KEKB Accelerator

The KEKB filling-pattern generating system

Eiji Kikutani*, Kazuro Furukawa, Masaaki Suetake, and Makoto Tobiyama

Accelerator Laboratory, High Energy Accelerator Research Organization (KEK), Oho 1-1, Tsukuba, Ibaraki 305-0801, Japan

*E-mail: kikutani@post.kek.jp

Received September 21, 2012; Accepted December 12, 2012; Published March 25, 2013

KEKB is a high-luminosity electron–positron collider, consisting of an 8 GeV electron ring and a 3.5 GeV positron ring. To achieve the required luminosity, beams are squeezed to become very sharp and a large number of particles are stored in the rings. In order to operate the KEKB rings smoothly under a very high-current environment, we developed a user-friendly filling-pattern generating system. With the help of high-speed bunch-current measuring systems, it contributed to the stable operation of the collider rings.

1. Introduction

KEKB is a double-ring electron–positron collider at KEK, Japan, constructed for an experiment studying \( B \)-meson physics. The accelerator system consists of two collider rings, an 8 GeV electron ring and a 3.5 GeV positron ring, as well as a linear accelerator complex (referred to as “the injector” hereafter), which provides the rings with electron and positron beams. The accelerator system has no damping rings or accumulation rings. Operation of KEKB started at the end of 1998, and tentatively stopped in the middle of 2010 to upgrade it to a higher-performance machine, SuperKEKB.

The physics motivation required a collider of very high luminosity to produce a tremendous number of \( B \)- and \( \bar{B} \)-meson pairs. In order to achieve the required luminosity, the collider design [1] is very challenging: the spot sizes of the colliding beams must be very small compared with colliders so far and, additionally, the number of particles circulating in the rings should be very large. The usual way of distributing these particles is into some number of beam bunches. However, as is well known, coherent oscillations of the bunches, which are known as coupled-bunch instabilities, would occur when many bunches are stored with a short spacing. In addition, for the positron ring, instability due to electron clouds [2] can be a serious problem. Thus, one of the crucial points of the accelerator operation is how we can avoid these instabilities.

The strengths of the instabilities depend on many factors, such as bunch spacings, bunch charges, and the characteristics of the impedance sources in the ring. Among these factors, the filling-pattern of the bunches, namely, which RF buckets are filled with beam bunches and which buckets are not, is very important. Though you can calculate the strengths by utilizing some formulae\(^1\), their prediction is only a rough estimation. A straightforward and promising method is to try a specific filling-pattern in the actual ring and check the machine performance with the pattern. Trial and error enable us to

\(^1\) For example, a code called ZAP is convenient for an estimation of the instabilities; see Ref. [3].
Table 1. Some machine parameters taken over from TRISTAN.

| circumference | RF frequency | harmonic # | RF freq. of the injector linac |
|---------------|--------------|------------|-------------------------------|
| ~3020 m       | ~509 MHz     | 5120       | 2856 MHz                      |

find the best filling-pattern for the maximum luminosity. Naturally, to do this, it is vital to construct a system that realizes the desired filling-patterns with an operator-friendly man–machine communication tool. From this point of view, we developed a filling-pattern generating system, which can realize any desired filling-pattern in a flexible manner.

The two collider rings of KEKB were constructed within the tunnel of the old accelerator TRISTAN (see, e.g., Ref. [4]) and the same injector as TRISTAN was used after a large-scale upgrade. Even though the design concept of KEKB is completely different from that of TRISTAN, many components of the old accelerator were reused. Accordingly, some of the basic parameters shown in Table 1 are the same as or very close to the previous values.

In contrast to these parameters, the number of bunches as well as bunch spacings for KEKB are completely different from the older ones. According to the original design of KEKB [1], it was supposed that all the RF buckets were filled with beam bunches. We reconsidered the RF frequencies of the injector and the rings within the bandwidths of the existing systems, and it was decided that the RF signals of these two systems were produced by frequency-multiplying a common signal.

