ISSN : 2229-6093

FPGA Based Kalman Filter for Wireless Sensor Networks

Vikrant Vij* , Rajesh Mehra**

*ME Student, Department of Electronics & Communication Engineering
**Faculty , Department of Electronics & Communication Engineering
National Institute of Technical Teachers' Training & Research, Sector-26, Chandigarh, India
* vikrantvij@gmail.com, ** rajeshmehra@yahoo.com

Abstract

A Wireless Sensor Network (WSN) is a set of tiny and low-cost devices equipped with different kind of sensors, a small microcontroller and a radio transceiver, typically powered by batteries. Target tracking is one of the very important applications of such a network system. Traditionally, KF (Kalman filtering) and its derivatives are used for tracking of a random signal. Kalman filter is a linear optimal filtering approach, to address the problem when system dynamics become nonlinear, researchers developed sub-optimal extensions of Kalman filter, two popular versions are EKF (extended Kalman filter) and UKF (unscented Kalman filter).The rapidly increasing popularity of WSNs has placed increased computational demands upon these systems which can be met by FPGA based design. FPGAs offer increased performance compared to microprocessors and increased flexibility compared to ASICs , while maintaining low power consumption

Key Words: Kalman filter**,** FPGA, Simulation, VHDL

1. Introduction

Wireless Sensor Networks (WSNs) consist of hundreds or thousands of sensor nodes. These sensor devices comprise sensing, processing, wireless data transmission and power components. They are usually deployed at a large scale to collect the physical world information for a wide variety of applications. Target tracking is one of the very important applications of such a network system. Target tracking is a path/trajectory prediction method for environmental targets such as vehicles, missiles, animals, hostiles, or even unidentified objects. The practical applications of wireless sensor networks require the sensor devices to be high in computation ability, low in power consumption, small in size, as well as competitive in cost. Kalman filtering provides a method to reduce sensor measurement noise, and thus provides a more accurate prediction of a target's future path. Kalman filter is an estimator for estimating the instantaneous state of linear dynamic system perturbed by white noise, by using measurements linearly related to the state but corrupted with white noise. The resulting estimator is statistically optimal with respect to any quadratic function of estimation error.

Kalman filter algorithm does not lent itself for easy implementation; this is because it involves many matrix multiplication, division and inversion. FPGA-based architecture and methodology (due to inherent advantage of parallelism) provides increased WSN flexibility and resulting in superior power consumption and performance compared to a microprocessor capable of satisfying similar demands. Both wireless sensor Networks and FPGA's are independent areas of research and their integration is an emerging area of research. New and innovative WSN applications demand longer battery life, faster responses, and enhanced results FPGAs offer increased performance compared to microprocessors and increased flexibility compared to ASICs, while maintaining low power consumption. FPGA is advancing rapidly as a highly important element of the future of computing. Already developments have shown that it can massively reduce the price of specialized system development and it can compete on a variety of attributes with top range commercially available microprocessors.

2. Discrete Kalman Filter

The Kalman filter addresses the general problem of trying to estimate the state $x \in R^n$ of a discrete-time controlled process that is governed by the linear stochastic difference equation

$$x_k = Ax_{k-1} + Bu_{k-1} + w_{k-1} \qquad (1)$$

with a measurement that is $z \in R^m$

$$z_k = Hx_k + v_k \qquad (2)$$

The random variables w_k and v_k represent the process and measurement noise (respectively). They are assumed to be independent (of each other), white, and with normal probability distributions

$$P(w) \cong N(0,Q)$$
$$P(v) \cong N(0,R) \qquad (3)$$

In practice, the *process noise covariance Q* and *measurement noise covariance R* matrices might change with each time step or measurement, however here we assume they are constant. The *n x n* matrix *A* in the difference equation (1) relates the state at the previous time step *k-1* to the state at the current step *k* , in the absence of either a driving function or process noise. Note that in practice *A* might change with each time step, but here we assume it is constant. The matrix *B* relates the optional control input $U \in R^l$ to the state *x*. The *m x n* matrix *H* in the measurement equation (2) relates the state to the measurement z_k. In practice *H* might change with each time step or measurement, but here we assume it is constant.

