Proceedings of the 5th WSEAS Int. Conf. on Signal Processing, Computational Geometry & Artificial Vision, Malta, September 15-17, 2005 (pp61-65)

The FPGA Implementation Of Kalman Filter

GANG CHEN and LI GUO
Department of Electronic Science and Technology
University of Science & Technology of China
CHINA

Abstract: - Based on the fact that Faddeev's algorithm can be easily mapped into the Systolic array for implementing. An FPGA implementation of Kalman Filter using Modified Faddeev [1] is proposed The Modified Faddeev uses Neighbor pivoting for triangularization substituting the Gaussian elimination.Gaussion elimination may cause the overflow of the datas, and Neighbor pivoting can guarantee the stability of data stream. Moreover due to apply the technology of resource sharing, we use one trapezoidal array instead of bitrapezoidal array [2], thus reducing the silicon area. Techniques employed for data skewing and storage organization are efficient, then reducing the complexity of control and increasing the speed of computation.

KeyWords: - FPGA,Kalman filter, Modified Faddeev's algorithm, Systolic array, Implemantation

1 Introduction

Since the Kalman filter [3] was introduced by R.E.Kalman in 1960, it has been widely used in the areas of modern control, signal processing,air-borne control systems, adaptive controls, radar signal processing, missile control, and on-board calibration of inertial systems. Being an optimal recursive estimator, Kalman filter provides a real-time algorithm to estimate the unknown state vector recursively for each measurement based on minimization of the mean squre error, which is a measurement of the quality of noisy data processing. Direct implementation of the kalman filtering algorithm is not efficient, because of its computational complexity , which involves many matrix multiplications and inversions. Based on the fact that Faddeev's algorithm can be easily mapped into the Systolic array for implementing [4]. Many authors have implemented the Kalman Filter directly using Faddeev's algorithm [5][6].

In this paper, an efficient systolic implementation of the Kalman filtering problem using the Modified Faddeev's algorithm and one trapezoidal array is presented. In our proposed implemementation, the Modified Faddeev's algorithm is used through Neighbor pivoting for triangularization substituting the Gaussian elimination, which guarantees the stability of data stream, also, the technology of sharing resource is appled in designing cell to reducing the silicon aeras .

2 Kalman filtering problem

Kalman filter is an optimal linear estimator which provide the estimation of signals in noise. Kalman used the state transition models for dynamic system. Kalman filter equations can be sloved numerically by using a recursive type structure whose outputs only depend on the current inputs and current states (previous output). The system and measurement model equations are:

State equation ：

$$X(k+1) = \Phi(k+1,k)X(k) + w(k) --- (1)$$

Measurement equation：

$$Y(k) = H(k)X(k) + V(k) --- (2)$$

Where

$$X(k+1) = [X_1(K) \quad X_2(K) \quad X_3(K) \quad X_4(K)]$$

$$Y(k) = [Y_1(K) \quad Y_2(K)]$$

W(k) is dicrete white noise serial, and $E[W(k)W^T(k+j)] = 0$.

V(k) is the measurement noise, and they are assumed to be white Gaussian noise.

The optimal estimate $\hat{X}(k)$ based on minimum covariance is given by the following set of Equations:

$$\hat{X}(k \mid k-1) = \Phi(k,k-1)\hat{X}(k-1 \mid k-1) + U(k)\bar{a}(k) \quad --(4)$$

$$K(k) = P(k \mid k-1)H^T(k)\left[H(k)P(k \mid k-1)H^T(k) + R(k)\right]^{-1} \quad --(5)$$

$$P(k \mid k-1) = \Phi(k,k-1)P(k-1 \mid k-1)\Phi^T(k,k-1) + Q(k-1) \quad --(6)$$

$$P(k \mid k) = \left[I - K(k)H(k)\right]P(k \mid k-1) \quad --(7)$$

Where R and Q are the covariance matrices of observation and system noises, respectively.

$\hat{X}(0)$ 和 $P(0)$ are known initially.

3 The implement of Kalman Filter

3.1 Faddeev's algorithm

Consider a matrix F as following :

$$F = \begin{bmatrix} A & B \\ -C & D \end{bmatrix}$$

Where A,B,C,D are matrices , and Faddeev's algorithm does the following's linear transformation:

$$\begin{bmatrix} A & B \\ -C & D \end{bmatrix} \Rightarrow \begin{bmatrix} A' & B' \\ -C+WA & D+WB \end{bmatrix}$$

If $W = CA^{-1}$, then the lower left-hand side are zeros, then D+WB $= D + CA^{-1}B$ is the desired output, appearing in the bottom right-hand quadrant after the process to annul the bottom left-hand quadrant.

