Air defense missile detonation delay control Based on FPGA/DSP

Lian-zheng Zhang, Pan-long Wu and Xin-yu Zhang
Department of Automation
Nanjing University of Science and Technology
No. 200, Xiao Ling Wei Street, Nanjing 210094
PEOPLE'S REPUBLIC OF CHINA
freedomzlz@126.com, plwu@163.com, 347692173@qq.com

Abstract: - This paper presents the hardware design of a real time tracking system based on DSP/FPGA regarding the research and air defense missile detonation delay. It is a new trend of missile technology using guidance integrated fuze (GIF) technology to realize optimal burst delay control algorithm. The optimal burst delay control algorithm includes the estimation of time-to-go and miss distance. The paper studied the application of second order debiased converted measurement Kalman (SDCMKF) filer in parameters estimation of burst delay control. In this tracking system, the FPGA is used as a floating point co-processor of the fixed point DSP, and the large amount of calculation of the second order debiased converted measurement Kalman Filter (SDCMKF) algorithm is realized in FPGA. DSP is in charge of the scheduling of the total tracking algorithm and the control of the data stream, which resolves the problem of the concurrency and real time in the realization of the single DSP scheme. The designed tracking system ensures the accuracy of the data processing as well. Simulation results indicate that the application of this algorithm improved the estimation accuracy of burst delay control. The estimation error reduces gradually and tends to be stable with the distance between target and missile shortened.

Key-Words: - Target tracking, air defense missile detonation delay, burst delay control, SDCMKF, coprocessor, FPGA, DSP.

1 Introduction

To improve the weapon system effectiveness in encounter with future air threat likely to have a high level of agility and sophisticated countermeasure, performance requirements of the fuze become more and more stringent, the guidance integrated fuzing (GIF) is an emerging trend in missile technology [1-2]. GIF is a kind of fuze technology based on the new concept and new theory of fuze, which aims for getting more encountering information about missile and target and improving the efficiency of matching of fuze with warhead. The speed of the final engagement and the difficulty in actually hitting maneuvering target require the use of proximity fuzes that are capable of detecting the target, detonating the warhead at the proper time to inflict maximum damage and extending the kill envelope. With increase in warhead directionality, target speed and decrease in target size, performance requirements of the fuze become more and more stringent. The GIF technology will be widely used and developed in the future [3].

The complexity of detonation delay control algorithm lies in the three-dimensional space rendezvous of missile and target, and is mainly reflected in the following two aspects: (1) The encounter rendezvous conditions are complex; (2) The independent localization capability of fuze is not good [1].

In the real system, the radar measurements in a spherical coordinate system, whereas, the target's state equation is generally established in the Cartesian coordinate system, the measurement equation in spherical coordinates to Cartesian coordinates is a nonlinear equation. State and measurement equations are often not in the same coordinate system of linear equations. General method is the use of extended Kalman filtering (EKF). Another method is to use the debiased converted measurement Kalman filter [2-3] (DCMKF). The standard DCMKF is commonly derived from a first order Taylor expansion of the state dynamics and measurement model. The truncation of Taylor series covers the bias of converted measurement error, which may lead to linearization error and divergence because of dealing with maneuvering targets as a type of nonlinear actual systems. However this problem can be avoided to some extent by using the second order term of Taylor series. In this paper, the second order debiased converted measurement Kalman (SDCM-KF) filer is applied to calculate the time-to-go and miss distance of burst delay control.

In the traditional software system design, SDCMKF is usually realized by digital signal processor (DSP), which would be restricted by the serial instruction stream due to the complex computations of SDCMKF and unable to meet the high-speed real-time signal processing needs. However, using the hardware parallel architecture feature of field programmable gate array (FPGA) to realize floating point of SDCMKF can resolve the problem of high precision and real time[18, 21-23].

This paper based on FPGA realization of floating point SDCMKF can resolve the problem of high precision and real time in air defense missile detonation delay system. Compared Matlab and Quartus IIsimulation data, both of the simulation results are consistent, this ensure that the computational accuracy. This paper mainly studies the use of FPGA to realize floating point SDCMKF and air defense missile detonation delay effective method.

2 General system design

In this paper, we adopt DSP TMS320VC5509A chip as core processor of the radar target tracking system. This fixed point DSP is responsible for the scheduling of the whole tracking algorithm and the control of data stream. The FPGA EP3C120F484C8N chip is adopted as floating point co-processor of fixed point DSP. DSP receives radar measurements values then transmits to FPGA. FPGA informs DSP to retrieve the filtered system state values when one frame data is processed.

2.1 Hardware design of the system

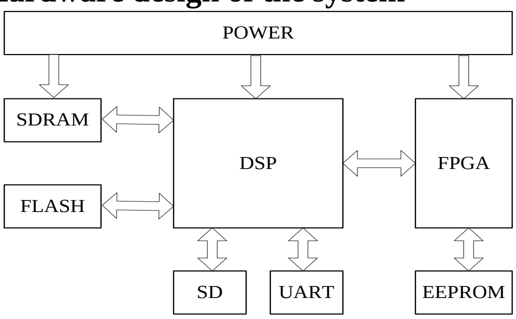

Fig.1 The structure diagram of system hardware design.

The structure diagram of system hardware design is shown in Fig.1. This system is composed of DSP subsystem and FPGA subsystem. DSP subsystem consists of DSP, FLASH, synchronous dynamic random access memory (SDRAM), secure digital (SD) card and some peripheral interface circuits, FLASH is used to store program. SDRAM is used to buffer pre-filtered and post-filtered data. SD card is used to store both filtered and unfiltered data. FPGA subsystem consists of FPGA and electrically erasable programmable read-only memory (EEPROM). FPGA is used to realize the complicated DCMKF algorithm, EEPROM is used to store FPGA program.

Fig.2 shows the hardware connection diagram between DSP and FPGA. External memory interface (EMIF) connects DSP and FPGA. Wherein, $\overline{\text{CE}}$ is chip electing signal, $\overline{\text{AOE}}$ is asynchronous output enable signal, $\overline{\text{AWE}}$ is asynchronous writing electing signal, $\overline{\text{ARE}}$ is asynchronous reading enable signal, $\overline{\text{INT}[4:0]}$ is interrupt enable signal, D[7:0] is data bus signal and A[7:0] is address bus signal. Wherein, IO(D[7:0]) is data bus signal , IO($\overline{\text{INT}}$) is interrupt enable signal.

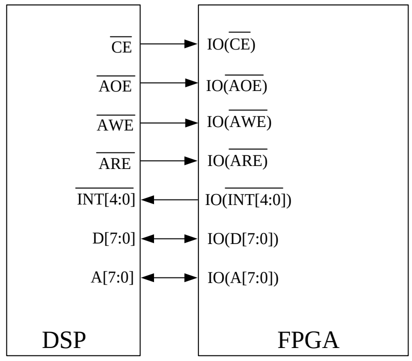

Fig.2 The hardware connection diagram between DSP and FPGA.

Fig.3 shows the sequence diagram of writing operation between DSP and FPGA. $\overline{\text{AOE}}$ and $\overline{\text{ARE}}$ are set high when writing.

