# Kalman Filter Embedded in FPGA to Improve Tracking Performance in Ballistic Rockets

**Conference Paper** · April 2013

DOI: 10.1109/UKSim.2013.149



**5 authors**, including:

Roberto Célio Limão de Oliveira
Federal University of Pará



Ernesto Ferreira
Universidade Federal do Maranhão

2013 UKSim 15th International Conference on Computer Modelling and Simulation

# Kalman Filter Embedded in FPGA to Improve Tracking Performance in Ballistic Rockets

João V. Fonseca, Roberto. C. L. Oliveira, José A. P. Abreu, Ernesto Ferreira and Madson Machado
*Laboratory of Systems Embedded and Smart Control*
*Federal University of Maranhão*
*São Luís, Maranhão, +559833018273*
*E-mail: vianafonseca@gmail.com*

***Abstract*— This research project deals with the application of dynamic modeling methods for prediction and correction (filtering) of state variables in the calculation of the position and speed training rockets launched at Alcântara Launch Center. Initially, an approach is made of the means involved in launching and tracking of rockets, with the purpose of obtaining an understanding of the acquisition of radar signals, which are input of the filter. Then it made an approach of mathematical treatment of the filter dynamic model to obtain the equations used in the prediction and filtering of the rocket trajectory. Finally is made the application of Kalman filter algorithm in LabVIEW (Laboratory Virtual Instrumentation Engineering Workbench) FPGA (Field Programmable Gate Array) to estimate the position and velocity of the rocket. The main contribution of this work is to obtain a gain in processing velocity of Kalman filtering using parallelism at the hardware level, implementing a reconfigurable FPGA architecture, ensuring a platform fast enough for radars with high precision and good tracking capability of rockets.**

***Keywords-Kalman filter; FPGA; ballistic rocket; state estimation; radar tracking; stochastic models.***

## I. Introduction

This paper proposes a hardware architecture based on FPGA for the problem of filtering of radar signals for rockets tracking. The architecture consists of hardware modules. The idea is to design an embedded system on FPGA and show that its performance is superior in at least one order of magnitude with respect to executions in software. This performance gain to be achieved by exploiting the parallelism in hardware [1] [2]. The complexity of implementing this work can be considered high, since it involves a large amount of arithmetic operations. Additionally, we are being created communication interfaces software and hardware for reading of tracking data. We intend demonstrate that, reconfigurable device type, FPGA is a viable platform for implementation of filtering systems used in rockets tracking.

The remainder of this paper is organized as follows: Section II describes the Kalman filter algorithm and its behavior during rocket tracking. Section III describes the importance of the application of FPGA technology in Kalman filtering. Section IV describes the Kalman filter design in LabVIEW FPGA. Section V discusses the results. Section VI summarizes our work methodology and results.

## II. System Model Development

The first activity is to understand the system dynamic response through creation of a system model.

### A. RTI Rocket

Rocket Training Intermediate (RTI) corresponds to a rocket to the operational training Launch Center, instrumented with S-band telemetry, C-band transponder and flight termination. The RTI has a total length of 5.5m, a total weight of 490kgf, burn time of 4 seconds and it has a placeholder for scientific experiments of 30kgf. This rocket has apogee above 60km and range impact around 85km. The RTI have three independent parts: Rocket Motor, Payload and Flight Termination, as seen in Figure 1.