By selecting appropriate valves for A,B,C,D in compound matrix E ,a systolic standard Kalman filter can be implement.

Then through analyzing the equations (3)—(4), Table 2 defines data required for the computation of Kalman filter [7].

3.2 Faddeev's algorithm mapped onto Systolic array [8]

We used the trapezoidal array illustrated in Fig.1 [4] to implement the Faddeev's algorithm. If the input matrix is 2*n rank, then the Systolic array is maked up of sub-array T and sub-array S , which including n*(n-1)/2 PE and n*(n-1)/2 PE, respectively. There are two types of PE : square and circular PE. As shown in the Fig.1, the elements of matrix A are firstly fed to the sub-array T, and B are fed to sub-array S, and both of them are fed to the array in a skewed way as shown in Fig.1. This skewing can be achieved through delay cells. The elements of matrix A are triangularized in the sub-array T, then being stored in the PE of sub-array T. At the same time, the multiplier M is fed to the right-hand sub-array S, and the same row elements of B make the same transformation, storing in the PE of sub-array S. The column elements of matrix C are fed into sub-array T after the matrix A, after the transformation , all the elements of matric C are zeros, and at the same time, the multiplier M is fed to the right-hand sub-array S. And the same row elements of D make the same transform after the transform, the desired result matrix E are out through the bottom of the sub-array S [4].

The data input is in a skewed way, then finish the triangularization and elimination. The trapezoidal array for solving four-state standard Kalman filter has 26 processing cells including 4 boundary cells and 22 internal cells. The processing elements have been specially designed and implemented, which will be illustrated in the following charter.

Proceedings of the 5th WSEAS Int. Conf. on Signal Processing, Computational Geometry & Artificial Vision, Malta, September 15-17, 2005 (pp61-65)

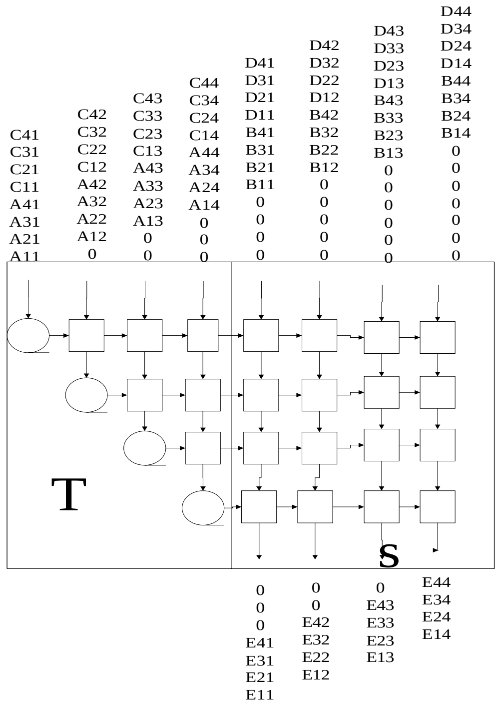

Fig.1 The architecture of Kalman filter using systolic array

3.3 The basic PE of applying Modifying Faddeev's algorithm

Traingularisation of matrix A is implemented by neighbor pivoting, because of its simple implementability and it also makes data stream stability than the Gaussian elimination. Triangularization using Given's Generation and Rotation[4] would require square rooting, which is not easily implemented. But neighbor pivoting is not used in handling the elements of matrix C, because neighbor pivoting may require change the position of the row , which may cause to change the row of matrix A and the row of matrix C and accordingly may change the row of matrix B and the row of matrix D, thus the output may not be the $D+CA^{-1}B$.

In this paper, we will use neighbor pivoting to handle the matrix [A B], and use the Gaussian elimination to handle the matrix [C D].This will require PE both have the function of Gaussian elimination and neighbor pivoting, which will be achieved by selecting the operation in term of the flag of the PE.

As shown in Fig.2, the basic systolic cell architecture of implementing triangularization

As shown in Fig.3, the basic systolic cell architecture of implementing elimination.