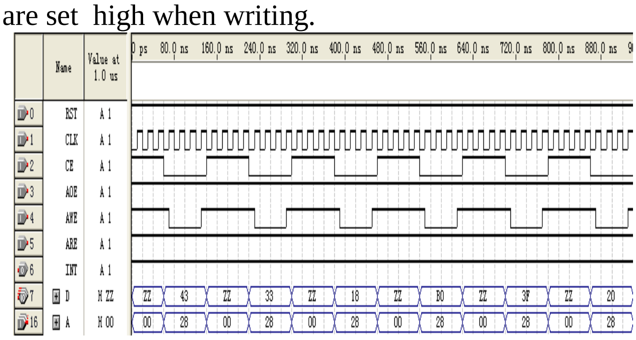

Fig.3 The sequence diagram of writing operation between DSP and FPGA.

2.2 Software design of the system

In software design of the system, intensive data and highly repetitive algorithm are processed by FPGA, while low repetitive algorithm is processed by DSP. In this radar target tracking system, initial value of SDCMKF needs to be calculated only once. Therefore, the initial value of SDCMKF is calculated by DSP then transmits to FPGA. However, the subsequent each frame data needs to be filtered and consumes a large number of processing time. Therefore, we choose FPGA to complete SDCMKF algorithm. Fig.4 shows the software block diagram of the radar target tracking system. The first three frames data of correctly received from radar are calculated for initial values of SDCMKF. After the system receives correctly the fourth frame data, DSP transmits the calculated initial values and the measurement value to FPGA for SDCMKF filtering. An interrupt signal is transmitted to DSP to retrieve the filtered target state values after FPGA processing each frame data.

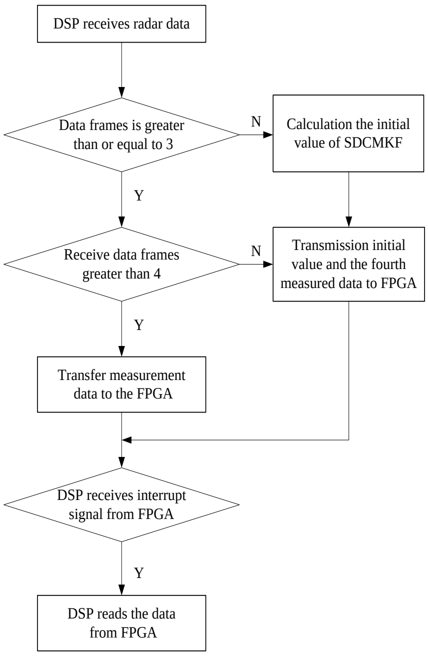

Fig.4 Software design of the system

3 The optimal detonation delay control

Mathematical models are established for optimal detonation delay control under arbitrary encounter conditions in missile body coordinate. The optimal detonation delay algorithm is formulated by the distance and angle information of the target which is provided by the radar seeker. For the convenience of analysis, the center of the seeker coincides with the missile vertex in this paper. The origin the missile body coordinates is located at the missile head. Target head is the vulnerable part and is the tracking point of missile. The encounter of missile and target is shown in Fig. 5 [10].

The OZ axis lies along the missile's longitudinal axis. The target is located at point T, $\overrightarrow{TC}$ is the tangential direction of the target trajectory, V_R is the relative velocity between the missile and target. t_{go} is the time-to-go for the missile to intercept the target, which is also the target flies along the relative trajectory from the current point to point C, $|OC|$ is the miss distance. The projection of point T on XOY plane is B, r is the radial distance of the target, θ is elevation angle of the target, β is azimuth angle of the target. When the encounter end, or the distance of the missile and target is small, the parameter errors caused by motor can be ignored, so it can be regarded as uniform motion [2].

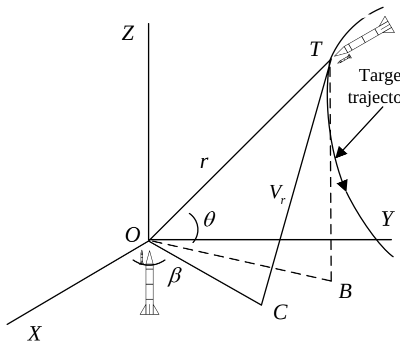

Fig.5 The encounter of missile and target

The target relative position vector is

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} r\cos\theta\cos\beta \\ r\cos\theta\sin\beta \\ r\sin\theta \end{bmatrix} \tag{1}$$

The relative velocity vector is

$$\vec{V}_r = -\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = -\begin{bmatrix} \dot{r}\cos\theta\cos\beta - \dot{\theta}r\sin\theta\cos\beta - \dot{\beta}r\cos\theta\sin\beta \\ \dot{r}\cos\theta\sin\beta - \dot{\theta}r\sin\theta\sin\beta + \dot{\beta}r\cos\theta\cos\beta \\ \dot{r}\sin\theta + \dot{\theta}r\cos\theta \end{bmatrix} \tag{2}$$

The time-to-go t_{go} is

$$t_{go} = -\frac{z}{\dot{z}} = \frac{r\sin\theta}{\dot{r}\sin\theta + r\dot{\theta}\cos\theta} \tag{3}$$

The miss distance $|OC|$ is

$$|OC| = |\overrightarrow{OT} + \vec{V}_r \cdot t_{go}| = \frac{r^2\sqrt{\dot{\theta}^2 + \dot{\beta}^2\sin^2\theta\cos^2\theta}}{r\dot{\theta}\cos\theta + \dot{r}\sin\theta} \tag{4}$$

The warhead detonation delay τ is

$$\tau = t_{go} - \frac{|OC|}{V_f} \tag{5}$$

Where V_f is static flying speed of warhead fragments, It can be seen that the estimation accuracy of time-to-go and miss distance determines the damage efficiency of the warhead to the target. Even if the missile is equipped with a directional warhead, the miss orientation of the target also needs higher estimation accuracy.

In (2) ,(3)and (4), where

$$\begin{aligned} \dot{\beta} &= \frac{r\dot{z} - z\dot{r}}{r\sqrt{r^2 + z^2}} \\ \dot{\theta} &= \frac{x\dot{y} - y\dot{x}}{x^2 + y^2} \end{aligned} \tag{6}$$

4 SDCMKF principle

4.1 Analysis of Converted Measurement Errors

The measured target position (the distance r_m, the azimuth β_m and the elevation θ_m) is defined with respect to the true position (the true distance r, the true azimuth β and the true elevation θ) as [4]

$$\begin{cases} r_m = r + \tilde{r} \\ \theta_m = \theta + \tilde{\theta} \\ \beta_m = \beta + \tilde{\beta} \end{cases} \tag{7}$$

where the errors in distance $\tilde{r}$, azimuth $\tilde{\beta}$ and elevation $\tilde{\theta}$ are assumed to be independent with zero mean and standard deviation σ_r, σ_β and σ_θ respectively. These polar measurements are converted to the Cartesian coordinate measurements[14-16]

$$Z = \begin{bmatrix} x_m \\ y_m \\ z_m \end{bmatrix} = \begin{bmatrix} r_m \cos\theta_m \cos\beta_m \\ r_m \cos\theta_m \sin\beta_m \\ r_m \sin\theta_m \end{bmatrix} \tag{8}$$

From (8), we know that the converted measurements are correlated and nonlinear with respect to the polar measurements (r_m, β_m and θ_m).

