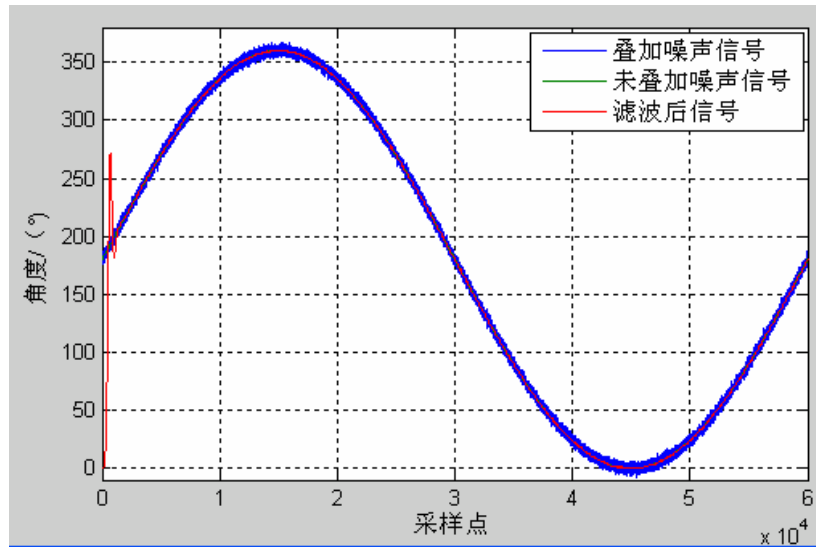


图11 叠加白噪声的正弦输入信号

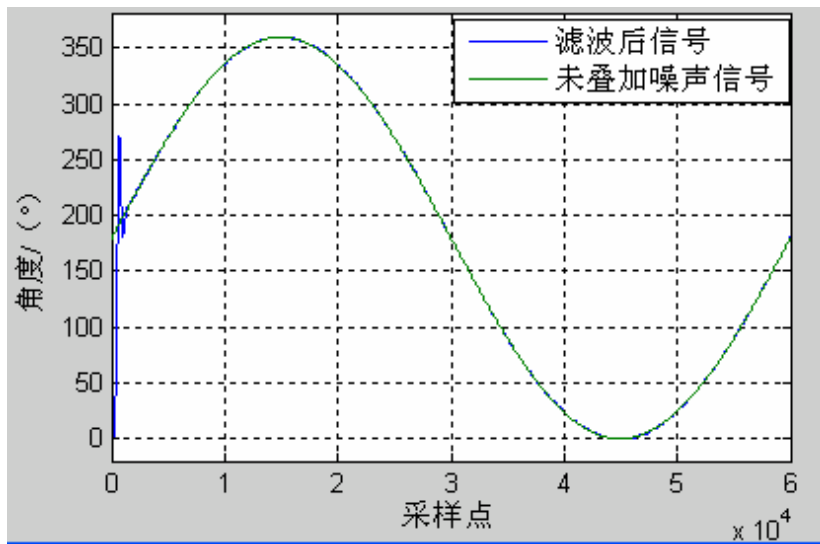


图12 滤波后的信号和未叠加白噪声的正弦输入信号

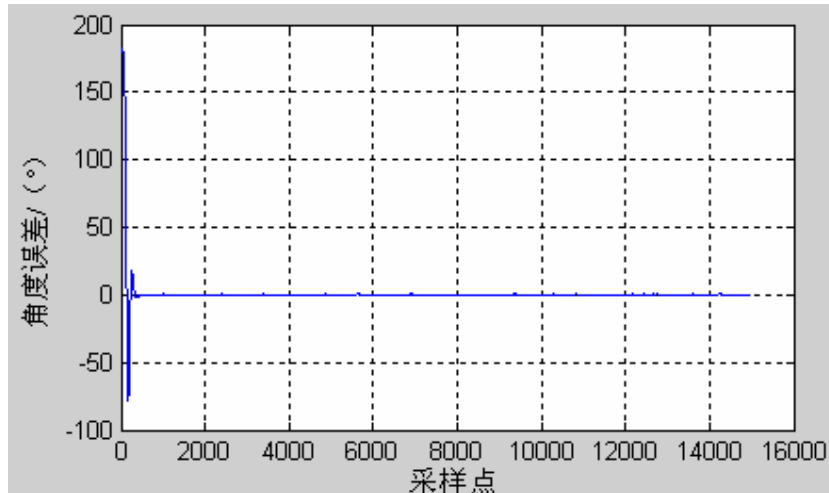


图13 误差波形

### (2) modelsim 功能仿真

在 Simulink 中进行的仿真是属于系统验证性质的，是对\*.mdl 文件进行的仿真，并没有对生成的 VHDL 代码进行过仿真。生成 VHDL 描述是 RTL 级的，是针对具体的硬件结构的，而在 Matlab 的 Simulink 中的模型仿真是针对算法实现的，这二者之间有可能存在软件理解上的差异，转换后的 VHDL 代码实现可能与 mdl 模型描述的情况不完全相符，这就需要针对生成的 RTL 级 VHDL 代码进行功能仿真<sup>[11]</sup>。

利用 Modelsim 进行功能仿真。设置输入输出信号均为模拟形式，仿真波形如图 14 所示的，与 Simulink 里的仿真结果一致。

