A data bank for on-line process control—The synchrotron injector

by A. DAN EELS

CERN
Geneva, Switzerland

INTRODUCTION

The need for a data bank results from the complete computerization of the CERN Proton Synchrotron injector process. This process was designed so as to allow on-line control of most of its variables and a complex software system was developed for multiple and simultaneous control. Emphasis was put on “interactive control” i.e. several operators may control in parallel part of the synchrotron injector through computer-driven consoles. Conflicting control has to be avoided and therefore a centralized information base or data bank was developed.

Broadly speaking this data bank is subdivided into two complementary parts: a very small one, which is permanently in core, and a large one (at present around 10 k), which is stored on disk.

The small core-resident section contains information which is likely to undergo frequent changes: a few “flags” called “cold start” flag (to indicate which part of the process is under no control after a cold start of the computer); “BUSY flag” to indicate which part of the process is under control from the MAXI-CONSOLE; “MANUAL flag” to indicate which part of the process is under control from its dedicated MIDI-CONSOLE; the starting address of a binary matrix containing all “status” information, such as “ready” or “not ready,” “on” or “off” and “polarity” status of the process variables. This matrix is built up of a number of 16-bit words (so far 25 words) in which every single bit represents a status (ready/not ready, or on/off, or polarity). Random arrangement of status bits within the matrix eases the hardware aspect of collecting this status information from the various variables. This matrix is permanently in core as part of a surveillance program, which is imbedded within the MIDI-CONSOLE program and which scans the various status bits at regular time intervals.

Nothing more needs to be said about this core resident section of the data bank. This paper gives a detailed discussion of the characteristics of the disk-based section of the data bank.

GENERAL

The control of a process is in general implemented by a great many control variables of different types.

To vary any control variable one needs to have some information at hand, as to its current control value, whether this variable is controlled via, for example, a stepping motor, what is its possible maximum value, whether it is switched on or off, etc. It will be shown here how, in general, one needs around twenty such pieces of information which are called the characteristics of every control variable.

“Control variable” has been mentioned so far without indicating whether they are single control elements, or several control elements combined into one control vector. Throughout this paper distinction will be made (where needed) between control elements and control vectors, where the latter are defined as a linear combination:

\[ X = \sum_{i=1}^{n} \alpha_i x_i. \]  

where

- \( X \) is the control value of a vector,
- \( x_i \) is the control value of the ith element,
- \( \alpha_i \) is a weighting factor,
- \( n \) is the number of elements.

Any statement about control variables is valid for either elements or vectors.

Any program controlling all or part of the process (e.g., on-line optimization) needs to know the characteristics of its control variables, which therefore should be stored, for example, on magnetic disk.

A process such as the synchrotron injector has around 500 variables each characterized by 20 pieces of information, hence one will need space for around 10,000 pieces of information. This plunges us into the problem of data banks, data structure, information retrieval and updating.

It will be shown how a data bank for control variables may be given a simple structure.
It will also be shown how to define every control variable by a single code, SOFT-CODE, for retrieving its characteristics.

Eventually a software system will be described, which allows easy retrieving and updating some of the characteristics without the need for direct access to the data bank.

CHARACTERISTICS OF A CONTROL ELEMENT

A control variable has been defined (see the previous paragraph) as either a control element or a control vector. This section will describe the characteristics of every single control element. The control vectors will be dealt with in a subsequent paragraph.

Control element’s name

This is for the purpose of displaying the name of the control element on alphanumeric displays as, for example, on the MIDI-CONSOLE. Such a name is in general a mnemonic indicating to which group the control element belongs, for example, a vertical steering magnet in the injection line of the synchrotron injector, I-DV, followed by an item number, e.g., the 7th one. This full name is thus I-DV07.

Consequently a full name will be a mnemonic of four alphanumeric characters indicating the group followed by a 2-digit item number. This arrangement covers a large number of possible “names.”

Physical units

Once again for displaying the measured value of a control variable one prefers this value to be indicated in physical units: for example, a current, even though measured as a voltage over a shunt, should be displayed in A or mA; an angular position, even though measured, for instance, as a voltage, should be indicated in mrad, etc. In general two characters will be sufficient for defining the physical units.

A BUSY indicator

Any control variable may be controlled from various places. This is particularly true in a multiprogramming environment. To avoid conflicts one needs a centralized indication to tell whether a control variable is already under computer control or not. This indication is available in a “flag” as part of the core-resident section of the data bank (cf. Introduction).