A Kalman filter combines all available measurement data, plus prior knowledge about the system and measuring devices, to produce an estimate of the desired variables in such a manner that the error is minimized statistically.

3. Phases in Kalman Filter

The Kalman filter estimates a process by making use of two phases[1] . The filter estimates the process state at some time and then obtains feedback in the form of (noisy) measurements. These two phases are "time update" and "measurement update" respectively. The time update phase is responsible for projecting forward (in time) the current state and error covariance estimates to obtain the *a priori* estimates for the next time step. The measurement update phase is responsible for the feedback—i.e. for incorporating a new measurement into the *a priori* estimate to obtain an improved *a posteriori* estimate.

The time update phase can also be referred as predictor *phase*, while the measurement update phase can be thought of as *corrector* phase. Thus in other words, filter operates as a predictor- corrector Algorithm as shown in figure

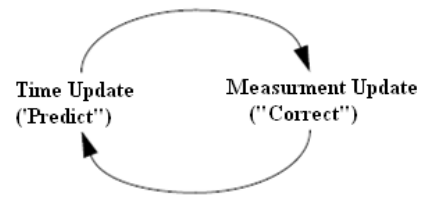

Figure1. Discrete Kalman Filter cycle

The *n x n* matrix *A* in the difference equation of Table1 , relates the state at the previous time step *k-1* to the state at the current step *k* , in the absence of either a driving function or process noise. In practice *A* might change with each time step, but here it is assumed to be constant. The matrix *B* relates the optional control input $u \in R^l$ to the state *x*.

Table 1. Time Update Phase

$$\boxed{\begin{aligned} \hat{x}_k^- &= A\hat{x}_{k-1} + Bu_{k-1} \\ P_k^- &= AP_{k-1}P^T + Q \end{aligned}}$$

Measurement update phase proceeds in three phases. The first task during the measurement update is to compute the Kalman gain K_k.The next step is to actually measure the process to obtain z_k, and then to generate an *a posteriori* state estimate by incorporating the measurement. The final step is to obtain an *a posteriori* error covariance estimate as shown in Table 2.

Table 2. Measurement Update Phase

$$\boxed{\begin{aligned} K_k &= P_k^- H^T (HP_k^- H^T + R)^{-1} \\ \hat{x}_k &= \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-) \\ P_k &= (I - K_k H)P_k^- \end{aligned}}$$

After each time and measurement update pair, the process is repeated with the previous *a posteriori* estimates used to project or predict the new *a priori* estimate. Kalman filter finds application in various engineering problems due to its robust nature and simpler implementation. The Kalman filter is implementable in the form of an algorithm for a digital computer, which may be slower but is more accurate than analog filters. The Kalman filter provides the necessary information for mathematically sound, statistically based decision methods for detecting and rejecting anomalous measurements. The Kalman filter requires information only from the previous state , updated for each iteration, older data can be discarded. This saves computational complexity and storage. Both the prediction and correction uses efficient matrix operations on the mean and covariance. This makes filter efficient. Kalman filter produces besides the optimal estimate, the covariance matrix which is an important estimate for the accuracy of system.

4. Target tracking by Kalman Filter in Wireless Sensor Networks

Wireless sensor network (WSN) is a collection of physically distributed sensing devices that can communicate through a shared wireless channel. Sensors can be deployed, for example, to detect the presence of a contaminant in a water reservoir, or to track the position of a moving target [3]. Sensor nodes in Wireless Sensor Network (WSN) need to communicate with other sensor nodes and / or a fusion centre in order to accomplish target tracking task. The limited onboard energy and wireless bandwidth are critical issues in many WSNs[4]. Kalman filter is a linear optimal filtering approach; to address the problem when system dynamics become nonlinear, researchers developed sub-optimal extensions of Kalman filter, two popular versions are EKF (extended Kalman filter) and UKF (Unscented Kalman filter).