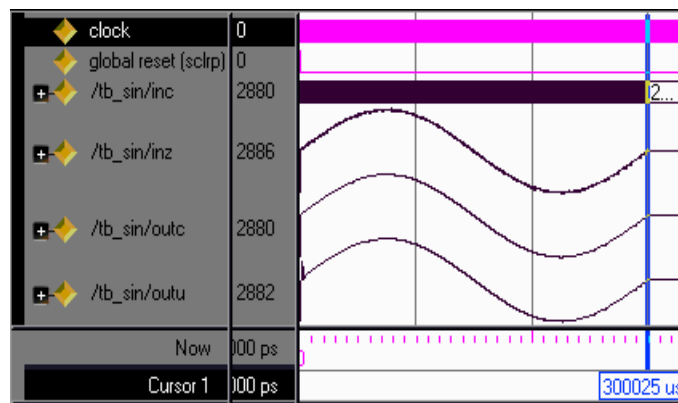


图14 modelsim功能仿真图

### (3) Quartus II 时序仿真

在 Quartus II 环境中打开 DSP Builder 建立的项目文件，选择器件型号进行编译和时序仿真，得到时序波形如下图所示。

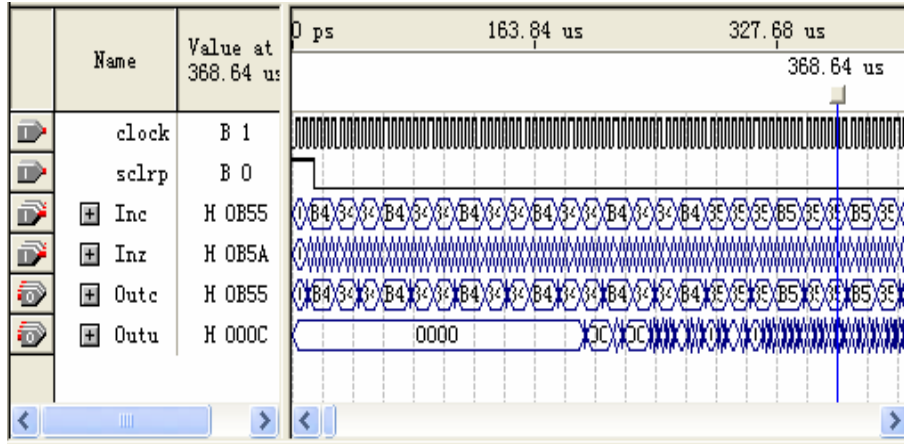


图15 卡尔曼滤波的时序仿真图

用 DSP Builder 设计卡尔曼滤波模型是整个系统设计中的一个子模块，把卡尔曼滤波模型转化后的 VHDL 文件生成图元文件，如下图所示，以便在整个系统设计中调用。

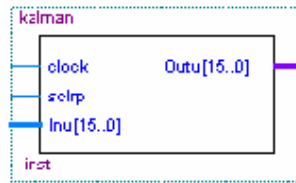


图16 生成的卡尔曼图元文件

在 Quartus II 环境下，整个系统原理图设计中，调用各个子模块，构成完整的设计，然后进行编译，仿真，引脚分配等工作。最后配置芯片，把 SRAM 对象文件(\*.sof)下载到 FPGA 芯片中，对硬件进行测试，利用广州致远电子有限公司的 LA1032 逻辑分析仪对 16 位实验数据进行读取（如下图所示），并转换处理，验证本设计的正确性。