At almost the same time, the PEP-II collider [5] was under construction at SLAC with the same physics motivation. They chose the RF frequency of the rings to be just $1/6$ of the injector's RF frequency. They were also studying the possible design of the filling-pattern generating system [6].

2. Design concepts

2.1. Background

In this section, we account for the guidelines for designing the filling-pattern generating system by picking up some related parameters. The injector shoots beam bunches at 50 Hz (at maximum), and the number of bunches in one pulse is one or two. The typical bunch-charge in the rings for ordinary physics runs would be about 20 times that of the injector. Then, by properly switching the bucket to be filled pulse by pulse, the desired filling-pattern will be formed. The actual switching can be done by changing the trigger timing of the electron gun of the injector.

Under high-current conditions, the lifetimes of the bunch-charges in the buckets can be different among the stored bunches because of the irregular density of electron clouds (for the positron ring) or of the ion density (for the electron ring). Sometimes buckets in some portion of the rings lose their beam bunches because of instabilities. In order to maintain the desired filling-pattern by trickling charges, it is necessary to watch the bunch-charges throughout the operation. This implies that the system should be equipped with a fast bunch-charge measuring system. However, the evenness of bunch-charges over the ring should not be as critical as some light-source accelerators [7].

Since the highest repetition rate of the injector linac is 50 Hz, the filling-pattern generating system must do the following tasks within 20 ms: (1) measure the bunch-charges in all the buckets, (2) compare the bunch-charges and determine the bucket to be fed by the next pulse (the target bucket), and (3) set the trigger timing of the injector to feed the beam to the target bucket. A crucial point in designing the filling-pattern generating system is ensuring the real-time execution of these tasks.
2.2. Multi-processor scheme

One of the choices in the design might be that these real-time tasks would be executed on hardware or firmware. However, considering the flexibility required in the system, we chose a software-based scheme. In this scheme, the power of the processor on which the timing-critical tasks run should be devoted to execution of the tasks, and an extra processor is prepared for communicating with operators, so as not to disturb the execution of the tasks. We call the first processor the “timing control processor” or simply the “timing processor”, and the second the “OP processor”. This scheme leaves the timing processor completely free from asynchronous tasks resulting from communication jobs.

Two other points in the design are the type of bunch-charge measuring system that is used, and the transfer of the bunch-charge information to the timing processor. The distance between the injector gun and the ring is about 500 m. We use fast-response bunch-current measuring systems that will be described in detail in Sect. 3.1.2. Thanks to the high sampling rate of the analog-to-digital converter (ADC) in the system, we can measure the charges in all the RF buckets in one turn of the beam in the rings. Two bunch-charge measuring systems are prepared for the two collider rings and they are controlled by a third processor, called the “BC processor”.

To transfer the bunch-charge information from the BC processor to the timing processor, we use a dedicated optical fiber network system to avoid using the standard control network of KEKB, since real-time execution is essential, as described above. In addition, this network is used for sending the desired filling-pattern information and various operation-related parameters from the OP processor to the timing processor.

Figure 1 shows the architecture of the system. The distances between the timing processor and the OP processor, and the timing processor and the BC processor are a few hundred meters.
2.3. The environment for control: The EPICS system

The control system of KEKB is based on the EPICS system [8]. Under the EPICS environment, the processors used fall into two categories; one is an “input/output controller” (IOC) and the other is an “operator interface” (OPI). In a typical integration, each component of the accelerator complex is connected to an appropriate VME plug-in, and these plug-ins in a VME sub-rack are supervised by a CPU. For many cases, the operating system of this processor is VxWorks [9], which can manipulate the real-time execution of a program. For KEKB, nearly 200 IOCs are used for the whole accelerator system. The hardware can be accessed through a “channel”, behind which a corresponding (software) driver, called “device support”, is running. The operator-interfacing program, i.e. programs in OPI, calls these device supports for the actual control of each device. The EPICS system provides the users with a standard GUI (graphical user interface) package, called MEDM (Motif Editor and Display Manager) for the operators’ interface. In the KEKB control system, some GUI are built with MEDM and others are built with the SAD system [10].