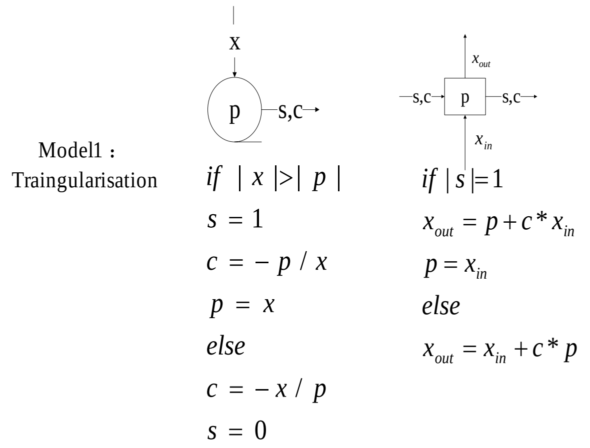

Fig.2 Neighbor pivoting for triangularizatior

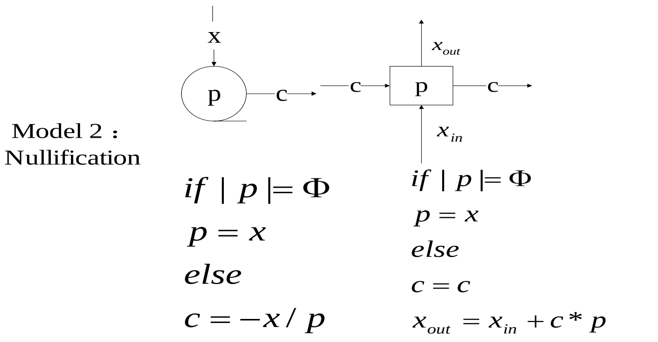

Fig.3 Gaussian elemination for nullification

3.4The architecture of basic PE

As shown in Fig.4 and Fig.5, there are hardware architecture of internal cell and boundry cell, respectively. Where COUNT is to generate the flag to control the PE to select Neighbor pivoting or Gaussian elimination for triangularization. MUXF is to select Neighbor pivoting or Gaussian elimination for triangularization. REG is register, MUXD is to select the input datas for shared unit. ABS is absolute value, MUX is multiplexer, INV is inverter, ADD is adder, MUL is multiplier, and DIV is divider, COMP is comparator.

All the datas in the paper are 32 floating point. The design is implemented in an Altera Stratix chip. All designed modules base on the library parameterized modules(LMP)[9]. Multipliers use the chip internal DSP units ,which includes a Wallace tree and Baugh and Wooley method.

These elements have performance figures as shown in Table 1.

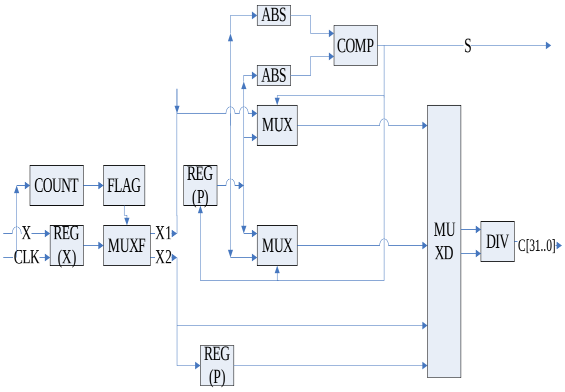

Fig.4 Design of boundary cell in Fig.2

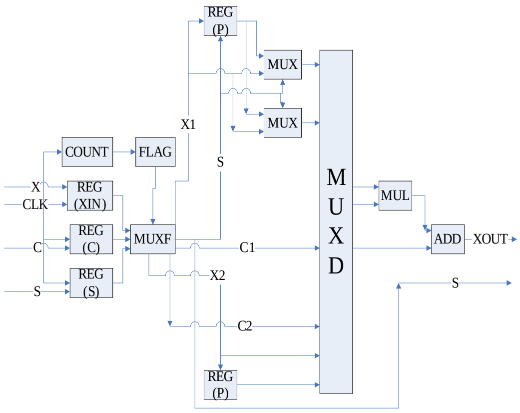

Fig.5 Design of Internal cell in Fig.3

3.5 The FPGA implement of Kalman filters

In the following, we illustrate the FPGA implement of Kalman filter using the modified Faddeev's algorithm. Table 2 defines datas required for the computation of Kalman filter and executed using trapezoidal array. The new data (A,B,C,and D) of each step could be shifted into the array from the top, row by row, as the calculation proceeds.

The overall architecture of the kalman filter as shown in Fig.6.

Where CONTROL is to generate the control signal to control TDR (Temporary Data Register), AGU (Address Generate Unit) and IOS (input and output selection).

FIFO Units are consisted of four FIFOs with the width 32 bits and the depth 8. The FIFO Units and TDR together are to handle the output, because Kalman filter is an recursive way , the output will be used in the following process.

TDR is to store the data that may be used for the next next step.

ROM Units are consisted of 8 ROM with the width 32 and the depth 48, and they are to store the constant matrixs.

AGU is to generate the address for ROMs, controlled by CONTROL to generate the correct address to control the ROMs .

DI is the data input IO. There are two clock, and CIN is system clock , controlling data output.