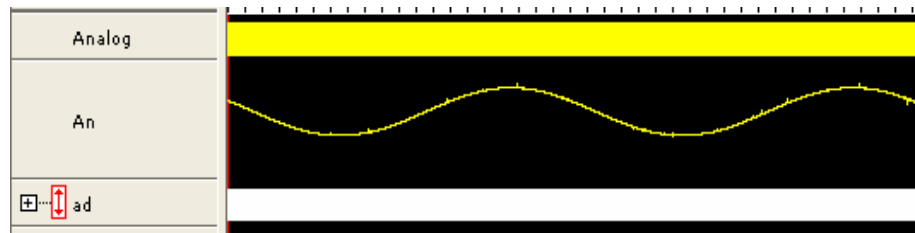


图 17 LA1032 逻辑分析仪读取数据

#### 4 结束语

卡尔曼滤波器在多领域中有重要的作用。FPGA 可实现数字信号处理功能，并能满足实时性的要求。本文以基于 FPGA 器件和 AD 转换器的数据采集系统为硬件平台，进行了数据采集及卡尔曼滤波算法设计，尤其对基于 DSP Builder 卡尔曼滤波器的设计实现进行了详述。采用 DSP Builder 进行设计是从与硬件完全无关的系统级仿真开始，便于研究者迅速地将算法级的构思应用于系统设计中，从而可以专注于系统级算法的设计，避免了繁琐的语言编程和电路设计，缩短系统实现的时间周期。在实际应用中，可以根据需要，对转化的 VHDL 程序进行修改扩充与优化，并可作为 IP 核的形式进行重复利用。

#### 参考文献：

[1] 韩梅, 陈禾. 卡尔曼滤波器的FPGA实现[J]. 电子工程师, 2006, 32(7): 38-40.

- [2]潘松,黄继业,王国栋.现代DSP技术[M].西安:西安电子科技大学出版社,2003.
- [3]郑建芬.基于高频注入和卡尔曼滤波的永磁同步电机无传感器控制[D]:学位论文.沈阳:沈阳工业大学电力电子与电力传动专业,2005.
- [4]钟黎萍,王长松,周晓敏,巩宪峰.卡尔曼滤波器在高频信号注入法中的应用[J].机电产品开发与创新,2007,20(5):141-142.
- [5]胡剑琛.微机继电保护数据预处理的研究[D]:学位论文.北京:华北电力大学电力系统及其自动化专业,2004.
- [6]谷重阳,汪渤,龙殊颖.并行模数转换器 ADS8364 与 TMS320F2812 的接口设计[J].微计算机信息,2006,2(22):137-139.
- [7]谭恒.高精度多通道 AD 芯片 ADS8364 及其在有源电力滤波器中的应用[J].电子科技,2005,7:10-13.
- [8]潘松,黄继业.EDA技术与VHDL(第2版)[M].北京:清华大学出版社,2007.
- [9]陈刚,郭立,史洪生,杨毅.最大平行结构的kalman滤波器的硬件电路设计[J].微电子学与计算机,2006,23(6):34-37,41.
- [10] C.R. Lee, Z. Salcic. High-performance FPGA-based implementation of Kalman filter. Microprocessors and Microsystems 21 (1997) 257-265.
- [11] 齐海兵,刘雄飞,张德恒.基于FPGA的数字滤波器的设计与实现[J].现代电子技术,2006,15:70-71.

#### 论文原创性声明:

本人郑重声明,该论文是本人在导师的指导下,在北京科技大学攻读硕士学位期间进行研究工作所取得的成果。据本人所知,论文中除了特别加以标注的地方外,不包含他人已经发表或撰写过的研究成果。本声明的一切法律结果由作者承担。

作者签名:仲婷婷

日期:2008年5月25日

联系方式:

仲婷婷

通信地址:北京科技大学 635 信箱 100083

电子邮箱:zhott0865@163.com

联系电话:010-62333342(实验室) 13717994967