If the measurement errors of r, β and θ are small and target distance is close, errors statistic approximations obtained in Cartesian coordinates are accurate. These approximations are obtained by taking the first-order terms of a Taylor series expansion for the (8) to approximate the Cartesian coordinate errors as

$$\tilde{x} = \tilde{r}\cos\theta\cos\beta - \tilde{\theta} r sin\theta cos\beta - \tilde{\beta} r\cos\theta\sin\beta \tag{9}$$

$$\tilde{y} = \tilde{r} cos\theta sin\beta - \tilde{\theta} r sin\theta sin\beta + \tilde{\beta} r cos\theta cos\beta \tag{10}$$

$$\tilde{z} = \tilde{r} sin\theta + \tilde{\theta} r cos\theta \tag{11}$$

Note that the approximation transformation errors are unbiased. However, the standard transformations are biased. The truncated high-order terms can significantly affect tracking accuracy if the cross-distance error is large. To reduce the linearization errors caused by truncated high-order terms of Taylor series expansion, we take the second-order terms of Taylor series expansion for the (8) to obtain the higher approximations of the Cartesian coordinate errors.

$$\begin{aligned} \tilde{x} = & -\frac{1}{2}(\tilde{\theta}^2 + \tilde{\beta}^2) r cos\theta cos\beta - \tilde{\theta} r sin\theta cos\beta + \tilde{r} cos\theta cos\beta \\ & - \tilde{\theta}\tilde{r} sin\theta cos\beta - \tilde{\beta} r cos\theta sin\beta + \tilde{\theta}\tilde{\beta} r sin\theta sin\beta - \tilde{\beta}\tilde{r} cos\theta sin\beta \end{aligned} \tag{12}$$

$$\begin{aligned} \tilde{y} = & -\frac{1}{2}(\tilde{\theta}^2 + \tilde{\beta}^2) r cos\theta sin\beta - \tilde{\theta} r sin\theta sin\beta + \tilde{r} cos\theta sin\beta \\ & - \tilde{\theta}\tilde{r} sin\theta sin\beta + \tilde{\beta} r cos\theta cos\beta - \tilde{\theta}\tilde{\beta} r sin\theta cos\beta + \tilde{\beta}\tilde{r} cos\theta cos\beta \end{aligned} \tag{13}$$

$$\tilde{z} = \tilde{r} sin\theta + \tilde{\theta} r cos\theta + \tilde{r}\tilde{\theta} cos\theta - \frac{1}{2}\tilde{\theta}^2 r sin\theta \tag{14}$$

Where the mean of the errors (12), (13) and (14) does not equal zero, the second-order Taylor series expansion approximations for the transformation are biased, which partially accounts for the bias of polar-to-Cartesian transformation measurement.

From (12), (13) and (14) the average true deviation μ_a and average true covariance R_a of converted measurement are described as

$$\mu_a = \begin{bmatrix} E(\tilde{x} \mid r,\beta,\theta) \\ E(\tilde{y} \mid r,\beta,\theta) \\ E(\tilde{z} \mid r,\beta,\theta) \end{bmatrix} = [\mu_a^x, \mu_a^y, \mu_a^z]^T \tag{15}$$

$$R_a = \begin{bmatrix} R_a^{xx} & R_a^{xy} & R_a^{xz} \\ R_a^{yx} & R_a^{yy} & R_a^{yz} \\ R_a^{zx} & R_a^{zy} & R_a^{zz} \end{bmatrix} \tag{16}$$

where

$$\mu_a^x = -\frac{1}{2}(\sigma_\theta^2 + \sigma_\beta^2)(1 - \frac{\sigma_\theta^2}{2} - \frac{\sigma_\beta^2}{2}) r_m \cos\theta_m \cos\beta_m \quad ,$$

$$\mu_a^y = -\frac{1}{2}(\sigma_\theta^2 + \sigma_\beta^2)(1 - \frac{\sigma_\theta^2}{2} - \frac{\sigma_\beta^2}{2}) r_m \cos\theta_m \sin\beta_m \quad ,$$

$$\mu_a^z = -\frac{1}{2}\sigma_\theta^2(1 - \frac{\sigma_\theta^2}{2}) r_m \sin\theta_m \quad ,$$

$$\begin{aligned} R_a^{xx} = & \sigma_\theta^2\sigma_\beta^2 r_m^2 \frac{\tilde{\alpha}_1}{4} + (\sigma_\theta^2\sigma_r^2 + \sigma_\theta^2 r_m^2)\frac{\tilde{\alpha}_2}{4} + (\sigma_\beta^2\sigma_r^2 + \sigma_\beta^2 r_m^2)\frac{\tilde{\alpha}_3}{4} \\ & + (\frac{\sigma_\theta^4}{2} r_m^2 + \frac{\sigma_\beta^4}{2} r_m^2 + \sigma_r^2)\frac{\tilde{\alpha}_4}{4} + \sigma_\theta^2\sigma_\beta^2\sigma_r^2\tilde{\delta}_1 + \sigma_\theta^2\sigma_r^2\tilde{\delta}_2 + \sigma_\beta^2\sigma_r^2\tilde{\delta}_3 \quad , \\ & + (\frac{\sigma_\theta^4}{2} + \frac{\sigma_\beta^4}{2})\sigma_r^2\tilde{\delta}_4 \end{aligned}$$

$$\begin{aligned} R_a^{yy} = & \sigma_\theta^2\sigma_\beta^2 r_m^2 \frac{\tilde{\alpha}_2}{4} + (\sigma_\theta^2\sigma_r^2 + \sigma_\theta^2 r_m^2)\frac{\tilde{\alpha}_1}{4} + (\sigma_\beta^2\sigma_r^2 + \sigma_\beta^2 r_m^2)\frac{\tilde{\alpha}_4}{4} \\ & + (\frac{\sigma_\theta^4}{2} r_m^2 + \frac{\sigma_\beta^4}{2} r_m^2 + \sigma_r^2)\frac{\tilde{\alpha}_3}{4} + \sigma_\theta^2\sigma_\beta^2\sigma_r^2\tilde{\delta}_2 + \sigma_\theta^2\sigma_r^2\tilde{\delta}_1 + \sigma_\beta^2\sigma_r^2\tilde{\delta}_4 \quad , \\ & + (\frac{\sigma_\theta^4}{2} + \frac{\sigma_\beta^4}{2})\sigma_r^2\tilde{\delta}_3 \end{aligned}$$

$$\begin{aligned} R_a^{zz} = & -\frac{1}{2}(4\sigma_\theta^2\sigma_r^2 - \sigma_r^2 - \frac{5}{2}\sigma_\theta^4\sigma_r^2)\sin\theta_m \sin\theta_m \\ & - \frac{1}{2}((\sigma_\theta^6 + \sigma_\theta^2) - \frac{5}{2}\sigma_\theta^4) r_m^2 \sin\theta_m \sin\theta_m \\ & + \frac{1}{2}(4\sigma_\theta^2\sigma_r^2 - \sigma_r^2 - \frac{5}{2}\sigma_\theta^4\sigma_r^2)\cos\theta_m \cos\theta_m \quad , \\ & + \frac{1}{2}((\sigma_\theta^6 + \sigma_\theta^2) - \frac{5}{2}\sigma_\theta^4) r_m^2 \cos\theta_m \cos\theta_m \\ & + \frac{1}{2}(\frac{1}{2}\sigma_\theta^4 + \sigma_\theta^2) r_m^2 + \frac{1}{2}(\sigma_r^2 + 2\sigma_\theta^2\sigma_r^2 + \frac{1}{2}\sigma_\theta^4\sigma_r^2) \end{aligned}$$