CLKIN is data input clock, control $X(k)$.

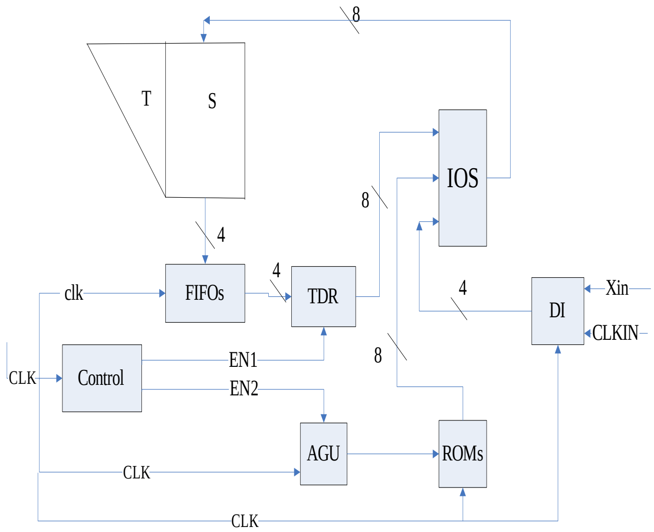

Fig.6 The overall hardware architecture of Kalman filters

4 Conclusion

In this paper, an effiective implement of Kalman filter is presented. Using the Modified Faddeev's algorithm guarantees the stability of data stream, but also due to apply the technology of sharing resource , we use one trapezoidal array instead of bi-trapezoidal array thus reducing the silicon area..

Table.1
Implementation of Arithmetic Element

Arithmetic Elements	Size (bits)	Space Occupied (LCs)	Delay Time (ns)	DSP Elements
Adder Subtracter	32 + 32 32 + 32	348 348	58	
Multiper	32 + 32	53	92.8	8
Divider	32 + 32	891	163	

Table.2
Kalman filter using Faddeev's Algotithm

Step	A	B	C	D	D+ C A^{-1} B
1	1	$\hat{X}(k-1\|k-1)$	$\Phi(k,k-1)$	0	$\hat{X}_1(k\|k-1)$
2	1	$P(k-1\|k-1)$	$\Phi(k,k-1)$	0	$\Phi(k,k-1)P(k-1\|k-1)$
3	1	$\Phi^T(k,k-1)$	$\Phi(k,k-1)P(k-1\|k-1)$	$Q(k-1)$	$P_1(k\|k-1)$
4	1	$H^T(k)$	$H(k)$	0	$P_1(k\|k-1)H^T(k)$
5	1	$P_1(k\|k-1)H^T(k)$	$P_1(k\|k-1)$	$R(k)$	$H(k)P(k\|k-1)H^T(k)+R(k)$
6	$H(k)P(k\|k-1)H^T(k)+R(k)$	1	$P_1(k\|k-1)H^T(k)$	0	$K(k)$
7	1	$[P_1(k\|k-1)H^T(k)]^T$	$-K(k)$	$P_1(k\|k-1)$	$P(k\|k)$
8	1	$\hat{X}(k-1\|k-1)$	$-H(k)$	$Y(k)$	$[Y(k)-H(k)\hat{X}(k\|k-1]$
9	1	$[Y(k)-H(k)\hat{X}(k\|k-1]$	$K(k)$	$\hat{X}_1(k\|k-1)$	$\hat{X}(k\|k)$

References:

[1] Nash J G ,Hanse S.Modified Faddeev algorithm for matrix manipulation Proc. 1984 SPIE Conf

[2] Padma Rao and Magdy A Bayoumi An effective VLSI implementation real-time Kalman filter 1990 IEEE

[3] R.E.Kalman, A new approach to linear filtering and prediction problems.Journal of Basic Engineering 82D(1960)34—45.

[4] Guo lang Chen VLSI calculation theory and parallel algorithm USTC Publishing House 1991

[5] Gentlelman W M , Kung H T . Matrix triangularization by systolic arrays . Proceedings of SPIE—The international Society of Optical Engineering ,Aug. 1981, 19—26

[6] Chuang Y H, He G. Design of problem—size independent systolic array systems. Proc. InternationalConference on computer Design: VLSI in Computers.1984,152—187

[7] Zoran Salcic ,Chung-Ruey Lee Scalar-Based Direct Algorithm Mapping FPLD Implementation of a Kalman Filter IEEE Transaction On Aerospace And Electornic Systems VOL.36,NO.3,JULY 2000.

[8] Hen—Geul Yeh Systolic Implementation on Kalman Filters 1988 IEEE

[9] Altera Corporation Data Book,1996