

PAPER • OPEN ACCESS

## Research on Sampling Data Processing of High Speed Pulse

To cite this article: Zhenjie Zeng *et al* 2018 *IOP Conf. Ser.: Mater. Sci. Eng.* **452** 042009

View the [article online](#) for updates and enhancements.

You may also like

- [Research on CNC Lathe Programming and Improving Machining Accuracy](#)  
Juan Zhang
- [Interference Analysis and Simulation of Active Jamming for Two Phase Code PD Radar Fuze](#)  
Shangjun Cai, Dongguang Zuo, Xiaofei Zhu et al.
- [Pseudo-Satellite Data for Monitoring Cloud Platform](#)  
Guofu Wang, Zhen Li, Jincai Ye et al.



**ECS**  
The  
Electrochemical  
Society  
Advancing solid state &  
electrochemical science & technology

**DISCOVER**  
how sustainability  
intersects with  
electrochemistry & solid  
state science research

# Research on Sampling Data Processing of High Speed Pulse

Zhenjie Zeng<sup>1</sup>, Xiaofei Zhu<sup>1,\*</sup>, Hao Chen<sup>2</sup>, Kai Wu<sup>3</sup>, Shiju Qi<sup>1</sup> and JiangRen Lu<sup>1</sup>

<sup>1</sup>Rocket Force University of Engineering, Xi'an, China

<sup>2</sup>Troops No. 92330, Qingdao, China

<sup>3</sup>Troops No. 96604, Beijing, China

\*Corresponding author e-mail: 1224048915@qq.com

**Abstract.** In high-speed pulse acquisition systems, the processing of high-speed pulsed signal sampling data is critical. Firstly, the high-speed pulse signal model is analyzed and established. Then the curve equations are fitted by Newton interpolation method to solve the important parameters of the discrete signal. Finally, the algorithm is applied to FPGA for verification, and the signal processing is realized, indicating that the Newton interpolation method is at high speed. Sampling data processing is feasible.

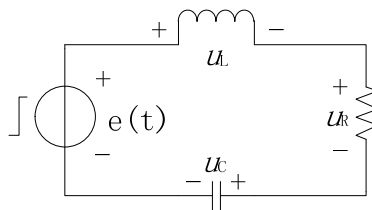
## 1. Introduction

In the high-speed pulse acquisition system, the processing of high-speed pulse signal sampling data is very critical. By processing high-speed sampling data, important parameters such as pulse amplitude and front-end time can be obtained. Using the FPGA chip in the pulse collector, using the idea of embedded computing to process the sampled data, the key lies in the research of data processing algorithms. Therefore, this paper carries out research on high-speed pulse sampling data processing algorithms and implements them through FPGA.

## 2. High-speed signal model analysis

In order to solve the discrete sampled data and obtain approximate curve equations and then solve the parameter information, firstly, a signal model needs to be established, and then interpolation and curve fitting methods are used to process the data.

When a high-speed signal passes through a transmission line, the transmission line is no longer a fully resistive circuit at this time due to the effects of distributed inductance and distributed capacitance, but is equivalent to an RLC circuit. Studying the model of high-speed signal transmission is equivalent to studying the characteristics of the step signal passing through the RLC circuit (Figure 1).



**Figure 1.** Step signal through RLC circuit schematic

When the excitation is a step signal, which is  $e(t) = \varepsilon(t)$ , The response current is  $i(t)$ . The initial state of the circuit is zero, and the response at this time includes only the zero state response component. The transfer function  $H(s)$  of the system is:

$$H(s) = \frac{1}{Ls + R + \frac{1}{Cs}} = \frac{1}{L} \left( \frac{s}{s^2 + \frac{R}{L}s + \frac{1}{LC}} \right) \quad (1)$$

The extremes are:

$$s_{1,2} = -\frac{R}{2L} \pm \sqrt{\frac{R^2}{4L^2} - \frac{1}{LC}} \quad (2)$$

Among them, the attenuation coefficient is  $\alpha = -\frac{R}{2L}$ . Resonance frequency is  $\omega_0 = \frac{1}{\sqrt{LC}}$ . So the pole can be expressed as: So the pole can be expressed as:

$$s_{1,2} = -\alpha \pm \sqrt{\alpha^2 - \omega_0^2} \quad (3)$$

The response current is:

$$I(s) = H(s)E(s) = \frac{1}{L} \left[ \frac{s}{(s-s_1)(s-s_2)} \right] \frac{E}{s} = \frac{E}{L} \left[ \frac{1}{(s-s_1)(s-s_2)} \right] \quad (4)$$

The Laplace transform of the response current is the same as the pole of  $H(s)$ . At this time, it needs to be divided into three cases.