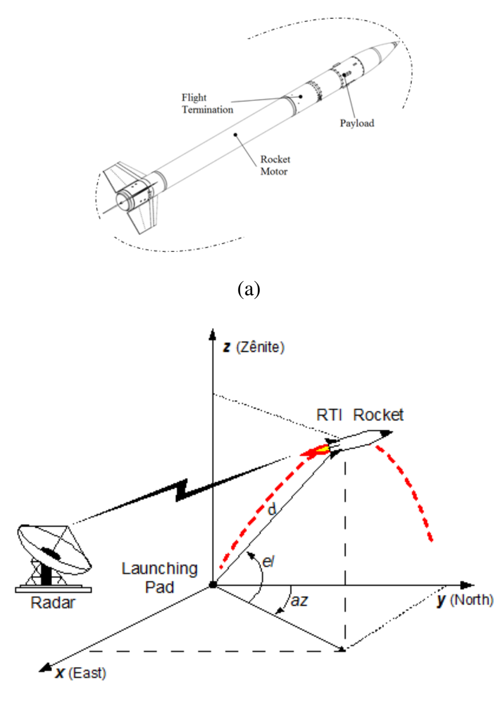


Figure 1. (a) RTI Rocket, (b) Parameters for locating a rocket by radar

978-0-7695-4994-1/13 $26.00 © 2013 IEEE
DOI 10.1109/UKSim.2013.149

The Rocket Motor is composed of three parts, namely:

- **Uploaded Tube Motor** - consists of a steel tube with cold drawn Electrophoretic painting, with thermal protection and Composite Propellant.
- **Nozzle with Gables, Fairings, Claw and Counter Weight** - composed of machined metal parts and painted inserts fitted with ablative materials and graphite.
- **Ignitor** - consisting of metal parts machined and painted. Loaded with composite propellants with electric start by Squib.

The payload is the part of the rocket which leads to S-band telemetry, telemetry antenna, C-band radar transponder, transponder antenna as well as a placeholder to take up to 30 kgf of scientific experiments.

The Flight Termination is composed of three main sections, namely:

- Housing cold drawn steel.
- Load Destruction consist of a steel housing with cone copper loading and RDX "explosive train" electric drive.
- Retractable Claw, made of special steel.

### *B. Rockets Tracking System*

A reliable rocket tracking system consists of two ground radar, two local computers running subroutines rocket launch (signal filtering, coordinate conversion, and so on) and other central computer which determines which radar has the best signal trajectory of the rocket. The central computer selects the better signal and sends it to the graphics system, where this trajectory is viewed by the team of flight safety. This system can be seen in Figure 2.

The system operation occurs as follows:

1) The position of the noisy signals are measured by radar sensor in spherical coordinates, to a fixed frequency;
2) The position coordinates are measured with respect to the radar, after suffering a change of reference to the launch pad in Cartesian form;
3) Are provided filtered position and estimated velocity of the rocket.

In general, for rocket tracking by radar, all proposed methodologies [3] [4], follow an algorithmic approach. The Kalman filter for the linear estimation algorithm is typically more complex and accurate in rockets tracking. However, the accuracy and complexity of this algorithm requires a high processing speed and a high degree of parallelism (in terms of hardware and software) that is beyond the computational capacity available.

The computational load for calculating algorithms based on Kalman filter becomes significant enough such that a radar computer hardly meet the performance requirements mentioned above, since the junction hardware/software does not provide a sufficiently rapid platform.

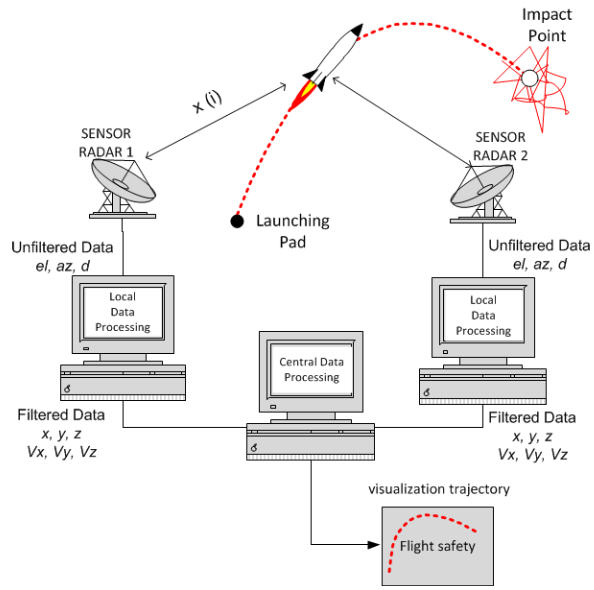


Figure 2. Radar Signal Processing System for Rockets Tracking

In some cases simpler algorithms have been employed, however, has low accuracy and capacity loss of tracking.