As described above, we use three processors, and they work as the IOCs. The process playing the central role in the filling-pattern generating system, which is running in the CPU of the timing processor, was not coded as a device support in the EPICS environment, since it should be completely free from communication with operators.

3. Hardware and software components

3.1. Hardware

3.1.1. Timing manipulation. The two collider rings have an RF system with a frequency of \(\sim 509\) MHz. The injector, on the other hand, has an RF system with a frequency of 2856 MHz. In the injection periods, these two frequencies are produced by frequency-multiplying a common RF source of 10.39 MHz. The multiplying factors are \(55 \times 5 = 275\) and 49 for the injector and the rings, respectively.

In principle, which bucket is filled by a pulse of the injector is determined by the relative timing of the gun-triggering-signal to the “the revolution clock” of each ring, which is made by frequency-dividing the ring’s RF signal with the harmonic number. In our case, only \(1/49\) of the revolution clock (reduced revolution clock) is used considering the relation of the RF signals of both systems. By delaying the reduced revolution clock by \(n\) counts of the RF clock, we can fill the bucket address \(b_{ad}\) of

\[
b_{ad} = \text{mod}(49n, 5120),
\]

where 5120 is the harmonic number of the rings. Inversely, we get the delay count \(n(b_{ad})\) for a given bunch address \(b_{ad}\) as

\[
n(b_{ad}) = 49 \times \{209 \times \text{mod}(b_{ad}, 49) + \lfloor b_{ad}/49 \rfloor\},
\]

where \(\lfloor b_{ad}/49 \rfloor\) is the largest integer that does not exceed \(b_{ad}/49\).

The timing manipulation is performed by a circuit schematically depicted in Fig. 2. To vary the timing we use programmable delay circuits packaged as VME plug-ins called “TD4”, which were developed at KEK. The input signal is delayed by counting the clock signal. It accepts a clock signal around 500 MHz and the jitter of the output signal is within 20 ps. The delaying count is programmable through the VME interface. In Fig. 2 they are labeled “TD4s”.
Fig. 2. Schematic block diagram of the frequency manipulation part. The bucket filled by the injector's beam is determined by the relative timing of the trigger signal for the electron gun. The actual timing control is done with a signal whose frequency is 1/49 of the ring's RF frequency. Four sets of the programmable delays “TD4” are used, but only one box is depicted in this figure.

Fig. 3. Detector circuit to create a pulse for the bunch-charge measuring systems.

3.1.2. Bunch-charge monitors. The bunch-charge monitor system is made up of two parts:

- A wide-band detector circuit that forms a train of pulses out of the signal from electrodes exposed to the beam. The height of each pulse is proportional to the charges in the corresponding beam bunch (Fig. 3).
- A board of a 500 MHz-sampling ADC with 8 bit resolution. On the same board, high-speed de-multiplexers and the bulk of the memory chips are loaded (Fig. 4).

A signal from a button-electrode is processed by the detector circuit shown in Fig. 3. The two power-combiners and three cables connecting them make up a broad-band analog FIR filter, which favors a frequency of 4 times the RF frequency of the ring. The output of the filter is multiplied a sinusoidal signal made by multiplying the RF frequency (factor 4) with the double-balanced mixer. The timings of the two input signals are adjusted to be in phase. After excluding the high-frequency component, we can obtain a pulse whose height is proportional to the bunch-charge. In order to
compensate the position dependence, the input for this detector circuit is provided by some of the signals from two button-electrodes in opposite positions.

The board shown in Fig. 4 accepts the output of the detector circuit. The analog-to-digital conversion is performed triggered by the RF signal of the rings, whose frequency is 509 MHz. The output of the ADC is de-multiplexed (by a factor 1/16) and stored in the memory chips embedded on the same board. The bunch charges of all buckets including empty buckets are stored in the memory chips at every moment. When the “data acquisition trigger” signal is accepted, a set of 5120 bunch-charge data is transferred to the memory of the BC processor. The “data acquisition trigger” signal is the injection-kicker signal after an appropriate delay when the ring is at the injection stage. When the ring is at the storage stage, a 1 Hz clock is provided for the “data acquisition”, since the bunch-charges are slow-varying values at the storage state. Figure 5 shows an example of the bunch-charge measured with this system. This bunch-charge measuring system is an application of the KEKB bunch-by-bunch feedback and its related systems [11].

