SOLVEMENTS OF THE ASYNCHRONIZATION BETWEEN THE BPMS AND CORRECTOR POWER SUPPLIES IN RCS OF CSNS*

Li Mingtao†， Chinese Spallation of Neutron Source(CSNS), Institute of High Energy Physics, Chinese Academy of Scieneces,Dongguan, China.

Liu Yudong, An Yuwen， Chinese Spallation of Neutron Source(CSNS), Institute of High Energy Physics, Chinese Academy of Scieneces,Dongguan, China.

Abstract

This paper studies the possible solvement of the asynchronization between the BPMs and Corrector Power Supplies in RCS of CSNS, to increase the accuracy of the response matrix measurement, the obit correction, or other commissioning task.

INTRODUCTION

The accelerator of Chinese Spallation of Neutron Source (CSNS) is mainly composed of a Linac and a Rapid Cycling Synchrotron (RCS), showing in Figure 1. [1] In the first phase it should provide output pulsed proton beam with a mean power of 100 MW. In the second phase after large upgrade, it would have an output beam power of 500 MW. After 4 DTL Tanks, the Linac injects an 80MeV H^- beam to the RCS injection point with a repetition rate of 25 Hz. In multi-turn injection in 0.2Ms, over the stripping foil, two electrons would be stripped out. H^- beam turns to proton beam. The RCS accelerates the proton beam from 80 MeV to 1.6GeV in 20 Ms. In the 20Ms, the proton beam transfers over the RCS about 20000 turns.

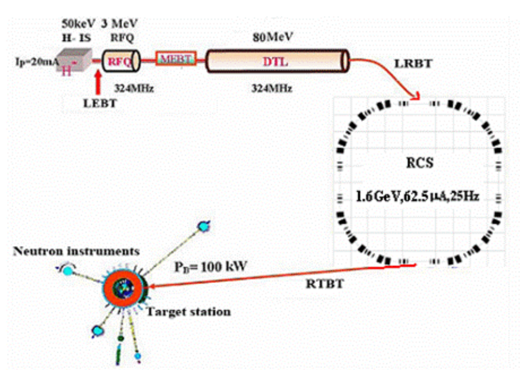

Figure 1: Layout of Chinese Spallation of neutron Source (CSNS).

The beam commissioning of the Linac began in 2015, and would be finished in May 2017. Because the klystron for DTL4 is not available in several month time, the beam commissioning of RCS will adopt the 60meV injection and 1.6GeV extraction mode, using the other three available DTLs.

* Work supported by National Natural Science Foundation of China (Grant NO. 11405189)
†limt@ihep.ac.cn

Beyond the commissioning plan, the asynchronization between BPMs and Corrector Supplies in RCS appeared an unexpected challenge at last October. This asynchronization is discussed as below. The 32 BPMs in RCS can only make COD measurement in every 2^N (N=7, 8, 9, 10) turns. But on the same time each one of the 34 corrector power supplies in RCS have a 21 bit Register. Then each power supply has 21 set points during the accelerating process in one period, 40Ms. Based on the design, these set points are set into the corrector with regular time intervals, that is, 1Ms. In 60 MeV injection mode, for example, if BPM's COD data is taken in every 1024 turns, each BPM takes the first COD data at 2.1Ms in one period, and it takes the second one at 3.7Ms, and so on. It is obviously that the BPMs and corrector supplies have a large mismatch. This mismatch might ruin the obit-correction and response matrix measurement.

After many tedious discussion with beam diagnostics system and power supply system to neutralize the negative the effect of these hardware design defect, it is realized that, to solve this kind of asynchronization, any improvements in hardware instruments is impossible. The only possible solution is to turn to the development of beam commissioning software.

POSSIBLE SOLVEMENT

In CSNS, the commissioning software of CSNS are developed based on the XAL framework, which was first developed by SNS. [2] The XAL beam commissioning software control all hardware throw the Experimental Physics and Industrial Control System (EPICS). [3] All possible solutions should be considered within these two software levels. The first feasible solution has been found is the so called "Pick out" method.

In the beginning, trying the best to achieve the goal of synchronizing the BPMs and corrector power supplies of RCS, the "Pick Out" method seems quite simple: 1, execute the COD measurements in every period more frequently; 2, choose out the COD data that are taken over every integral Ms; 3, now, as the COD data has been "picked out", beam commissioning could utilize these selected BPM COD data for orbit-correction, obit-display, response matrix measurement, and so on.

As at the beginning of CSNS RCS bean commissioning, only 60MeV H^- injection/1.6GeV extraction mode could be used, thus the time consumed by the beam transferred over the RCS in every turn could easily be calcu-

ISBN 978-3-95450-182-3

lated out. Table 1 shows the time of the beam transferred over the RCS in every turn. Only some typical turns are selected to be listed in the table to show some necessary physical information.

