CONTROL AND ANALYSIS SYSTEM FOR DIGITAL FEEDBACK IN HLS

M.Meng, J.H.Wang, Z.R.Zhou, Y.B.Cheng, Y.L.Yang

NSRL, School of Nuclear Science and Technology, University of Science and Technology of China, Hefei 230029, P. R. China.

Abstract

In HLS we employ the TED FPGA based processor for digital feedback system. To make feedback system work better and more easily, we developed the control and analysis system based on Matlab chiefly. The system do jobs as following: sampling data online and finishing its analysis; calculating FIR filter parameters and generating .CSV (format for FPGA) file to get the best gain and phase flexibly according to different beam working point; simulating the beam changes in different feedback gain and other stations to check whether the system work properly.

INTRODUCION

HLS uses the octopole magnet to help injection at the present time, but octopole magnet will cause nonlinearity and change the dynamic aperture, after the update plan of HLS finished, octopole magnet cannot be used anymore, then the feedback system should be used to suppress beam instabilities and improve injection efficiency.

Compared to analog feedback system, digital feedback system is more simple and flexible, also it's more reliable because of less affected by transport noise. We can use the fir filter in processor to get more feedback effect. So developing a control system fitting digital feedback for doing some analyzing is very useful for taking full advantage of digital feedback.

CONTROL SYSTEM DESCRIPTION

The structural diagram of digital feedback system is shown in Fig. 1.

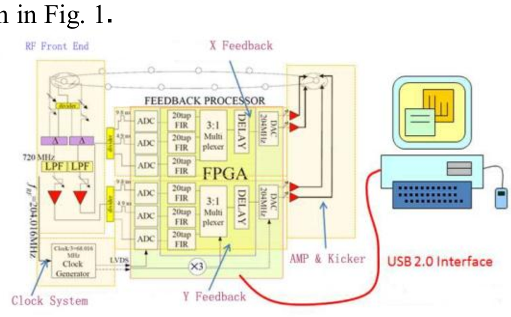

Figure 1: Structure block of digital feedback.

After the beam signal is processed by RF front-end, it is sampled by 6 ADC[1][2] in the feedback processor .The FPGA inside handles data and outputs it to kicker through DAC. During the process, command and data is transported in/out by USB2.0 interface when we need to change the system.

In the processor made by TED the USB chip used is CY7C68 series, it's the first micro processor controller interface chip with USB2.0 protocol inside in the world. The max transform speed can be 480 Mbps. The interface is Hot Swap so that it can protect the instrument, and the control can be very flexible too.

In the past the control soft we use is presented with machine. It is a command line programme and only contains some basic functions so that using of it is also very inconvenient while we need to debug the feedback machine; On the other hand the updated data which sampled by ADC is too much to analyze online, while sometimes we need to know some parameters of beam online, so we can develop an new control system to change the situation.

To decrease the amounts of work as far as possible, we take use of DLL part of the programme used before after reprogramming it. DLL (Dynamic Link Library) packages some executable codes and data in it while supplying us with calling in/out interface. The existed DLL is already compiled with C language based on driver of CY7C68 chip, so using it we don't have to write driver again, that is not so easy. In our work we rewrite sublayer communication functions; they call basic I/O functions to control the system by CY7C68.

The control system is carried out with Matlab. Matlab has a big advantage on data analyzing. In new edition Matlab such as 2008, support of calling function in DLL is more perfect. The contrast of data type used in C and MATLAB is in Fig. 2

C type	(unsigned)char, byte, Short, …	float	double	char *
Matlab	(u)int[SIZE]	single	double	string

	needle pointer			
C type	short *, int *, long *, …	float *	double *	struct
Matlab	int[SIZE]Ref	singleRef	doubleRef	libstruct

Figure 2: Data type compared in c and Matlab.

To make the structure of control system simple, we put most of the operation in DLL, so when using Matlab soft we just have to know the name of functions in DLL and what parameter we should pass to it. The soft uses MATLAB GUI (Graphical user interface) to make users easily operate it. Fig. 3 shows the panel of control GUI

*Work supported by National Natural Science Project (10175063)
mengming@mail.ustc.edu.cn

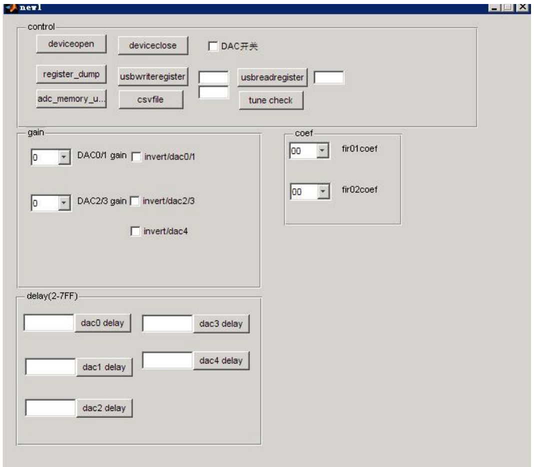

Figure 3: Control system with Matlab.