$$\begin{aligned} R_a^{yz} = & ((\frac{1}{2}\sigma_\theta^4 - \sigma_\theta^2) r_m^2 + \sigma_r^2 - \sigma_\theta^2\sigma_r^2)\left(1 - 2\sigma_\theta^2 - \sigma_\beta^2\right)\sin\theta_m \cos\theta_m \sin\beta_m \quad , \\ & + (\frac{1}{2}\sigma_\theta^4 - \sigma_\theta^2)\sigma_r^2 \sin\theta_m \cos\theta_m \sin\beta_m \end{aligned}$$

$$\begin{aligned} R_a^{xz} = & ((\frac{1}{2}\sigma_\theta^4 - \sigma_\theta^2) r_m^2 + \sigma_r^2 - \sigma_\theta^2\sigma_r^2)\left(1 - 2\sigma_\theta^2 - \sigma_\beta^2\right)\sin\theta_m \cos\theta_m \cos\beta_m \quad , \\ & + (\frac{1}{2}\sigma_\theta^4 - \sigma_\theta^2)\sigma_r^2 \sin\theta_m \cos\theta_m \cos\beta_m \end{aligned}$$

$$\begin{aligned} R_a^{xy} = & ((\frac{\sigma_\theta^4}{2} + \frac{\sigma_\beta^4}{2} - \sigma_\beta^2) r_m^2 + \sigma_r^2 - \sigma_\beta^2\sigma_r^2)\frac{\tilde{\gamma}_1}{4} + (\frac{\sigma_\theta^4}{2} + \frac{\sigma_\beta^4}{2} - \sigma_\beta^2)\sigma_r^2 \frac{\tilde{\gamma}_2}{4} \\ & + (\sigma_\theta^2 - \sigma_\theta^2\sigma_\beta^2)\sigma_r^2 \frac{\tilde{\gamma}_4}{4} + ((\sigma_\theta^2 - \sigma_\theta^2\sigma_\beta^2) r_m^2 + \sigma_\theta^2\sigma_r^2)\frac{\tilde{\gamma}_3}{4} \quad , \\ & - \frac{1}{4}((\frac{\sigma_\theta^4}{2} + \frac{\sigma_\beta^4}{2} - \sigma_\beta^2)\sigma_r^2 - (\sigma_\theta^2 - \sigma_\theta^2\sigma_\beta^2)\sigma_r^2) \\ & - \frac{1}{4}((\frac{\sigma_\theta^4}{2} + \frac{\sigma_\beta^4}{2} - \sigma_\beta^2 + \sigma_\theta^2 - \sigma_\theta^2\sigma_\beta^2) r_m^2 + \sigma_\theta^2\sigma_r^2 + \sigma_r^2 - \sigma_\beta^2\sigma_r^2) \end{aligned}$$

$$\begin{aligned} \tilde{\alpha}_1 = & 1 - (1 - 2\sigma_\beta^2) os 2\beta_m - (1 - 2\sigma_\theta^2)\cos 2\theta_m \\ & + (1 - 2\sigma_\theta^2 - 2\sigma_\beta^2)\cos 2\theta_m \cos 2\beta_m \end{aligned}$$

$$\tilde{\alpha}_2 = 1 + (1-2\sigma_\beta^2)\text{os}\,2\beta_m - (1-2\sigma_\theta^2)\cos 2\theta_m + (1-2\sigma_\theta^2 - 2\sigma_\beta^2)\cos 2\theta_m \cos 2\beta_m ,$$

$$\tilde{\alpha}_3 = 1 + (1-2\sigma_\beta^2)\text{os}\,2\beta_m + (1-2\sigma_\theta^2)\cos 2\theta_m - (1-2\sigma_\theta^2 - 2\sigma_\beta^2)\cos 2\theta_m \cos 2\beta_m ,$$

$$\tilde{\alpha}_4 = 1 - (1-2\sigma_\beta^2)\text{os}\,2\beta_m + (1-2\sigma_\theta^2)\cos 2\theta_m + (1-2\sigma_\theta^2 - 2\sigma_\beta^2)\cos 2\theta_m \cos 2\beta_m ,$$

$$\begin{aligned}\tilde{\delta}_1 &= 1 - \cos 2\beta_m - \cos 2\theta_m + \cos 2\theta_m \cos 2\beta_m \\ \tilde{\delta}_2 &= 1 + \cos 2\beta_m - \cos 2\theta_m - \cos 2\theta_m \cos 2\beta_m \\ \tilde{\delta}_3 &= 1 - \cos 2\beta_m + \cos 2\theta_m - \cos 2\theta_m \cos 2\beta_m \\ \tilde{\delta}_4 &= 1 + \cos 2\beta_m + \cos 2\theta_m + \cos 2\theta_m \cos 2\beta_m\end{aligned} ,$$

$$\begin{aligned}\tilde{\gamma}_1 &= 1 + (1-2\sigma_\beta^2)\sin 2\beta_m + (1-2\sigma_\theta^2 - 2\sigma_\beta^2)\cos 2\theta_m \sin 2\beta_m \\ \tilde{\gamma}_2 &= 1 + \sin 2\beta_m + \cos 2\theta_m \sin 2\beta_m \\ \tilde{\gamma}_3 &= 1 + (1-2\sigma_\beta^2)\sin 2\beta_m - (1-2\sigma_\theta^2 - 2\sigma_\beta^2)\cos 2\theta_m \sin 2\beta_m \\ \tilde{\gamma}_4 &= 1 + \sin 2\beta_m - \cos 2\theta_m \sin 2\beta_m\end{aligned} .$$

When measurement in the polar coordinate is converted to be in the Cartesian coordinate, the measurement is modified as

$$Z_c = Z - \mu_a = \begin{bmatrix} r_m \cos\theta_m \cos\beta_m \\ r_m \cos\theta_m \sin\beta_m \\ r_m \sin\theta_m \end{bmatrix} - \mu_a \tag{17}$$

4.2 Target tracking model

This paper selects Singer acceleration model as target dynamic model [12]. State equation of the system is

$$X_{k+1} = \Phi X_k + \Gamma_k W_k \tag{18}$$

Measurement equation is

$$Z_k = H_k X_k + V_k \tag{19}$$

where, $X_k = [x_k, y_k, z_k, \dot{x}_k, \dot{y}_k, \dot{z}_k, \ddot{x}_k, \ddot{y}_k, \ddot{z}_k]^T$ serves as state vector of the system, including target's position, velocity and acceleration in X, Y and Z direction, respectively. Φ is system state transition matrix, Γ_k is noise gain matrix, W_k is system process noise matrix, Z_k is the system measurement vector, H_k is measurement matrix, V_k is measurement noise vector.