(1) as  $\alpha > \omega_0$ , which is  $R > 2\sqrt{\frac{L}{C}}$ :

The pole at this time is two different real roots, available:

$$i(t) = \frac{E}{L} \frac{e^{(-\alpha + \sqrt{\alpha^2 - \omega_0^2})t} - e^{(-\alpha - \sqrt{\alpha^2 - \omega_0^2})t}}{2\sqrt{\alpha^2 - \omega_0^2}} \varepsilon(t) = \frac{E}{L} \frac{\sinh \sqrt{\alpha^2 - \omega_0^2} t}{\sqrt{\alpha^2 - \omega_0^2}} e^{-\alpha t} \varepsilon(t) \quad (5)$$

(2) as  $\alpha = \omega_0$ , which is  $R = 2\sqrt{\frac{L}{C}}$ :

There is a second-order pole in  $-\alpha$  at this time, available:

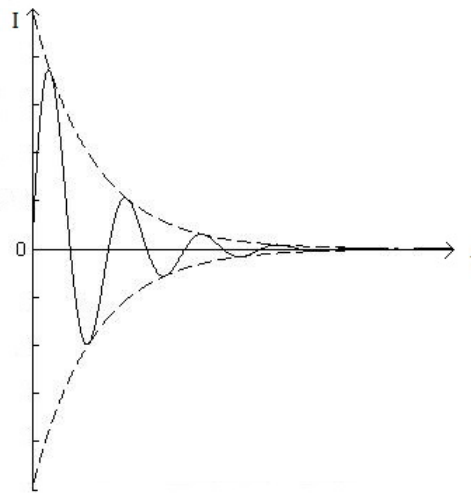
$$i(t) = \frac{E}{L} t e^{-\alpha t} \varepsilon(t) \quad (6)$$

(3) as  $\alpha < \omega_0$ , which is  $R < 2\sqrt{\frac{L}{C}}$ :

At this point there is a pair of conjugated poles available:

$$i(t) = \frac{E}{\omega_n L} e^{-\alpha t} \sin(\omega_n t) \varepsilon(t) \quad (7)$$

Among them,  $\omega_n = \sqrt{\omega_0^2 - \alpha^2}$ , For damped natural frequencies. The current response waveform is a sine wave with amplitude decaying according to an exponential coefficient. Figure 2. The decay index coefficient is  $\frac{E}{\omega_n L} e^{-\alpha t}$ .



**Figure 2.** Response current of step signal through RLC circuit under overdamped condition

The above conditions depend on the loss in the actual circuit. When the loss is too large, no oscillation occurs. When there is no loss, equal amplitude oscillations are generated. Cases (1), (2), and (3) are underdamping, critical damping, and overdamping, respectively. A high-speed pulse passes through the RLC circuit when it passes through the RLC circuit similar to a step signal under overdamped conditions ( $R < 2\sqrt{\frac{L}{C}}$ ), so case (3) is its ideal transmission model.

### 3. Signal fitting algorithm

After the high-speed pulse is acquired, a series of discrete data is obtained. When solving the curve parameters, a curve fitting method is needed to recover the continuous waveform according to the discrete data points. At the same time, because the signal rate is too high, when solving for a certain interval (such as calculating the local maximum), there are relatively few sampling points, and the existing data points cannot completely reflect the original signal conditions. In this case, interpolation data needs to be used from the existing data. Further find the interval function. In the design of the algorithm, the best square approximation method and Newton interpolation method with good effect are selected. Firstly, the principle of the two methods is briefly introduced.