3.1.3. Resource for communication between the three processors. As explained before, we use three processors (timing, OP, and BC) for the filling-pattern generating system. The timing processor is installed near the electron gun of the injector and the OP processor sits at the central control station of KEKB. The distance between them is roughly half a kilometer. The bunch-charge measuring systems as well as the BC processor are installed in one of the sub-control stations of KEKB, which is about 100 m away from the central control station. These three processors are connected with a dedicated network system, as mentioned above.

The network system is a commercial product, the Shared Memory System (model ID: PVME1522), developed by ADVANET Inc. [12]. The key parameters of this system are summarized in Table 2. The transfer speed of this system is fast enough for our purpose. This data-transfer system consists of a number of VME plug-ins, each of which is installed in a sub-rack, in which a processor to be networked is installed (see Fig. 1). The plug-ins are connected with a loop network of optical cables. If one of the CPUs writes new information into some part of the memory, a copy is made in the same address of the memory in the other sub-racks on the network. This copy-action is performed...
Fig. 5. An example of the data obtained by the bunch-charge measuring system. The upper graph (blue) shows bunch-charges in the electron ring while the lower graph (red) shows those of the positron ring. The head part of the positron bunches has lower currents due to the slight lifetime change caused by tune changes.

Table 2. Key parameters of the Shared Memory System.

| Parameter                        | Value                  |
|----------------------------------|------------------------|
| Packaging                        | double-height VME      |
| Connection                       | optical fiber cables   |
| Max. distance between stations   | 1000 m                 |
| Bit rate                         | 250 Mbit/s             |
| Max. # of stations               | 255                    |
| Memory capacity                  | 8 kbytes–24 Mbytes     |

completely by hardware, so that a programmer can code software as if these processors had a common memory area. The system is also equipped with an interrupt function for efficient data transfer actions.

In our case, each of the three processors has its own plug-in with 128 kb memory. A conceptual memory map in each sub-rack is shown in Fig. 6. The memory bank of each plug-in occupies the same address in each memory map. In this area, filling-patterns, bunch-charge information, and various pieces of control information are stored.

3.2. Software

3.2.1. Software in the OP processor. The OP processor takes charge of not only the filling-pattern generation but also communication between the control system and the operators in the main control station of KEKB. The main part of the software in the OP processor is a large device support for the Shared Memory System. Since this memory holds many operation parameters concerning the stored bunches, it is like a dynamic table for operating the storage rings.
Fig. 6. A conceptual memory map based on the Shared Memory System for our application. The memory banks in the three sub-racks occupy the same address as the memory.

Operators can inform the system of desired filling-pattern by two methods, one is the regular-bunch-train method and the other is the filling-pattern-file method. In the regular-bunch-train method, operators type parameters: (1) the number of bunch-trains, (2) bunch-spacing, (3) the number of bunches in a train. The system calculates the bunch addresses to be filled, and lays the series of the addresses in the Shared Memory System. In the filling-pattern-file method, an operator makes a human-readable text file with 5120 lines for each ring, where 5120 is the number of buckets, i.e. the harmonic number of the rings. Each line has two or three columns; the figure in the first column is the bucket-identifying number from 0 through 5119, and the second is a flag for fill/blank, which is specified simply by figure 1 or 0. The third column is an option; if non-equal bunch-charges are desired, the figure in the third column gives a factor (0.1–0.9) specifying the charges relative to ordinary buckets. Operators can edit these files with an appropriate editor or can generate them with some software. The system accepts these pattern-files and creates a table of addresses of buckets to be filled on the Shared Memory System.