$$\Phi = \begin{bmatrix} 1 & 0 & 0 & T & 0 & 0 & \phi_{17} & 0 & 0 \\ 0 & 1 & 0 & 0 & T & 0 & 0 & \phi_{28} & 0 \\ 0 & 0 & 1 & 0 & 0 & T & 0 & 0 & \phi_{39} \\ 0 & 0 & 0 & 1 & 0 & 0 & \phi_{47} & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & \phi_{58} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & \phi_{69} \\ 0 & 0 & 0 & 0 & 0 & 0 & e^{-\alpha_x T} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & e^{-\alpha_y T} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & e^{-\alpha_z T} \end{bmatrix} \tag{20}$$

Where

$\phi_{17} = (\alpha_x T - 1 + e^{-\alpha_x T}) / \alpha_x{}^2$, $\phi_{28} = (\alpha_y T - 1 + e^{-\alpha_y T}) / \alpha_y{}^2$,

$\phi_{39} = (\alpha_z T - 1 + e^{-\alpha_z T}) / \alpha_z{}^2$, $\phi_{47} = (1 - e^{-\alpha_x T}) / \alpha_x$, $\phi_{58} = (1 - e^{-\alpha_y T}) / \alpha_y$,

$\phi_{69} = (1 - e^{-\alpha_z T}) / \alpha_z$.

$$\Gamma_k = \begin{bmatrix} \Gamma_1 & \Gamma_2 & \Gamma_3 \end{bmatrix}^T \tag{21}$$

where

$$\Gamma_1 = diag\left(\begin{bmatrix} \gamma_x \left[1 - \alpha_x T - e^{-\alpha_x T} + (T^2 \alpha_x{}^2/2)\right] / \alpha_x{}^3 \\ \gamma_y \left[1 - \alpha_y T - e^{-\alpha_y T} + (T^2 \alpha_y{}^2/2)\right] / \alpha_y{}^3 \\ \gamma_z \left[1 - \alpha_z T - e^{-\alpha_z T} + (T^2 \alpha_z{}^2/2)\right] / \alpha_z{}^3 \end{bmatrix}\right),$$

$$\Gamma_2 = diag\left(\begin{bmatrix} \gamma_x (\alpha_x T + e^{-\alpha_x T} - 1) / \alpha_x{}^2 \\ \gamma_y (\alpha T + e^{-\alpha_y T} - 1) / \alpha_y{}^2 \\ \gamma_z (\alpha T + e^{-\alpha_z T} - 1) / \alpha_z{}^2 \end{bmatrix}\right),$$

$$\Gamma_3 = diag\left(\begin{bmatrix} \gamma_x (1 - e^{-\alpha_x T}) / \alpha_x \\ \gamma_y (1 - e^{-\alpha_y T}) / \alpha_y \\ \gamma_z (1 - e^{-\alpha_z T}) / \alpha_z \end{bmatrix}\right),$$

$\gamma_i = sqrt(a^2{}_{\max}(i)(1 + 4P_{\max}(i) - P_0)/3), i = x, y, z$.

where, γ_x, γ_y, γ_z, $\alpha_x = \alpha_y = \alpha_z = \alpha$ describes the first-order forming filter parameter of the attacking target's acceleration in the Cartesian coordinate [5]. T is system measurement period.

$$H_k = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{22}$$

4.3 SDCMKF algorithm

The block diagram of SDCMKF algorithm is shown in Fig.6.

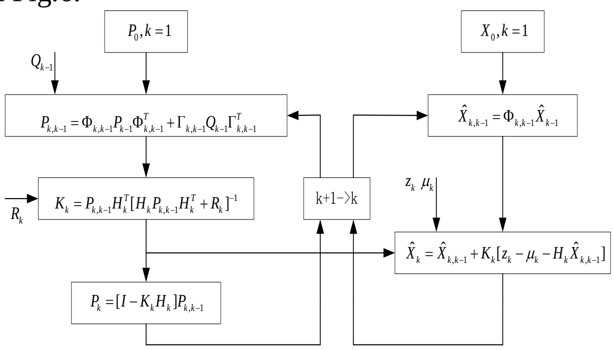

Fig.6 Block diagram of DCMKF algorithm

The calculation steps of SDCMKF is given as

1) Calculating the initial value X_0 and initial covariance matrix P_0, and assuming

$$X_f = X_0,\ P_f = P_0 .$$

2) Predicting state vector

$$X_p = \Phi X_f \tag{23}$$

3) Calculating covariance matrix of the predicted states

$$P_p = \Phi P_f \Phi^T + \Gamma Q \Gamma^T \tag{24}$$

4) Calculating the mean deviation μ_a and covariance R_a of the converted measurement according to equation (15) and (16).

5) Calculating gain matrix

$$K_k = P_p H^T (H P_p H^T + R_k)^{-1} \tag{25}$$

6) Updating state vector

$$X_f = X_p + K_k(Z_k - \mu_k - HX_p) \quad (26)$$

7) Updating covariance matrix

$$P_f = (I - K_k H)P_p \quad (27)$$

8) Repeating step 2) to 6) for recursive computation

5 Realization of hardware of SDCMKF algorithm

5.1 Structural hierarchy design

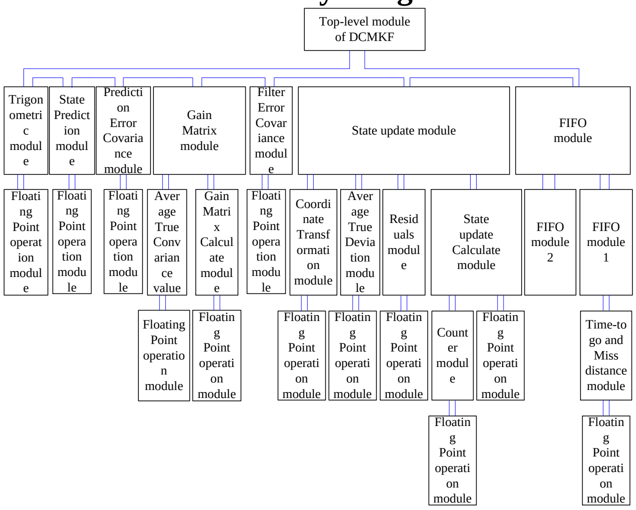

Fig.7 Structural hierarchy design of SDCMKF and air defense missile detonation delay based on FPGA

This paper adopts the structural hierarchy design idea to design SDCMKF algorithm, the hierarchy diagram of SDCMKF based on FPGA is shown in Fig.7. The bottom module selects very high speed integrated circuits hardware description language (VHDL) as input, while the top module selects schematic diagram as input. This design scheme improves code readability, facilitates simulation in the design, and easy to module partition. To guarantee the calculation precision, the 32-bit single precision floating point number based on IEEE754 standard is utilized in the module design. We choose Altera' Cyclone Ⅲ series EP3SL200H780 chip to achieve SDCMKF algorithm on Quartus Ⅱ platform. The arithmetic operation modules of Quartus Ⅱin floating point number consist of addition, subtraction, multiplication and division. The basic floating point number operation modules are instantiated and the corresponding parameters are set accordingly in the VHDL code design, which would improve the design performance and shorten the design time, simply the realization the data path of floating point.