### *C. Kalman Filter*

The Kalman filter was first proposed in 1960, when Rudolf E. Kalman published his famous paper describing a recursive solution to the problem of filtering data in a discrete linear system. Given some initial values, we can predict and adjust the parameters of the system model through each new measurement, obtaining the estimate of the error in each update. Its performance recursively and computational progress made the Kalman filter had a wide field of applications, specially in aerospace as the rocket trajectory analysis [5] [6].

The Kalman Filter is a powerful linear estimator and has been used in models of dynamic systems [7]. The Kalman filter equations can be solved numerically using a recursive structure type that outputs depend only on the current state of the inputs and outputs earlier. This form of the Kalman filter approach is very interesting for implementation in hardware (FPGA). Equations models of the process and measurement that have been implemented in this article are:

$$X(k+1) = AX(k) + w(k), \tag{1}$$

$$Y(k) = CX(k) + v(k), \tag{2}$$

where
$X(k) = [X_1(k), X_2(k)]^T$,
$Y(k) = [Y_1(k)]$,

$w(k) = [0, W_1(k)]^T, \quad v(k) = [V_1(k)],$

$A = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 \end{bmatrix}.$

From Equation (1) we have that $X_1(k)$ is the altitude of the rocket relative to the launch pad, $X_2(k)$ is the speed of the rocket and $w(k)$ represents the process noise. From Equation (2) we have that $y_1(k)$ is the position measurement by the sensor and $v(k)$ represents the measurement noise.

The equations for the Kalman filter are divided into two groups: time update equations and measurement update equations. The time update equations ("prediction") are as follows:

$$P1(k/k-1) = AP(k-1/k-1)A^T + G(k-1), \quad (3)$$

$$\widehat{X}1(k/k-1) = A\widehat{X}(k-1/k-1), \quad (4)$$

and measurement update equations ("correction") are as follows:

$$K(k) = P1(k/k-1)C^T[CP1(k/k-1)C^T + R(k)]^{-1}, \quad (5)$$

$$\widehat{X}(k/k) = \widehat{X}1(k/k-1) + K(k)[Y(k) - \widehat{Y}(k)], \quad (6)$$

$$P(k/k-1) = P1(k/k-1) - K(k)CP1(k/k-1). \quad (7)$$

$P1(k/k-1)$ is the a priori error covariance estimate, $\widehat{X}1(k/k-1)$ is the priori state estimate, $\widehat{Y}(k)$ is the output estimate, $K(k)$ is the Kalman gain, $\widehat{X}(k/k)$ is the posteriori state estimate, $P(k/k)$ is the posteriori error covariance estimate, $Q(k)$ is the system noise covariance matrix, $R(k)$ is the measurement noise covariance.

The Kalman filter equations can then be rewritten as follows:

$\widehat{Y}(k/k) = C\widehat{X}1(k/k-1),$

$K(k) = \begin{bmatrix} K_{11}(k) \\ K_{21}(k) \end{bmatrix},$

$P(k/k) = \begin{bmatrix} P_{11}(k) & P_{12}(k) \\ P_{21}(k) & P_{22}(k) \end{bmatrix},$

$P1(k/k) = \begin{bmatrix} P1_{11}(k) & P1_{12}(k) \\ P1_{21}(k) & P1_{22}(k) \end{bmatrix},$

$Q(k/k) = \begin{bmatrix} 0 & 0 \\ 0 & \sigma_1^2(k) \end{bmatrix}, \; R(k) = \sigma_p^2(k),$

where

$P1_{11} = P_{11} + TP_{12},$
$P1_{12} = P_{12} + TP_{22},$
$P1_{21} = P_{21} + TP_{22},$
$P1_{22} = P_{22} + \sigma_1^2,$
$X1_1 = X_1 + TX_2,$
$X1_2 = X_2,$
$\widehat{Y}_1 = X1_1,$
$K_{11} = P1_{11}/(P1_{11} + \sigma_p^2),$
$K_{21} = P1_{21}/(P1_{11} + \sigma_p^2),$
$X_1 = X1_1 + K_{11}(Y_1 - \widehat{Y}_1),$
$X_2 = X1_2 + K_{21}(Y_1 - \widehat{Y}_1),$
$P_{11} = P1_{11} - P1_{11}K_{11},$
$P_{12} = P1_{12} - P1_{12}K_{11},$
$P_{21} = P1_{21} - P1_{11}K_{21},$
$P_{22} = P1_{22} - P1_{12}K_{21},$