#### 3.1. The principle of the algorithm

##### 1. Optimal Square Approximation (1)

The function  $y=f(x)$  is known to have a value  $y_i$  ( $i=1, 2, \dots, m$ ) at  $m$  points  $x_i$ , requiring a function

$$p(x) = c_0 \phi_0(x) + c_1 \phi_1(x) + \dots + c_n \phi_n(x) \quad (8)$$

Where  $c_0, c_1 \dots c_n$  are the undetermined constants,  $\varphi_0(x), \varphi_1(x) \dots c_n \varphi_n(x)$  are linearly independent functions, determine the value of the pending constant, so that

$$S = \sum_{i=1}^n \omega_i [p(x) - y_i]^2 \quad (9)$$

Is minimal. Where  $\omega_i$  is a known weight coefficient, and the function  $p(x)$  solved at this time is the least squares fitting function of function  $f(x)$ .

In the interval  $[a, b]$ , corresponding changes are needed to solve the approximate function. Let  $\omega(x)$  be a positive function except that individual points are 0,  $a=x_0 < x_1 < \dots < x_{m-1} < x_m=b$ , and change the above equation to

$$S = \sum_{i=1}^n \omega(i) \Delta x_i [p(x_i) - y_i]^2 \quad (10)$$

Among them,  $\Delta x_i = x_i - x_{i-1}$ , assumed  $\lambda = \max_{1 \leq i \leq m} \Delta x_i$ , as  $\lambda \rightarrow 0$ , The above formula is

$$S = \int_a^b \omega(x) [p(x) - f(x)]^2 dx \quad (11)$$

The  $p(x)$  when the value of  $S$  is the smallest is the optimal square approximation function.

The weight coefficient  $\omega_i$  and the weight function  $\omega(x)$  are usually 1, and the definition of the inner product can be used to derive that when  $S$  is minimum, the undetermined constants  $c_0, c_1 \dots c_n$ , satisfy:

$$\sum_{j=0}^n (\varphi_k, \varphi_j) c_j = (\varphi_k, y) \quad k = 0, 1, \dots, n \quad (12)$$

The normal equations called the optimal square approximation  $p(x)$  are expressed in matrix form:

$$\begin{bmatrix} (\varphi_0, \varphi_0) & (\varphi_0, \varphi_1) & \cdots & (\varphi_0, \varphi_n) \\ (\varphi_1, \varphi_0) & (\varphi_1, \varphi_1) & \cdots & (\varphi_1, \varphi_n) \\ \vdots & \vdots & \ddots & \vdots \\ (\varphi_n, \varphi_0) & (\varphi_n, \varphi_1) & \cdots & (\varphi_n, \varphi_n) \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \\ \vdots \\ c_n \end{bmatrix} = \begin{bmatrix} (\varphi_0, y) \\ (\varphi_1, y) \\ \vdots \\ (\varphi_n, y) \end{bmatrix} \quad (13)$$

From the normal equations, the undetermined constants  $c_0, c_1 \dots c_n$ , can be determined to determine the optimal square approximation function.

## 2. Newton interpolation (2) (3)

A known  $(x_0, y_0), (x_1, y_1) \dots (x_n, y_n)$ , thus

$$N_n(x) = c_0 + c_1(x - x_0) + c_2(x - x_0)(x - x_1) + \cdots + c_n(x - x_0)(x - x_1) \cdots (x - x_{n-1}) \quad (14)$$

Meet the requirements:

$$\begin{cases} y_0 = N_n(x_0) = c_0 \\ y_1 = N_n(x_1) = c_0 + c_1(x_1 - x_0) \\ y_2 = N_n(x_2) = c_0 + c_1(x_2 - x_0) + c_2(x_2 - x_0)(x_2 - x_1) \\ \vdots \\ y_n = N_n(x_n) = c_0 + c_1(x_n - x_0) + c_2(x_n - x_0)(x_n - x_1) + \cdots + c_n(x_n - x_0)(x_n - x_1) \cdots (x_n - x_{n-1}) \end{cases} \quad (15)$$

Then formula (15) is the general form of the Newton interpolation polynomial, available

$$\begin{aligned} c_0 &= y_0 = f(x_0) \\ c_1 &= \frac{y_1 - y_0}{x_1 - x_0} \end{aligned} \quad (16)$$