Target tracking is a path/trajectory prediction method for environmental targets such as vehicles, missiles, animals, hostiles, or even unidentified objects. In order to make this prediction, sensors measure and record target characteristics such as speed, past trajectory, acceleration, etc. However, the prediction accuracy is only as reliable as the measured data and this data is frequently noisy[2]. A Kalman filter combines all available measurement data, plus prior knowledge about the system and measuring devices, to produce an estimate of the desired variables in such a manner that the error is minimized statistically.

Figure 2 shows the application of Kalman filter to improve target tracking results. Actual path followed by target is indicated by a solid bold line. Path depicted by sensor modules is indicated by numbered circles. Sample d and f are noisy samples. An interpolated result by using these samples is indicated by a dashed line i.e target predicted path. This predicted path is unfiltered path. Kalman filter improves the prediction results as indicated by thin solid line (i.e filtered path).

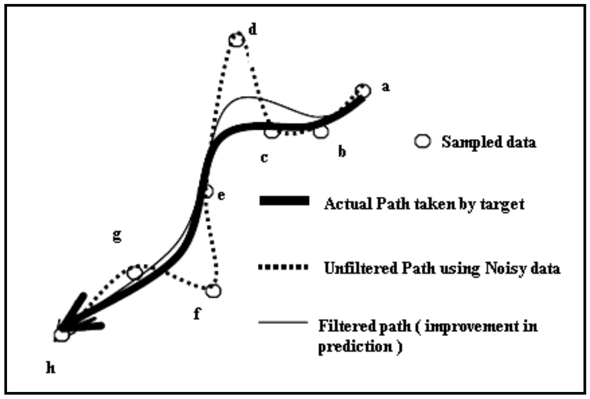

Figure 2. Target tracking by Kalman filter in WSN

5. FPGA Implementation

The rapidly increasing popularity of WSNs has placed increased computational demands upon these systems, due to increasingly complex operating environments and enhanced data-sensing technology. Whereas introducing more powerful microprocessors into sensor nodes addresses these demands, sensor nodes do not contain sufficient energy reserves to support these microprocessors. FPGAs offer increased performance compared to microprocessors and increased flexibility compared to ASICs, while maintaining low power consumption. FPGA technology is indispensable wherever long-term availability or harsh industrial environments are involved[5] . Some of advantages of using FPGA platform are listed below:

a. Ability to operate in Harsh Environments
One of the most critical requirements is a qualified operational temperature between -40 and +85°C. Many FPGAs are defined for -40 to +85°C operation temperature. It conforms to EN, CECC or IEC standards, is manufactured according to ISO 9000 and a second source is also available.

b. Long-Term Availability
Another important aspect is long-term availability. The advantage of FPGAs and their nearly unlimited

availability lies in the fact that — even if the device migrates to the next generation — the code remains unchanged. This is in accordance with norms like the EN 50155 which prescribes that customized parts like FPGAs must be documented to allow reproduction and that the documentation and the source code must be handed out to the customer.

c. Time to market
FPGA technology offers flexibility and rapid prototyping capabilities in the face of increased time-to-market concerns. It is possible to test an idea or concept and verify it in hardware without going through the long fabrication process of custom ASIC design. It is possible to implement incremental changes and iterate on an FPGA design within hours instead of weeks.

d. Performance
Taking advantage of hardware parallelism, FPGAs exceed the computing power of digital signal processors (DSPs) by breaking the paradigm of sequential execution and accomplishing more per clock cycle This level of performance is ideal for building very fast single channel systems or slower rate systems that comprise hundreds of channels.

6. Design Methodology and Results

For FPGA based design , speed and area are the major performance criteria and they places a separate, often conflicting, constraint on the design of a filter. Various performance criteria are individually investigated and analyzed; and the interaction of these criteria is examined to develop both a qualitative and a quantitative understanding of the various design tradeoffs. Most of the work done on Kalman filter developed for wireless sensor networks is done on microprocessor based systems. There is urgent need to optimize both speed and area factors simultaneously on FPGA platform to meet the ultra-high digital signal processing (DSP) bandwidth and lower system-cost requirements of wireless sensor networks. The main objective is to design and implement Kalman filter on XILINX FPGA to provide speed enhancement and area efficiency by efficient utilization of resources available on target device; In order to provide high performance solution for filtering applications.