## III. LabVIEW FPGA

In many research applications has increased the need for methods of signal processing faster and more accessible. The increased complexity of the problem involves a higher cost of the system. The designer's solution remains the achievement of maximum performance and flexibility for cost [8]. Compared with DSP and ASIC, the Field Programmable Gate Array (FPGA) provides an attractive platform for the implementation of many complex algorithms in real time with higher performance. The advantage of using FPGA instead of using DSP is the hardware parallelism [9] instead of sequential execution. Multiple tasks can be performed in FPGA using a dedicated set of ports. The FPGA also allows the designer the flexibility to use reconfigurable logic to form computing structures corresponding to the desired application [10]. In this paper we use a graphical programming language to implement and test the Kalman filter used for filtering of radar signals. The next step is embed the Kalman filter in FPGA through the LabVIEW FPGA module, as seen in Figure 3. LabVIEW FPGA is the portion of LabVIEW that allows targeting of FPGAs. LabVIEW FPGA supports hardware design at different levels of abstraction, [11].

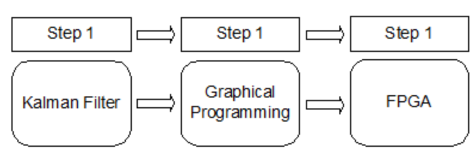


Figure 3. Kalman filter Embedded in FPGA system

We start the design process at high level, which allows us to rapidly prototype the system. So we can use the LabVIEW FPGA module to successfully implement projects of high-speed hardware [12].

## IV. Kalman Filter Design in LabVIEW FPGA

The Kalman filter equations for rockets tracking were implemented in two parts in LabVIEW, using two subVIs (sub-Virtual Instruments), with blocks that perform basic mathematical operations of addition, subtraction, multiplication and division. In this project we have used nine adders, six subtractors, ten multipliers and two dividers. The platform used is the Xilinx Spartan 3E board with LabVIEW FPGA with clock at 50 MHz. The data of the trajectory the rocket will be read through the serial port of the FPGA and the filtered data will be sent to a video terminal to visualize the actual and estimated trajectories. The following we present the block diagrams implemented on the board Spartan-3E.

The subVI, Figure 4, shows the block diagram of the first part of the Kalman filter that was implemented in LabVIEW.

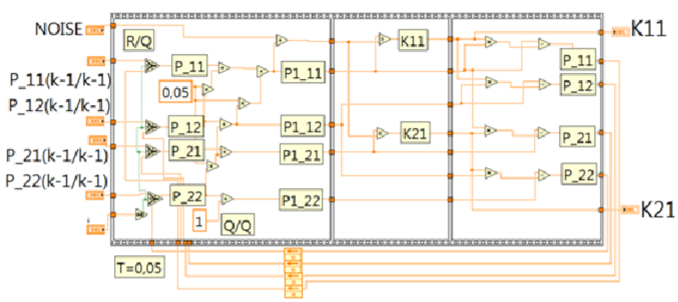


Figure 4. First Part of Kalman Filter

This subVI receives as input parameters the covariance matrix initialization of the error estimate and the ratio of the covariance of the measurement noise and the covariance error process. The outputs are the Kalman Filter gains that will be part of the input parameters of the subVI that is responsible for calculating the estimated position, estimated speed and error estimation. In addition to the gains that subVI takes as input the actual measurement and the initial estimates of position and velocity.

Figure 5 illustrates this subVI with its input parameters and output.