We call  $c_1$  the first-order difference quotient at  $x_0, x_1$ , denoted by  $f(x_0, x_1)$ , and bring it into the equations, available

$$c_2 = \frac{y_2 - y_0 - f(x_0, x_1)(x_2 - x_0)}{(x_2 - x_0)(x_2 - x_1)} = \frac{f(x_1, x_2) - f(x_0, x_1)}{x_2 - x_0} \quad (17)$$

The above formula is the second-order difference quotient at  $x_0, x_1, x_2$ , denoted as  $f(x_0, x_1, x_2)$ . Similarly, the  $n$ -degree difference quotient is

$$c_i = f(x_0, x_1, \dots, x_i) = \frac{f(x_1, x_2, \dots, x_i) - f(x_0, x_1, \dots, x_{i-1})}{x_i - x_0} \quad (18)$$

The Newton interpolation polynomial can be derived as:

$$\begin{aligned} N_n(x) &= f(x_0) + f(x_0, x_1)(x - x_0) \\ &\quad + f(x_0, x_1, x_2)(x - x_0)(x - x_1) + \cdots \\ &\quad + f(x_0, x_1, x_2, x_3)(x - x_0)(x - x_1)(x - x_2) \\ &\quad + f(x_0, x_1, \dots, x_i)(x - x_0)(x - x_1) \cdots (x - x_{n-1}) \end{aligned} \quad (19)$$

To facilitate the calculation of the coefficients, the difference quotient list (Table 1) is calculated and the interpolation polynomial is calculated according to Equation 19.

**Table 1.** Newton interpolation method difference table

Variable	Difference quotient				
$x_0$	$f(x_0)$				
$x_1$	$f(x_1)$	$f(x_0, x_1)$			
$x_2$	$f(x_2)$	$f(x_1, x_2)$	$f(x_0, x_1, x_2)$		
$x_3$	$f(x_3)$	$f(x_2, x_3)$	$f(x_1, x_2, x_3)$	$f(x_0, x_1, x_2, x_3)$	
$x_4$	$f(x_4)$	$f(x_3, x_4)$	$f(x_2, x_3, x_4)$	$f(x_1, x_2, x_3, x_4)$	$f(x_0, x_1, x_2, x_3, x_4)$
$\vdots$	$\vdots$		$\cdots$		$\ddots$

### 3.2. Research on fitting algorithm

When solving the unknown function based on the sampled value, in order to reduce the sampling error and avoid accidental bad values affecting the result, it is necessary to use the original data as much as possible to avoid using part of the data to obtain the result.

According to the high-speed signal analysis model, the form of the function to be measured is:

$$y = \frac{E}{\omega_n L} e^{-\alpha t} \sin(\omega_n t) \varepsilon(t) \quad (20)$$

If this function is directly fitted, the amount of calculation is huge, and the result is also difficult to guarantee. Observe that its positive envelope function is:

$$y = \frac{E}{\omega_n L} e^{-\alpha t} \quad (21)$$

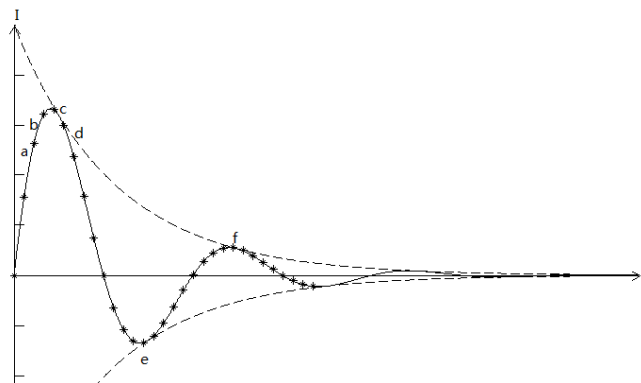
This function is simple in form and difficult to fit. This determines the idea of signal fitting:

1. The maximum point of the signal is on both the curve and the positive envelope. The envelope function expression can be obtained by the optimal square approximation method. At the same time, after calculation, it can be seen that the two envelope curves are symmetric about the x-axis to include more original information and reduce the error. The absolute value of the minimum value can be used to fit the envelope function together with the maximum value.