This objective of simultaneous speed and area optimization is met by using partially serial and partially parallel approach. Partially serial design on FPGA results in area efficiency and parallelism leads to speed efficiency. Design methodology proceeds as indicated in Figure 3.

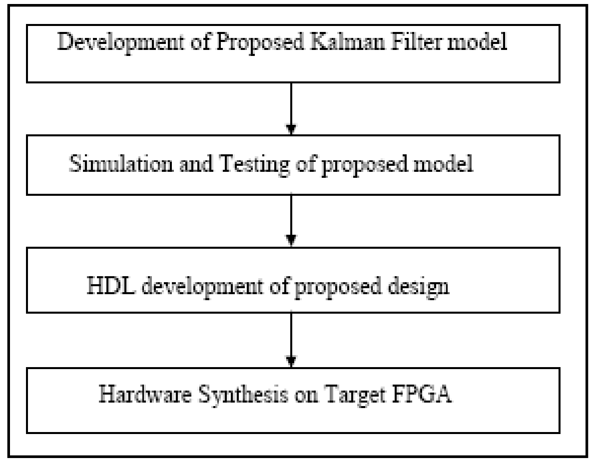

Figure3. Design Methodology

This work begin with understanding of present work on Kalman filter developed on various platforms followed by development of proposed model. Simulation and verification of the proposed model will be carried out. This is followed by HDL development of proposed model, followed by Hardware synthesis on Target FPGA.

Our experimental setup used a Xilinx Virtex-4 FPGA. Kalman filter modules reach 87 MHz clock frequency. The elimination of dedicated multiplier units provides a 47.2% reduction in device usage with a 24% performance penalty for this tradeoff.

7. Conclusion and Future Work

Wireless Sensor Networks requires real time processing of data. Wide spread use of this technology leads to increased computational burden which can be met by using complex microprocessors. But limited power supply is also one of the key constraints. All these issues demands on-board processing solutions. We have presented a design methodology for exploiting FPGAs for target tracking applications in WSN. The low device utilization of our methodology makes our methodology highly amenable to small devices.

Future work may be development of variants our design for situation based reconfiguration based upon for e.g type of target etc.

Acknowledgements

The authors would like to thank Dr. Parijat De, Director, NITTTR, Chandigarh for constant encouragement and support during this research work. The authors would also like to express their sincere thanks and deep sense of gratitude to Dr. S. Chatterji, Professor and Head, Electronics & Communication Department and Dr. S.S.Pattnaik, Professor and Head, ETV Department, NITTTR, Chandigarh for their constant guidance and helpful suggestions throughout this research work**.**

References

[1] Mohinder singh Grewal, Angus P. Andrews, "Kalman filtering : Theory and Practice using Matlab" , John Wiley & Sons, 2007,pp 8-32

[2] Rajeev Shorey, A. Ananda, Mun Choon Chan Wei Tsang Ooi "Mobile, Wireless and Sensor Networks: Technology , applications and future directions "IEEE press , pp173- 182, 2006

[3] Alejandro Ribeiro , Ioannis D. Schizas, Stergios I. "Kalman filtering in Wireless Sensor Networks " IEEE Control System Magazine , pp 66-87, April 2010

[4] Jianyong Lin, Lihua Xie "Target tracking in wireless sensor networks using compressed Kalman filter" International Journal of Sensor Networks. Volume 6 , Issue 3/4 , pp 251- 262 , November 2009

[5] White Paper: Comparing and Contrasting FPGA and Microprocessor System Design and Development, WP213 (v1.1) July 21, 2004

[6] Mahdy Nabaee, Ali Pooyafard, Ali Olfat " Enhanced Object Tracking with Received Signal Strength using Kalman Filter in Sensor Networks" 2008 IEEE International Symposium on Telecommunications, pp 318-324.