Table 1: Time Consumed of Beam Transferred Over RCS in Each Turn

Turn	Time	Total Time	Frequency
	Ms	Ms	Hz
0	0	0	---
1	2.21E-03	2.21E-03	452526
2	2.21E-03	4.42E-03	452526
463	2.07E-03	1.00	482902
982	1.79-03	2.00	559879
1589	1.52E-03	3.00	656949
2296	1.32E-03	4.00	755397
3097	1.18E-03	5.00	845162
3982	1.08E-03	6.00	922518
4938	1.01E-03	7.00	986888
5952	0.96E-03	8.00	1039285
7013	0.92E-03	9.00	1081383
8112	0.90E-03	10.00	1114949
9240	0.88E-03	11.00	1141556
10394	0.86E-03	12.00	1162627
11565	0.85E-03	13.00	1179192
12751	0.84E-03	14.00	1192158
13948	0.83E-03	15.00	1202190
15154	0.83E-03	16.00	1209805
16367	0.82E-03	17.00	1215381
17584	0.82E-03	18.00	1219184
18805	0.82E-03	19.00	1221399
20027	0.82E-03	20.00	1222126

In the first design of RCS BPMs, they are set to execute the COD measurement every 1024 turns of the beam transferred over the RCS. Fortunately, they have the margin to execute the COD measurement in less turns. Because the hardware designing limitation, the turns must be 2^N (N=1, 2, 3 …14), as mentioned before. If at least 20 COD measurements are executed in one period, 40Ms, the N must be less than 11. On the same time, too small N also couldn't be selected. Because if it is too small, COD measurements are executed too frequently. This problem would results in the electronics of the BPMs overwhelmed. Second, it would also occupy the precious bandwidth resource of the network. Third, if the COD data are taken in not enough turns, more data deviation might be introduced in the COD data processing. Therefore, the available N must be larger than 6. So N could be 7, 8, 9, or 10. In the last COD measurement in each period, the remaining turns before extraction might be less than 2^N, it is executed in these remaining turns.

After comprehensive consideration of the advantages and disadvantages, N is chosen as 8. That is, COD measurement is executed every 256 turns of the beam transferred over RCS. As the total cycling turns within each period, as showed in Table 1, is 20027, so the COD measurement is executed 79 times. The second one of the 79 COD data is taken from 257th to 512ed turns. Converted into time, it is from 0.56Ms to 1.10Ms, because it crosses 1Ms, so this data is chosen as the first available data. The other 19 data are chosen in the same way. Then the 20 BPM COD data that have been picked out are listed in Table 2.

Table 2: 20 COD Data Picked Out From 79 Data

Turn	Picked out NO.
463	2
982	4
1589	7
2296	9
3097	13
3982	16
4938	20
5952	24
7013	28
8112	32
9240	37
10394	41
11565	46
12751	50
13948	55
15154	60
16367	64
17584	69
18805	74
20027	79

Now the general principle has been explained in detail. The challenge is how to put this principle into practise.

The XAL commissioning software in CSNS are transplanted from SNS. The biggest difference between the two facilities is the RCS of CSNS and the cumulated ring of SNS. The beam commissioning of the RCS not only has the DC mode as SNS, but also AC mode. The most difficulty in the software transplanting task is to develop the DC orbit-correction of SNS, which is almost the most complicated software in XAL, to the AC orbit-correction of the RCS. After years of hard work, this AC orbit-correction software is well prepared. Before the asynchronization problem occurred, the AC orbit-correction software had passed the test on virtual accelerator which is a unique property in XAL. The most direct idea is to modify the AC orbit-correction software. But after some attempt, it is realized that modifying such sophisticated program to realize the "Pick out" feature is a huge task. This approach is not feasible, because it will inevitably lead to too much time consumption and reduce the software running reliability. As a conclusion, to avoid the modifying of the AC orbit-correction software, the only sensible way is to realize the "Pick out" feature in other program outside the AC orbit-correction software.

XAL framework communicates information with EPICS throw thousands of Process Variables (PV). The BPM COD data are PVs, so do the corrector power supply set points. In the control system of CSNS, one PV naming convention is adopted. In experimental physics division, the PV names start with "*EXP_*", while in accelerator division, the PV names start with "*ACC_*". But in the beginning of commissioning software development, the PV naming convention of SNS is adopted. In this PV naming convention, PVs have such form "*RCS_Diag:R1BPM02:xAvg*", take one BPM PV as example. These PV names are fixed in the lattice files. To solve the contradiction, all the PVs used by the beam commissioning software have two names. The second set of PVs are only the aliases of the first one. Before the asynchronization problem appears, these two set of PVs cause a lot of trouble in the communication between different systems. But now this problem turns into an advantage to solve the asynchronization. Now based on these hardware and software conditions, we have two choices to achieve the "Pick out" purpose.