2. The time interval between adjacent extreme points (maximum and minimum values) is a half-cycle of the sine function. The period of the extreme point can be used to find the period of the sine function.

3. The fitting function of the high-speed signal can be finally obtained from the envelope function and the sine function period.

At the same time, due to the high signal rate and relatively few sampling points, directly solving the extreme points may cause large errors. In this case, the Newton interpolation method can be used to fit the interval function curve with the data near the extreme points to solve the maximum value. . Figure 3 is a schematic diagram of the sampling result of the pulse signal. Taking the diagram as an example, the details of the algorithm idea are introduced.



**Figure 3.** Schematic diagram of pulse signal sampling

#### 1. Find the sampling extreme point

Use the successive comparison method to find all extreme points (including maximum and minimum values) of the sampled data, as shown in the c, e, and f points.

## 2. Calculate the actual signal extreme points

The extreme point of the actual signal is calculated by Newton interpolation. Take the first extreme point in Figure 3 as an example. Taking the sampling extreme point  $c$  and the neighboring two point's  $b$ ,  $d$  and the larger of the two points a little apart, the four points are the sampling points closest to the maximum point of the actual signal. For cubic interpolation, the fitting function in this extreme region can be obtained as follows:

To facilitate the calculation, according to the sampling time sequence, set the four points to  $a$ ,  $b$ ,  $c$ , and  $d$ , and the sampling time interval is 1, so that the four points can form a third-order differential quotient (2):

**Table 2.** Newton third-order differential table

Order	Difference quotient			
0	$a$			
1	$b$	$b - a$		
2	$c$	$c - b$	$a - 2b + c/2$	
3	$d$	$d - c$	$b - 2c + d/2$	$-a + 3b - 3c + d/6$

The interpolation polynomial is:

$$\begin{aligned}
 N(t) &= a + (b - a)t + \frac{1}{2}(a - 2b + c)t(t - 1) + \frac{1}{6}(-a + 3b - 3c)t(t - 1)(t - 2) \\
 &= a + pt + \frac{q}{2}t^2 + \frac{r}{3}t^3
 \end{aligned} \tag{22}$$

Among them:

$$\begin{aligned}
 p &= \frac{1}{6}(-11a + 18b - 9c + 2d) \\
 q &= 2a - 5b + 4c - d \\
 r &= \frac{1}{2}(-a + 3b - 3c + d)
 \end{aligned} \tag{23}$$

When  $r \neq 0$ , the function has two extreme points:

$$t_{1,2} = \frac{-q \pm \sqrt{q^2 - 4rp}}{2r} \tag{24}$$

There is one and only one extreme point  $t_x$  in the interval, and the value of the interpolation function corresponding to this point is considered as the extreme value of the actual signal.

## 3. Calculate the actual extreme point corresponding time

Due to the "first sampling, post-triggered" strategy, it is necessary to determine the zero point of the trigger time, that is, the time point at which valid data is sampled. From the first extreme point starting point by point query, the first point where the first digit is zero is the time zero point, the number of sampling intervals between this point and a point is  $t_s$ , the sampling interval between a point and the extreme point It is  $t_x$  in Formula 24. So the time corresponding to the actual first extreme point is:



$$t_{ml} = (t_s + t_x)\Delta t \quad (25)$$

This method can find the corresponding time  $t_m$  of other actual extreme points.

#### 4. Fitting the envelope curve

Using the extreme points obtained in 3, the envelope curve is fitted using the optimal square approximation method. Since the envelope function is  $y = \frac{E}{\omega_n L} e^{-\alpha t}$ , Computation is complicated and transformation is performed first.

Make  $E_0 = \frac{E}{\omega_n L}$ , easy to make.

$$\ln y = \ln E_0 + (-\alpha t) = c_0 + c_1 t \quad (26)$$

Among them,  $c_0 = \ln E_0$ ,  $c_1 = -\alpha$ , the normal equations are:

$$\begin{bmatrix} \sum 1 & \sum t_m \\ \sum t_m & \sum t_m^2 \end{bmatrix} \begin{bmatrix} c_0 \\ c_1 \end{bmatrix} = \begin{bmatrix} \sum \ln y_m \\ \sum t_m \ln y_m \end{bmatrix} \quad (27)$$

Among them,  $t_m$  is the actual extremum corresponding time,  $y_m$  is the actual extremum value. Solving the normal equations can calculate  $c_0$ ,  $c_1$ , and find  $E_0$  and  $a$ .