Figure 7 shows the operators’ panel developed with MEDM. Through this panel, the accelerator operators can load the desired filling-patterns, as well as several parameters concerning the operation of the system.

3.2.2. Software in the BC processor. The software in the BC processor provides the bunch-charge information as a channel of the EPICS standard I/O. Another of its jobs is to send the information to the timing processor through the Shared Memory System. The process for this task is an interrupt-driven one and the interrupt signal is synchronized with the trigger signal for the pulse-magnets for the injection. The signal rate is 50 Hz at maximum. It reads the bunch-charge information first and writes it into the Shared Memory, and after completion it sends a interrupt signal to the timing processor with the interrupt function of the Shared Memory System.

3.2.3. Software in the timing processor. The software running in the timing processor performs its task triggered by the interrupt issued by the BC processor. The interrupt is processed by utilizing the real-time manipulating environment of VxWorks. For example, we use the “semTake” function to wait for the interrupt.

The simplified flow-chart shown in Fig. 8 depicts the flow of the program. Since the interruption to this task is synchronous to the injection trigger, one passage of this flow corresponds to one pulse of the injector. On accepting the interrupt signal, the program wakes up, and, at first, it checks if a
Fig. 7. Operators’ panel of the filling-pattern generating system developed with MEDM. This panel has functions not only for the filling-pattern generating system but also several actions on the ring operation.

A new filling-pattern has been loaded or not. If a new filling-pattern has not been loaded, it checks if the “BCE” flag is on or off, where “BCE” stands for “bunch-charge equalizer”. The “BCE” flag is set on when the operator needs an even bunch-charge distribution throughout the ring, i.e. for the usual physics runs. In this case, the bunch-charges of all the “beam-on” buckets are compared. At this stage the comparison is done with 8-bit integer data (direct output of the ADC), i.e. we do not take the absolute value (the value calibrated by a direct-current current-transformer) into account. Then, the bucket with the least charge is found and this bucket is filled with the beam in the next pulse. Sometimes, the BCE flag is set off, for machine development such as comparing the lifetimes of bunches in the ring. In this case, the order of injection is independent of the bunch-charges.

In the “set delay” box in the flow-chart, the delay count is calculated from the target-bucket address based on expression (1) and the obtained delay-count is actually set into the TD4s. After that the program goes to the stage of waiting for the next interrupt.

4. Evolution of the system

4.1. Two-bunch injection

In the early stages of KEKB operation, the injector provides only one beam bunch per pulse with the rings. It is, naturally, preferable to accumulate the beams at a higher rate, particularly when starting...
up the operation of the rings. For this purpose, the injector was upgraded to deliver two bunches in one pulse [13]. In order to maintain the good injection condition considering the RF frequencies of the injector and rings, the two bunches in a pulse must be separated by 49 buckets of the rings. Due to this tight timing condition, filling-patterns with simple bunch-trains are not realistic. The system is informed of the desired filling-pattern with the filling-pattern-file method only. The operators are requested to prepare a filling-pattern file containing a pair of tables; the first one is a table for the desired filling-pattern itself and the second is one for the filling-pattern that would be realized if only the leading bunch were delivered.

Usually, on starting up the operation, the two-bunch injection mode is used, while, at the trickling stage, the one- and two-bunch injection modes are mixed to optimize the filling efficiency. Only in the one-bunch injection mode can the bunch-charge-equalizing function work. Two-bunch injection and single-bunch injection modes can be easily switched from one to another, so that operators are able to change the modes quickly considering the operation conditions.

4.2. Adoption of the Event System

The injector provides beams not only with the KEKB rings but also with two light-source accelerators at KEK, the Photon Factory (PF) ring and the PF advanced ring (PF-AR). Up to 2008, the operation modes of the injector (which species, which energy) were interchanged manually by operators. A new injection scheme [14] was planned so that the three rings, (KEKB’s two rings and the PF ring) can be topped up “simultaneously”, namely, the operation modes are changed pulse by pulse in a programmed order.