5.2 Design of computing modules of SDCMKF

To realize SDCMKF algorithm by FPGA, this algorithm needs to be pre-processed from matrix form to vector form [6]. This design scheme can realize codes easily, simplify scalar calculation, avoid the complicated multiplication, addition calculation of sparse matrix with a large number of zero cells, and save FPGA resource efficiently [7]. Scientific Workspace software can show clearly the variable relationship between input matrix array and output matrix array. When FPGA processes floating point calculation, it occupies more computing resource. Therefore, time division multiplexing technology is applied for the fundamental calculation module in this paper [8, 9].

Take the state prediction module as an example. State predicted value can be modified as

$$Xp = [Xp_0, Xp_1, Xp_2, Xp_3, Xp_4, Xp_5, Xp_6, Xp_7, Xp_8]^T$$

where

$$Xp_0 = Xf_0 + T * Xf_3 + \phi_{17} * Xf_6$$
$$Xp_1 = Xf_1 + T * Xf_4 + \phi_{28} * Xf_7$$
$$Xp_2 = Xf_2 + T * Xf_5 + \phi_{39} * Xf_8$$
$$Xp_3 = Xf_3 + \phi_{47} * Xf$$
$$Xp_4 = Xf_4 + \phi_{58} * Xf_7$$
$$Xp_5 = Xf_5 + \phi_{69} * Xf_8$$
$$Xp_6 = e^{-\alpha T} * Xf_6$$
$$Xp_7 = e^{-\alpha T} * Xf_7$$
$$Xp_8 = e^{-\alpha T} * Xf$$

Fig.8 is the structure diagram of state prediction module. State prediction module occupies 2 floating point addition operation units and 2 floating point multiplication units. In this paper, the period parameters of floating point addition and multiplication units are set respectively as 7 and 5 clock cycles in library parameter module (LPM). Because the data number of input port cannot always make up 2n, the data which don't participate in computing should be set for 5 clock delays at the processing data in the first stage floating point multiplication processing to guarantee data synchronization. Similarly, the previous stage results which don't participate in computing should be set for 7 clock delays in the second stage floating point addition. When input port receives 9 state values one clock cycle by another, the data distribution module sends its corresponding multiplicand and multiplier to the right register for processing at each clock. After 19(5+7+7) clock cycles, state prediction module output processed data at per clock cycle.

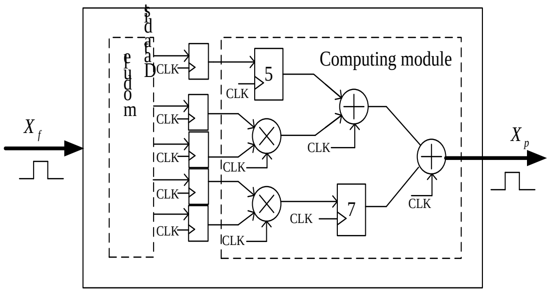

Fig.8 Structure diagram of state prediction module

Fig.9 is the Top level schematic diagram of prediction module in step one.

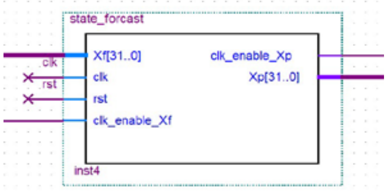

Fig.9 Schematic diagram of predictable module in step one

where Xf is the input data port of state prediction module, namely the state estimate before a moment. clk is the clock signal, rst is the reset signal, clk_enable_Xf is the enable signal of output data of receiving former module, clk_enable_Xp is the enable signal of receiving data next operation module, Xp is output data port of state prediction module, the design idea of input and output port of other operation module is similar, here not repeated introduction.

5.2.1 Counter module

When the encounter end, or the distance of the missile and target is small, the parameter errors caused by motor can be ignored, so it can be regarded as uniform motion. So this stage design is a key point and difficulty, here we can design a counter, when the counter counts to 78, that is, measuring the distance to the target is not higher than 350 meters, it can be treated as uniform motion.

Fig.10 is the Top level schematic diagram of counter module.

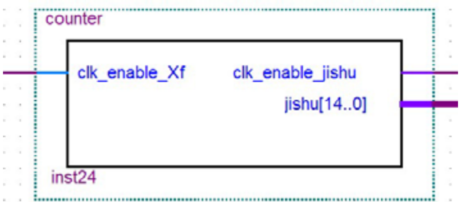

Fig.10 Schematic diagram of counter module.

where clk_enable_Xf is the enable signal of output data of receiving former module, jishu is output data port of counter module, clk_enable_jishu is the enable signal of counter module, and initial value of jishu is zero, and it has began to count until the clk_enable_Xf from low level became into a high level. The clk_enable_jishu has always been low level until it became high level when counter counted to 78.

5.2.2 Design of air defense missile detonation delay

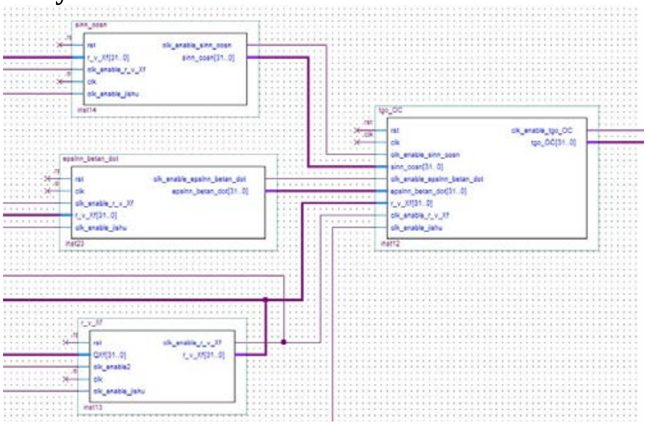

Fig.11 Schematic diagram of air defense missile detonation delay.

Fig.11 is the top level schematic diagram of air defense missile detonation delay module. Air defense missile detonation delay module consists of sinn_cosn, epslnn_betan_dot, r_v_Xf and tgo_OC module. sinn_cosn is the angle (Angle of pitch and azimuth) of sin and con after filtering module, epslnn_betan_dot is the angle (Angle of pitch and azimuth) derivative after filtering module, r_v_Xf is the distance, speed and the direction of the position vector after filtering module, tgo_OC is the time to go and miss distance module.

Where clk_enable2 is the reading enable signal of FIFO memory module(lpm_fifo1 module is used to temporarily store state update value), QXf is output data port of lpm_fifo1 module, here it is used as input data port of r_v_Xf module, clk_enable_r_v_Xf is the output enable signal of r_v_Xf module, r_v_Xf is output data port of r_v_Xf module.

6 Simulation and results

Fig.12 shows the hardware circuit board , this subsystem is composed of DSP, FPGA, SDRAM, FLASH, SD and other components. The power supply of circuit board of DSP subsystem is DC 5V. The power supply of the circuit board of FPGA subsystem is 24V DC. Firstly, the power convert chip converts the DC 24V to DC 5V, then supply to the DSP subsystem through the connection. In the FPGA subsystem, DC 5V is converted to 3.3V, 2.5V and 1.2V respectively.