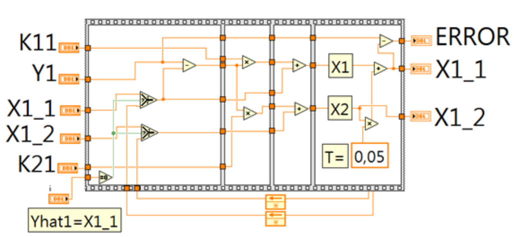


Figure 5. Second Part of Kalman Filter

Figure 6 have the complete block diagram of the Kalman filter embedded in FPGA.

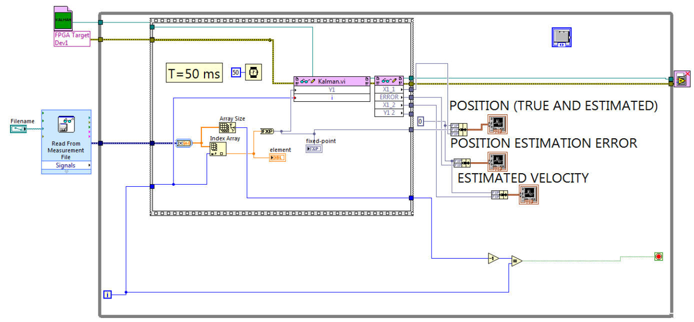


Figure 6. Complete Diagram Blocks

The input data of the Kalman filter are real data measured during the tracking of an RTI. The data initialization of the filter are: $R = 6$, $Q = 2$, $\widehat{Y} = 0$, $P = \begin{bmatrix} 100 & 0; 0 & 100 \end{bmatrix}$, and $T = 50ms$.

Figure 7 shows the real and estimated position by the Kalman Filter.

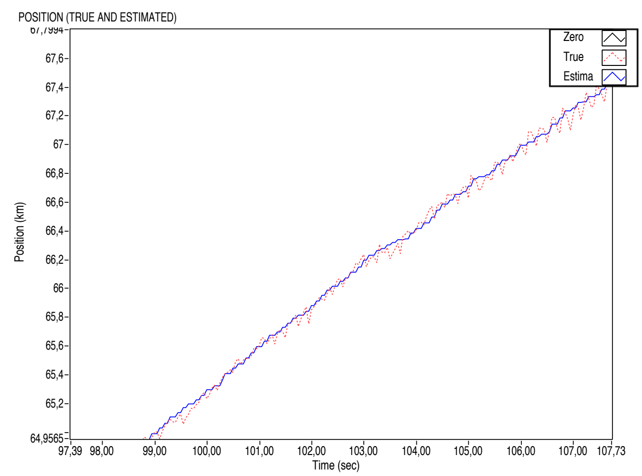


Figure 7. Real and Estimated Position.

Figures 8 and 9 illustrate the estimated speed and error of the rocket, respectively. Through the graphs shown we can see that the Kalman filter Embedded in FPGA had a very good performance, improving tracking performance in ballistic rockets

## V. Results

In the final result of this project, we have used real world data of the trajectory of launching a RTI. We propose the implementation of the Kalman filter in LabVIEW FPGA using the virtual instruments presented for radar signals filtering for rockets tracking. The results of our investigation demonstrate that the performance of the Kalman filter embedded in FPGA is superior to personal computers mainly due to the exploitation of the parallelism of the FPGA hardware.