#### 5. Calculate the sine function period

The difference between the corresponding actual extreme points corresponds to a half cycle of the sine function, from which the angular frequency can be calculated. Finally get the fitting function  $y = E_0 e^{-\alpha t} \sin(\omega_n t) \varepsilon(t)$ .

#### 6. Solve important parameters

According to the fitting function, the maximum  $U_m$  of the signal, the leading edge of the pulse (from 0.1  $U_m$  to 0.9  $U_m$ ), the zero time of the triggering time and the zero time of the sampling time are easily obtained.

### 4. Implementation and Simulation in FPGA

The development of the FPGA uses the Verilog HDL language. Although the hardware description language is similar to C language in terms of syntax and expression, the implementation principle is completely different. When writing Verilog HDL programs, always have hardware ideas and consider ways to implement statements in the circuit (4). In the design of this program, pay special attention to three aspects:

#### (1) Control of timing and use of trigger conditions

Verilog HDL language is suitable for dealing with sequential logic. In this program, the sequence logic does not use much, but it also needs special attention to the design of trigger conditions. Such as in the design of the data processing part of the algorithm, you can use the always procedure statement to achieve, then design a read data associated with the cumulative variable  $i$  and the flag variable  $flag$ , when the read data is complete, the flag variable  $flag$  flip, this variable can be As a trigger condition for level-sensitive always statements, the form is always @ (flag).

#### (2) Use of Parallel and Serial Statements

Parallel processing is one of the features of FPGAs, but it is also prone to timing logic errors. Special attention should be paid to the timing in the program design. The blocking assignment statement should be used correctly to avoid timing errors.

#### (3) Use of hard nuclear resources

The data processing part of the program uses more multiplication and division operations, but less floating point data. Because the implementation of the operation consumes a lot of logic resources, it is necessary to call the embedded hard DSP resources as much as possible to complete the multiplication and division operations. (5)

After the program is compiled, it needs to be further simulated to verify the calculation effect of its design logic and algorithm. The simulation method is simple and easy: use Matlab to generate analog sampling data according to the theoretical sampling frequency (700MSPS) and real signal parameters (frequency, amplitude, envelope, etc.), and store the data in the FPGA, run the program to process the data, and then read result. Comparing the fitting result with the real signal, it is found that the fitting effect is good, the error is small, and the result can reflect the characteristics of the sampled signal.

## 5. Conclusion

This paper first analyzes and establishes the high-speed pulse signal model, then uses Newton interpolation to fit the curve equation, solves the important parameters of the discrete signal, and then studies the FPGA implementation of the algorithm. Finally, by using Matlab simulation experiments, the signal processing is realized. It is verified that the algorithm has a good fitting effect and the error is small. It shows that the Newton interpolation method is feasible in the processing of high-speed sampling data, which provides reference and reference for the research of high-speed sampling data processing.

## References

- [1] Li Jian-guang. Study on the fit method of characteristics curve of wear resisting mine drainage pump [J]. Journal of Hebei University of Engineering (Natural Science Edition), 2008 (01): 45-48.
- [2] Lin Xiao-yu, Jiang Yue-wen, Wen Bu-biao. Determination of optimal wind power acceptance level based approaching and Newton interpolation [J]. Power System Protection and Control, 2015, 43 (18): 12-17.
- [3] Chen Xi-liang, Zhang Jia-hua. Application of Newton Interpolation on Fitting the Red Edge of Vegetation [J]. Hubei Agricultural Sciences, 2016, 55 (07): 1828-1831+1835.
- [4] Yan Qi, Yan Jiang-tao, Ma Xi-hong. Design and Implementation of High-Speed Signal Acquisition Circuit Based on FPGA [J]. Research and Exploration in Laboratory, 2015, 34 (04): 124-128.
- [5] Liu Ji, Yang De-wei, Li Li-jing, etc. Design of High Speed Signal Process System Based on FPGA [J]. Electronic Design Engineering, 2011, 19 (02): 158-161.