(1) Pure Java solution:

XAL framework is Java development project integrated with JCA/CAJ which is the EPICS channel access client library for Java. So some software could be developed to implement the "Pick out". Specifically, two software are needed. First is a PV server, which creates PVs' channel. In this server, the PVs' channel like "*RCS_Diag:R1 BPM02:xAvg*" are created. In the same time the second set of PVs are no longer the aliases of the first ones. The function of the second software is following the role described by Table2, to "Pick out" the 20 bit BPM COD data from the 79 bit BPM COD data from the channels of the first set of PVs, like "*ACC_RCS_DIAG:R1BPM 01:X:average*" ; and write these twenty data into the channels of the second set of PVs, like "*RCS_Diag:R1 BPM02:xAvg*". Once the server is running, after loading the accelerator lattice files, the beam commissioning software could get and process the target 20 bit BPM COD data taken out from the second set of PVs as double arrays. So the AC orbit-correction software itself doesn't need any modification.

(2) EPICS solution:

The pure Java solution sounds easy, but its data processing speed and long-time reliability is questionable. This might bring out many troubles in beam commissioning. The more reliable solution, is to process the "Pick out" procedure in low-layer development in EPICS. Fortunately, the EPICS is quite convenient to realize the "Pick out" function.

In EPICS, the aSub (Array Subroutine) record [4] is very powerful to realize many kinds of functions. In this asynchronization condition, the solution of EPICS is also quite simple. In the SoftIOC created by EPICS, one aSub record will be added to the "*/softIoc/csnsApp/Db/BPM .db*" file:

```
"record(aSub," Pick out" ){
    field(SCAN, " 1 second" )
    field(SNAM," pickout" )
    field(FTA," DOUBLE" )
    field(NOA,79)
    field(INPA," ACC_RCS_DIAG:R1BPM01:X:
  average ")
    field( FTVA, " DOUBLE" )
    field(NOVA,20)
    field(OUTA," RCS_Diag:R1BPM02:xAvg" )
}".
```

The "pickout" represents the subroutine to implement the "Pick out" task, which must be added to the "/softIoc/csnsApp/src/dbSub.c" file developed in C:

```
"static int pickout(aSubRecord *prec) {
    long i;
    double *a;
  static int[]
num={2,4,7,9,13,16,20,24,28,32,37,41,46,
  50,55,60,64,69,74,79};
    a = (double *)prec->a;
    for (i=0; i<20; i++) {
  ((double *)prec->vala)[i] = ((double *)prec->a)
  [num[i]];
      }
    return 0;
}"
```

After this processing, the second set of BPM PVs are also no longer the aliases of the first ones. Once the SoftIOC begins running in the background, the "*RCS_ Diag:R1BPM02:xAvg*" channel could also process the 20 bit BPM COD data for orbit correction, response matrix measurement, sand so on.

Obviously, this method is proved to be more robust than the pure Java solution. So in the beam commissioning of the RCS of CSNS, this solution has been adopted.

CONCLUSION

To eliminate the error brought out by the asynchronization problem between BPM's COD measurements and the corrector power supply set points in RCS of CSNS, the "Pick out" method has been proposed. It might compensate this problem in beam commissioning as much as possible, based on the current hardware conditions. Another "inverse interpolation" method, which is evaluated as more reliable, is also under consideration. But to be frank, the software improvements mightn't make up the hardware designing defect completely. The best has been done, and the effect will be tested in the coming RCS beam commissioning. Other possible solutions, if possible, will also be considered in the beam commissioning process.

ACKNOWLEDGEMENT

We thank the enlightening discussions about EPICS with M.S. Wu Xuan and M.S. Zhang Yuliang from control group of CSNS. We also thank the discussions about BPM electronics with Prof. Xu Taoguang, Prof. Qi Xin and M.S. Liu Yuntao. These discussions are all very helpful for the final solvement of the problem.

REFERENCES

[1] Physics Design and Technology Development of CSNS Accelerator, http://csns.ihep.ac.cn/english/image/CSNS%20Accelerator.pdf
[2] J. Galambos *et al.*, "XAL Application Programming frame-work", presented at Proc. of ICALEPCS 2003, Gyeongju, Korea.
[3] http://www.aps.anl.gov/epics
[4] https://wiki-ext.aps.anl.gov/epics/index.php/RRM_3-14_Array_Subroutine