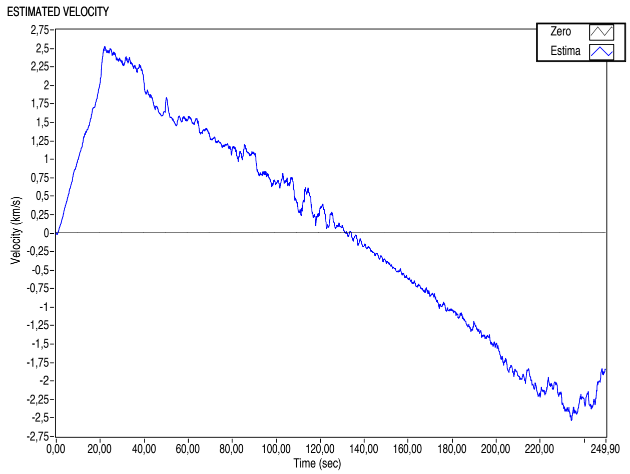


Figure 8. Estimated Velocity

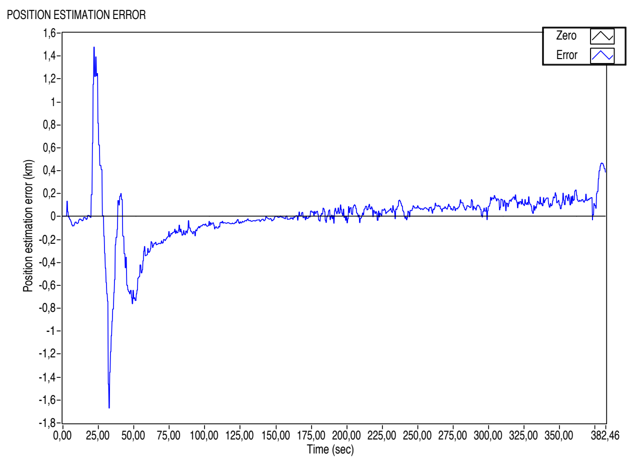


Figure 9. Estimation Error

## VI. Conclusion

In conclusion, we have created a successful FPGA based Kalman Filter. Kalman filter is a powerful tool to track moving rockets. Several approaches were implemented to achieve a better performance for the Kalman operation. The computational load becomes larger and larger, limiting the performance of radar. In [1], a hardware parallel architecture was proposed and implemented in FPGA. We propose using FPGA based tracking filter to replace the traditional software based one. By using FPGAs, the performance (velocity) can be improved two to three orders of magnitude over other approaches.

## References

[1] C. R. Lee and Z. Salcic, "A fully-hardware-type maximum-parallel architecture for kalman tracking filter in fpgas," *International Conference on Information, Communications and Signal Processing*, 1997.

[2] Z. Merhi, M. Ghantous, M. Elgamel, M. Bayoumi, and A. E. Desouki, "A fully-pipelined parallel architecture for kalman tracking filter," *IEEE - The International Workshop on Computer Architecture for Machine Perception and Sensing*, 2006.

[3] S. M. Bozic, *Digital and Kalman Filtering*. Edward Arnold, 1984.

[4] X. R. Li and V. P. Jilkov, "Survey of maneuvering target tracking," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1333–1364, Oct 2003.

[5] J. A. P. Abreu, J. V. F. Neto, and R. C. L. Oliveira, "Ballistic rockets tracking: Kalman versus $\alpha$-$\beta$-$\gamma$ filters," *IEEE Computer Society*, 2011.

[6] J. V. F. Neto, J. A. Farid, and J. A. P. Abreu, "Qr-duality tuning of standard kalman filter oriented to rocket velocity indirect," *International Conference on Computer Modelling and Simulation, IEEE Computer Society*, 2010.

[7] F. Silva, S. Y. C. Catunda, J. Neto, and A. van Haandel, "Dissolved oxygen pwm control and oxygen uptake rate estimation using kalman filter in activated sludge systems," *Instrumentation and Measurement Technology Conference (I2MTC), IEEE*, 2010.

[8] N. Devineni and I. M. S. Panahi, "Analysis and real time implementation of 2-channel adaptive speech enhancement using labview fpga," *IEEE International Conference on Electro/Information Technology (EIT)*, 2011.

[9] F. Vahid, S. Narayan, J. Gong, and D. D. Gajski, *Specification and Design of Embedded Systems*. Prentice Hall, 1994.

[10] Z. Chen, Z. Sun, and W. Wang, "Design and implementation of kalman filter," *IET International Conference on Communication Technology and Application*, 2011.

[11] M. B. McMickell, P. J. Tanzillo, T. Kreider, and K. Ilic, "Rapid development of space applications with responsive digital electronics board and labview fpga," *NASA/ESA Conference on Adaptive Hardware and Systems (AHS)*, 2010.

[12] G. Wang, T. N. Tran, and H. A. Andrade, "A graphical programming environment for design and fpga-based hardware," *International Conference on Field-Programmable Technology (FPT)*, 2010.