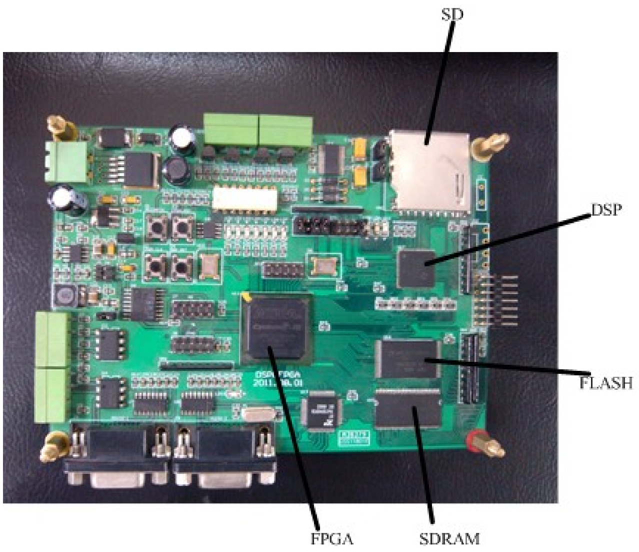

Fig.12 Hardware circuit board

In this section, a simulation scenario is presented to track a highly maneuvering target. The target has different acceleration at different time segment. The parameters of target are given as follows. The initial conditions of the target is $(954m, 697m, 302m)$ for position and $(-634m/s, -465m/s, -200m/s)$ for velocity. The sampling rate is $t = 10ms$. The segments are defined as follows. 1st segment, $t = (0-250)ms$, constant velocity flight with acceleration $(5m/s^2, 2m/s^2, 0)$. 2nd segment, $t = (251-500)ms$, accelerated flight with acceleration $(-5m/s^2, -2m/s^2, 0)$. 3rd segment, $t = (501-750)ms$, accelerated flight with acceleration 0. In Singer module, the variance of the measured distance r, elevation θ and azimuth β are $\sigma_r = 5m^2$, $\sigma_\theta^2 = 5\times10^{-5}rad^2$, $\sigma_\beta^2 = 5\times10^{-5}rad^2$, The variance of the process noise for Singer module $Q = diag(2,2,2)$, the probability of biggest maneuverable acceleration $P_{\max} = 0.5$, the probability of non-maneuverable $P_0 = 0.5$. The target's first-order forming filter parameters of acceleration in Cartesian coordinate are $\gamma_x = 2.041$, $\gamma_y = 1.291$, $\gamma_z = 0.2887$, $\alpha = 0.1$. In the process of simulation, firstly, the seeker's measurement information are filtered using the SDCMKF, then the time-to-go and miss distance are estimated. The SDCMKF algorithm is realized respectively in FPGA and Matlab platform using the same measurements. Fig.13 is after and before filtering of X. Fig.14 is after and before filtering of Y. Fig.15 is after and before filtering of Z. Fig.16 is after and before filtering of Time-to-go. Fig.17 is after and before filtering of miss distance . It is easily seen from the 5 figures above that the SDCMKF is capable of denoising and smoothing for target position. Fig.13, Fig.14, Fig.15, Fig.16 and Fig.17 show the results of FPGA are consistent with the simulated results by Matlab. The high precision proves the correctness of this design scheme.

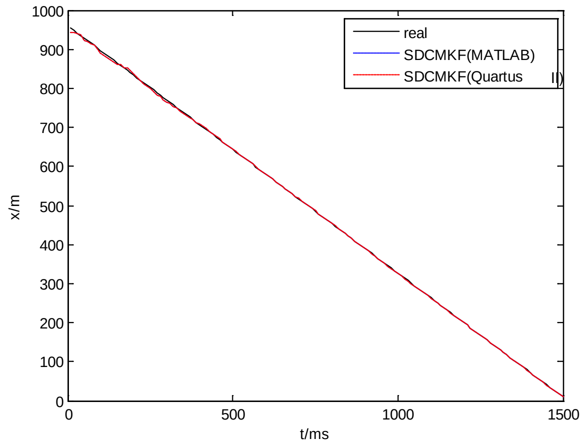

Fig.13 Comparision in axis X

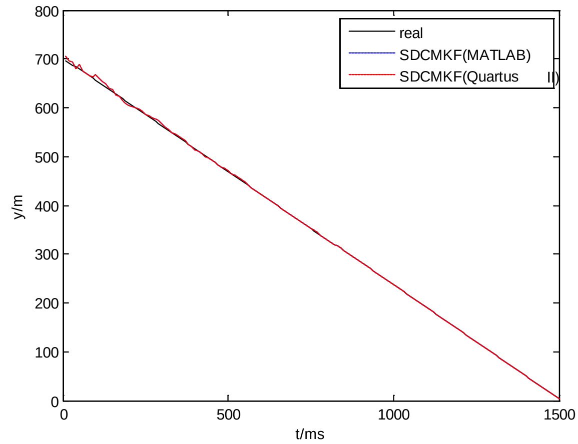

Fig.14 Comparision in axis Y

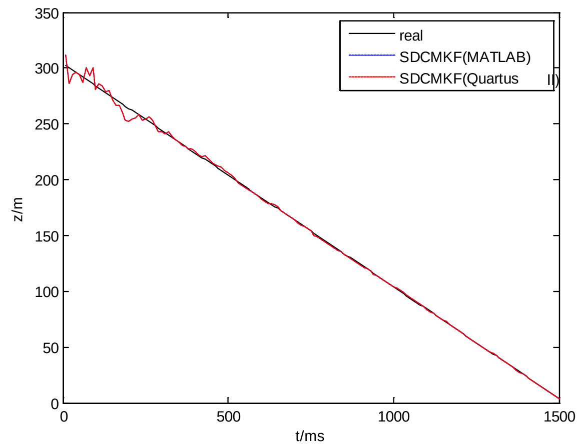

Fig.15 Comparision in axis Z

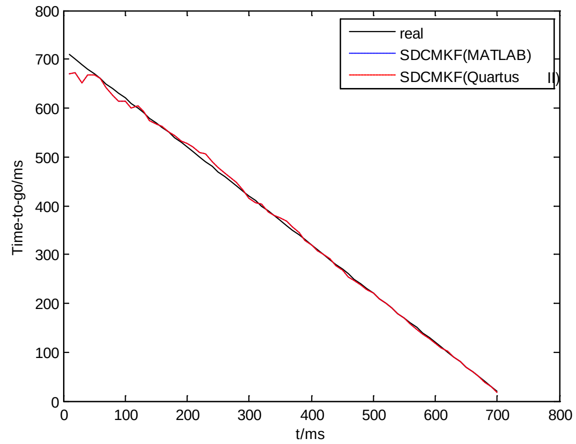

Fig.16 Comparison in axis time-to-go

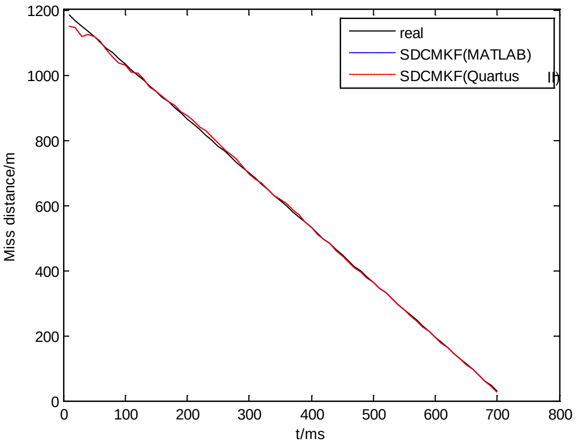

Fig.17 Comparision in axis miss distance

In [4], the second-order debiased converted measurement Kalman filter (SDCMKF) algorithm is proposed to reduce the effect of the linearization approximation error of the conventional EKF and CMKF algorithms in the 3D tracking system. A numerical simulation example is given, which indicates that the SCMKF effectively reduces the nonlinear effect by the polar measurement and improves the accuracy and the convergence of the tracking system.