Its working chain is as follows: calling loadlibrary function to load DLL, calling calllib function to use functions in DLL, among which registerwrite, registerread function is for register reading and writing, adcmemoryupload function is for ADC data operating. First we should call deviceopen function to open device and establish connect; for DAC, address 0x8022 determines its on/off state.

0x8008,0x802A,0x800A,0x802C,0x800C are the output delay of DAC0-4，the last three bits of 0x8020 determine inverting state of 3 DAC，0x8012,0x8014 are the gain of DAC, 0x8004 0x801A determine the fir filter parameters; For ADC, 0x805C assigns the uploadsize of updating data, of which minimum value is 128K, 0x8058 are in charge of updating by writing 2,1,4 to it in queue, 0x805a shows the change of ADC station while translating data. Following the steps, we finish the control system.

DATA ANALYSING

With the data sampled by control system, we can do some analyse to get some important parameters of beam.

Fir Coefficient

In the processor, we use FIR filter in FPGA, the filter can separate unwanted DC constituent and other noise at cyclotron frequency, avoiding the saturation of ADC. It also has to exert right phase and gain on signal.

FIR can be carried out easily when hundreds of Mb data should be processed in some ns time. In the FPGA, we employ least square method to design FIR filter [3].

FIR can be expressed as

$$y(n)=\sum_{i=1}^{N} a_{k_i} x_{n-k_i} \qquad (1)$$

x_{k_i} is input data， N is filter tap number， And the number k turn of beam signal has the form of

$$x(k)=\sum_{m=0}^{M} A^{(m)} \sin(k\phi^{(m)}+\Delta^{(m)}+\psi^{(m)}) \qquad (2)$$

In it m=1 means horizontal movement， m=2 means vertical movement， $\phi^{(m)}$ is phase offset each turn , in direct proportion with freedom frequency of oscillation， $\Delta^{(m)}$ is phase offset each turn because of tune shift.

After transformation, the coefficient of fir filter should be

$$a_{kj}=\sum_{m=0}^{M} G^{(m)}(D_{i_m,j}\cos\zeta^{(m)}+D_{i_{m+1},j}\sin\zeta^{(m)}) \qquad (3)$$

$G^{(m)},\zeta^{(m)}$ means gain and phase shift of feedback. D is transforming matrix. Now we can develop soft with Matlab GUI to calculate the FIR coefficient quickly.

Fig. 4 shows the interface of FIR getting system.

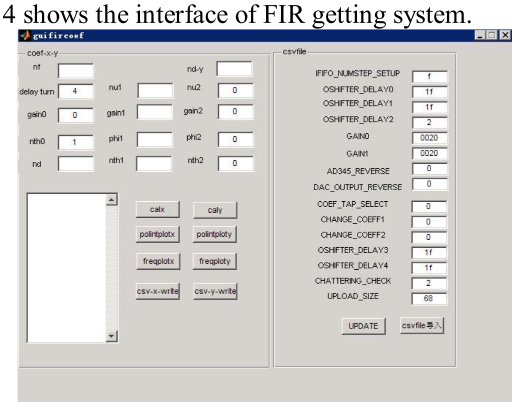

Figure 4: Fir coefficient calculating soft.

With it, we get the 16 tap, 20tap coefficient of X direction and 10 tap, 12 tap coefficient of Y direction.

Fig. 5 shows the contrast of Amplitude-frequency characteristic for X, Y direction.

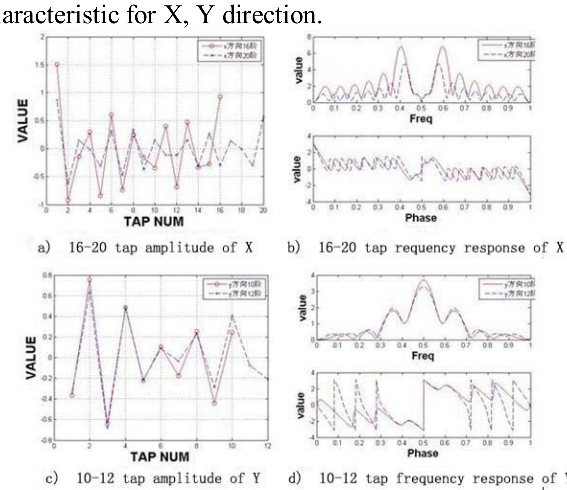

Figure 5: Result of x and y Fir coefficient.

Tune

While calculating tune with beam data, we need to get better frequency resolution with less points. Normally we use NAFF method with 256 points, and the resolution is up to 0.00005[4].