In accordance with this upgrade, the timing systems of the injector and the rings were replaced with new ones based on the Event System developed by Micro-Research Finland Oy [15]. In this reconstruction of the timing system, the interrupt source for the program in the timing processor was
Fig. 9. An example of the filling-patterns in the electron ring. This shows an actual bunch-charge distribution measured with the bunch-charge measuring systems. In the head part of the bunch-train, i.e. 0 through 40 in the bucket ID #, the density of the bunches is slightly higher compared with the rest of the buckets. Experience has shown that such a filling-pattern regulates the lifetimes of the bunches. At bucket ID 4842, the “pilot bunch” is stored for the tune measurement.

changed to the signal created by an event generator, which is the main module of the Event System. The event generator sends out information on the beam (which ring is filled in the next pulse) at 50 Hz. On each interrupt, the timing processor executes the required tasks based on this information.

5. Operation with the system

Each of the KEKB rings has 5120 RF buckets. About 10% of them (corresponding to $\sim 1 \mu s$) should be empty, since we must not store bunches to sweep out all the bunches completely when aborting the beam (the abort gap). This gap introduces an irregular distribution of the wake field within a beampipe. On the tail of the bulk of the bunches, a special bunch (called a “pilot bunch”) is stored for monitoring the betatron-tune, as shown in Fig. 9. The addresses of the pilot bunches for the electron and positron rings are different from each other, since watching and adjustment of the tune of a non-colliding bunch is important for maintaining the stable operation of the collider.

The number of stored bunches has been about 1500 in typical physics runs over the last few years. Though full-bucket filling was supposed at the design stage, we found that bunch-spacings of a few buckets were best for maximizing the luminosity. Actually, the filling-pattern often used was a mixture of 3-bucket spacing and 4-bucket spacing. This filling-pattern was given with the filling-pattern-file method. Figure 9 shows a typical example of the filling-pattern. At the early stages of the operation, i.e. roughly the first few years, the filling-patterns were usually made up with a set of bunch-trains, except for the abort gap.

6. Conclusion

The KEKB filling-pattern generating system was designed and constructed to operate the KEKB storage rings under a very high-current environment. Accelerator operators can inform the system of the filling-patterns that they require using an operator-friendly interface. The bunch-current measurement systems are powerful tools for maintaining a regular filling-pattern throughout the operation. In conclusion, we succeeded in manipulating the filling-patterns with the system.

Acknowledgements

The authors would like to thank Dr Takao Urano for his contribution in constructing the injector KEKB timing system.
References

[1] N. Toge (ed.), KEK Report 95-07, 1995.
[2] M. Izawa et al., Phys. Rev. Lett. 74, 5044 (1995).
[3] M. Zisman, Proc. 1987 IEEE Particle Accelerator Conf., p. 991 (1987).
[4] Y. Kimura, Proc. European Accelerator Conf., p. 23 (1990).
[5] PEP-II Group, SLAC Report 418, LBL-PUB-5379, June 1993.
[6] R. Chestnut et al., Proc. ICALEPCS 2001, p. 656 (2001).
[7] B. Kalantari et al., Proc. 2004 European Accelerator Conf., p. 2885 (2004).
[8] Information on the EPICS (Experimental Physics and Industrial Control Systems) is given in 
   http://www.aps.anl.gov/epics/.
[9] VxWorks is a real-time operating system developed by Wind River Systems,
   http://www.windriver.com/products/vxworks.
[10] SAD (Strategic Accelerator Design) is a computer program complex developed at KEK,
    http://acc-physics.kek.jp/SAD.
[11] M. Tobiyama et al., Phys. Rev. ST Accel. Beams 3, 012801 (2000).
[12] Advanet Inc., http://www.advanet.co.jp/en/index.html.
[13] S. Ohsawa et al., Proc. 2001 Particle Accelerator Conf., p. 3284 (2001).
[14] N. Iida et al., Proc. 2009 IEEE Particle Accelerator Conf., p. 2769 (2009).
[15] Micro-Research Finland Oy, http://www.mrf.fi/.