From the timing simulation in Quartus II, we can see that experimental results prove the designed SDCMKF algorithm based on FPGA spends 455 clock periods to complete one filter process. If the clock periods is 40ns, a filtering cycle is 18.2us. This time fully satisfies the real time requirement in target tracking system. The radar tracking performance gained in our design scheme includes two to three orders of magnitude higher speed than single DSP design scheme.

7 Conclusion

In this paper, mathematical models are established for optimal detonation delay control under arbitrary encounter conditions in missile coordinate and a new SDCM tracking algorithm using radar measurements is proposed for optimal detonation delay control. In the radar target tracking system, the tracking precision and real time are highly required. SDCMKF algorithm includes a great deal of matrix arithmetic, such as matrix addition, matrix subtraction, matrix multiplication and inverse. The computational time for calculating SDCMKF algorithm in software is too long to meet the real time of target tracking. In this paper, the FPGA is used as a floating point co-processor of fixed point DSP. This software and hardware reasonable design scheme can solve the concurrency and speed problems and guarantee the tracking precision. Therefore, it is an effective approach to complete target tracking algorithm. The design based on FPGA has large degree flexibility for programming, updates codes at any time, and largely reduces the research cost. This research results have been successfully applied to a certain type of short-range defence radar.

Acknowledgement
This work is partially supported by the National Nature Science Foundation of China (NO. 61104196), “Zijin star” Research Funding(NO. AB41381).

References:

[1] Chopper, K., H. Jaeger, L. Stephens, et al, Guidance integrated predictive fuzing design, Proc. *American Control Conference*, Chicago, IL, June 1992.

[2] Chopper, K., H. Harold, L. Stephens, et al, Guidance integrated fuzing analysis and simulation, Proc. *Conference on Control Applications*, Dayton, OH, Sept. 1992.

[3] Yan Han-xin, Jiang Chun-lan, Research on terminal efficiency of air-air missile against hypersonic weapons with GIF (Guidance Integrated Fuzing), *2010 2nd International Asia Conference on Informatics in Control, Automation and Robotics*, 262-265.

[4] Chen H., Tan J.B., Target tracking based on second-order converted measurement Kalman filter, *Opto-Electronic Engineering*, Vol.35, No. 4, 2008,pp.6-11.

[5] Bar-Shalom Y, Li X R, Kirubarajan T, Estimation with Applications to Tracking and Navigation: Therory, *Algorithm and Software*, New York: Wiley, 2001.

[6] Dr S M Shalinie, Associate Menber. Design and Analysis of Customized Embedded Kalman

Filter. IE(I) *Journal-CP*, Vol.88, NO.5 2007, pp.39-42.
[7] Neri F. Cooperative Evolutive Concept Learning: An Empirical Study. WSEAS *Transaction on Information Science and Applications*, Vol.2, No.5, 2005, pp.559-563.
[8] Abbas Bigdeli, Morteza Biglari-Abhari, Zoran Salcic, and Yat Tin Lai. A New Pipelined Systolic Array-Based Architecture for Matrix Inversion in FPGAs with Kalman Filter Case Study. *EURASIP Journal on Applied Signal Processing*, Vol.2006, Article ID 89186, 2006, pp.1-12.
[9] Ventzas, D. A 4-bit Quantized Skip Algorithm Software Correlator for Microcomputer Systems. *IASTED Symposium on Measuremen*, *Signal Processing and Control*, MECO'86, Taormina, Italy, IASTED.
[10] Zhuang Z.H, Tu J.P., Wang H.B., Prediction of time-to-go during missile-target encounter, *Journal of Astronautics*, Vol.23, No.5, 2002, pp.32-37.
[11] Pan-Long Wu,Bao-Bao Wang,Cun-Hui Ji, Design and Realization of Short Range Defence Radar Target Tracking System Based on DSPFPGA, *WSEAS Transactions on System*. Vol.10, No.11, 2011, pp.379-380.
[12] ROBERT A. SINGER. Estimating optimal tracking filter performance for manned manoeuvring targets. IEEE *Transactions on Aerospace and Electronic Systems*, Vol.6, No.4, 1970, pp.473-483.
[13] Wu P.L., Wang B.B., Ji C.H., Design and realization of short range defence radar target tracking system based on DSP/FPGA, *WSEAS Transactions on System*, Vol.10, No.11, 2011, pp.376-386.
[14] Farina Alfonso, Ristic Branko, Benvenuti Dario. Tracking a Ballisitic Traget: Comparision of Several Nonlinear Filters. IEEE *Transactions on Aerospace and Electronic Systems*, Vol.38, No.3, 2002, pp.854-867.
[15] YU Yihua, CHENG Qian-sheng. Particle filters for maneuvering target tracking problem. *Signal Processing*, Vol.86, 2006, pp.195-203.
[16] Hsieh Chien-Shu. General Two-Stage Extended Kalman Filters. IEEE *Transactions on Automatic Control*, Vol.48, No.2, 2003, 289-293.
[17] CHEN Gang, GUO Li. The FPGA Implementation of Kalman Filter. *Proceedings of the 5th WSEAS Int. Conf. on Signal Processing*, 2005, pp.61-65.
[18] C.R. Lee, Z. Salcic. High-performance FPGA-based Implementation of Kalman Filter. *Mircroprocessors and Microsystems*, Vol.21, No. 4, 1997, pp.257-265.
[19] ZHOU Hong-bo, GENG Bo-ying. Converted Measurements Kalman Filter Algorithm for Target Tracking. *Journal of System Simulation*, Vol.20, No.3, 2008, pp.682-688.
[20] HE Ming-ke, WANG Zheng-ming, ZHU Ju-bo. Debiased Converted Measurement KF for Radar Target Tracking. *Journal of National University of defence technology*, Vol.24, No.5, 2002, pp.57-60.
[21] ZHOU Ning-ning, CHEN Yan-li, LI Ai-qun. Design and implementation of floating point calculator based on FPGA technology. *Computer Engineer and Design*, Vol.26, No.6, 2005, pp.1578-1581.
[22] ZHONG Sheng, HOU Chao-huan, YANG Chang-an. Optimized design of matrix multiplier based on FPGA. *Electronic Measurement Technology*, Vol.31, No.2, 2008, pp.95-102.
[23] CHEN Gang, GUO Li. The FPGA Implementation of Kalman Filter. *Proceedings of the 5th WSEAS Int. Conf. on Signal Processing*, 2005, pp.61-65.