Beam data is a discrete sequence f(n)

$$f(n) = m...m + N - 1 \quad (4)$$

We use the next formula to calculate tune, $\chi_n = 1 + \cos(\frac{\pi \cdot n}{N})$, $\nu^k_{m,N}$ is tune, we set the range of $\nu^k_{m,N}$, then find the max value of $I(\nu^k_{m,N})$, there the $\nu^k_{m,N}$ is what we want.

$$I(\nu^k_{m,N}) = \sum_{n=m}^{m+N-1} f(n)\exp(-i2\pi n \cdot \nu^k_{m,N})\chi_{m-n} \quad (5)$$

The key step to carry out the method is using next iteration formula

$$x_{k+1} = x_k - (x_k - x_{k-1}) \times \frac{f(x_k)}{f(x_k) + f(x_{k-1})} \quad (6)$$

So we can get tune with some beam data in Fig. 6

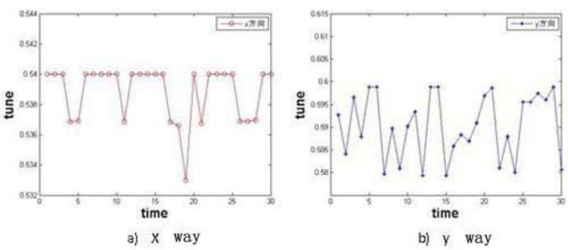

Figure 6: X and y tune with NAFF.

While using this method, we find that if we set a too small step to find max value of $I(\nu^k_{m,N})$, the processing will take too much time, so we may try FFT transformation directly to find tune. FFT has the formula as next

$$y_i = \sum_{k=0}^{n-1} x_k e^{-j\frac{2\pi}{n}ik} \text{ ------ } i = 0,...,n-1 \quad (7)$$

We get a section of data, like 512 points or more one time, take FFT transformation of it then we try to find the point where amplitude is max near value of tune we normally know, because tune shift couldn't be too much. The code number of point we found corresponds with tune. In Fig. 7 we get with FFT method

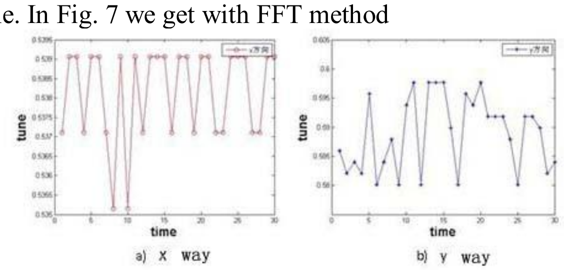

Figure 7: Tune of x and y with FFT.

Phase-Space

Phase-Space differential form of beam position [5] can be expressed as

$$x'(s) = \frac{-1}{\beta(s)}[\alpha x(s) + a\sqrt{\beta(s)}\sin(\frac{ds}{\beta(s)} + \phi_0)] \quad (8)$$

In experiment, we cut 50000 turns from sampled beam data, compute one time every 500 points. Fig. 8 shows the variation tendency of phase-space.

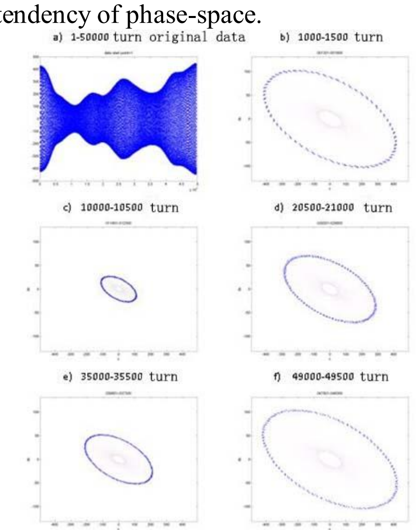

Figure 8: Phase space block with sampled data.

SUMMARY

The control system we develop with MATLAB GUI is now working well for digital feedback system. Besides basic control of processor, it's also very flexible, because the foundation statement used in Matlab programme is very simple, with the function supplied, everyone can bring about very complex function if it is achieved through changing value of FPGA register.

With the soft, we sample beam data and analyze it online. The tune, phase-space can be calculated to take a monition of beam to help debugging digital feedback system.

REFERENCES

[1] J.H. Wang, W.M. Li, L. Wang, Z.P. Liu, Experiment of Transverse Feedback System at HLS. PAC09.

[2] Z.R. Zhou, J.H. Wang, B.G. Sun, Lin Wang, Ming Meng, e.g. Commissioning of the Digital Transverse Bunch-By-Bunch Feedback System for the HLS. PAC09.

[3] T. Nakamura∗.Time Domain Least Square Fitting Method for FIR filters. PAC 29.27.BD

[4] Caussyn.D.D, Ball.M, Brabson.B, et al. Experimental studies of nonlinear beam dynamics[J]. Phys Rev A. 1992 Dec 15;46(12):7942-7952

[5] Y.L.Yang,J.H.Wang,Z.P.Liu, The Measurement of Tune and Phase Space at HLS, PAC'05, USA,2005