Value of the least significant bit (LSB) at acquisition

Suppose one measures a current. In general this is done through a shunt resistor with a fixed value so that one actually measures a voltage which depends “linearly” (assume the shunt is linear) on the value of the current. The ratio voltage/current depends on the value of the shunt resistor. The analogue value of the voltage is now converted into a digital value, so that going back through all the various conversions one finds a definition of the LSB (i.e., digital value = 1) as a function of the physical unit, e.g.,

\[ \text{LSB} = 44.6 \text{ mA}. \]

Position of the decimal point

In previous examples, the smallest value one could measure corresponded to the value of the LSB (digital value = 1) at acquisition. Therefore, the least significant digit in physical units is 10 mA or 0.01 A, and consequently if the measured value were to be displayed as a 5-digit number in A one would have to insert a decimal point in front of the second digit right justified: e.g.,

170.15 A.

Value of the LSB at control

Having observed the current value of the control variable, for example, on the display, one decides now to increase or to decrease this value. Therefore, one introduces a positive or negative increment defined in its physical units. This increment therefore has to be processed by the computer and converted into the appropriate digital value before sending it to the digital control register of the control variable: this implies that one should define the physical value of the LSB at control (digital control value = 1). If, for example, the maximum value of a power supply is 15 A and it is controlled by a 10-bit digital register then, of course, the LSB is 14.6 mA.

Maximum and minimum value

This has been mentioned implicitly in the previous section. One needs to be safeguarded against exceeding limits which may not be symmetrical (e.g., unipolar supplies).

Type of control element

One distinguishes, in general, between three types of control elements.

(a) A-type

This is defined as the control element whose control value results from the algebraic sum of its previous control value and an increment, for example

\[ x_t = x_{t-1} + \Delta_t, \tag{2} \]

where \( t \) is the time of control.
(b) \textit{S-type}

This is defined as the control element whose control value results from an increment only (e.g., Stepping motors)

\[ x_t = \Delta, \quad (x_{t-1} \text{ always} = 0). \tag{3} \]

(c) \textit{C-type}

This is in fact an A-type control variable for which there is no acquisition (measurement) foreseen. This applies more specifically to the preset counters of the synchrotron injector: in particular their control value includes the selection of a proper timing clock.

\textit{Indication for READY-NOT READY}

Some control variables may be READY or NOT READY depending, for example on whether there is a power supply connected to it or not. This indication is supplied to the computer for most elements of the synchrotron injector.

\textit{Indication \textit{ON-OFF}}

This indication allows one to find out whether an element has been switched on or off. If it is found to be switched off the computer should, of course, switch it on before control can start.

\textit{Indication \textit{POLARITY}}

For some control elements the polarity of the measurement may be supplied in a different piece of information.

\textit{RESET}

When an element is found to be NOT READY the computer may attempt to RESET this element.

\textit{Acquisition and control addresses}

The process computer is connected to the process by a transmission system* through which measurements are acquired and control values sent out. In order to “multiplex” the transmission to or from the proper control elements, every element is given two addresses, one for acquisition, one for control.

\textit{Current control value}

In order to avoid non-linearities between control value and measured value and also to avoid errors in the acquisition (e.g., missing bits) one may decide not to compute the new control value as the sum of the previous measured value and an increment but as the sum of the previous control value and an increment. This implies that the computer has memorized what value is actually on the digital control register of any control element.

\textit{“Reference" value}

This is a control value which was found to be satisfactory in the past as it ensured a good working condition of the process. If the current control value proves to harm the working conditions of the process one wants to return to the “reference” values.

\textit{“Buffer”}

This allows one to compare the working conditions of the process resulting from the “current control value,” with the conditions resulting from the “reference value,” without destroying the “current control value” which is stored in the “buffer.”

So far the characteristics of a control element have been enumerated. However, in general, some of these characteristics will be common to various control elements. As an example, all vertical steering magnets of the injection line of the synchrotron injector upstream to the distributor are called I-DV, their value is displayed in \( A \), and they are of the A-type, etc.

However, the 7th member of this group will be called I-DV07, and has an individual address for acquisition and control, etc.

So in general one can state that the first eight characteristics will be common to a group of control elements, whereas characteristics nine through sixteen will be specific to each element of this group. Hence we come to a two-level data structure with group characteristics and individual characteristics.

\[
\begin{aligned}
\text{GROUP(J)} & \quad \text{INDIV(J,1)} \quad \text{INDIV(J,2)} \ldots \text{INDIV(J,M)}
\end{aligned}
\]

\textbf{CHARACTERISTICS OF CONTROL VECTORS}

The control vectors have been defined previously (see the Introduction). A control vector is characterized by:

(i) \textit{the number of its control elements:} e.g., \( N \),
(ii) \( N \) codes: each defining a control element,
(iii) \( N \) coefficients (which need not necessarily be integers):

\textit{to weight the control value of every one of its control elements.}

\textbf{DATA BANK STRUCTURE ON DISK}

\textit{Control elements}

The simplest method is to store contiguously the record representing the characteristics of group \( J \), then the char-

* The synchrotron injector uses the digital STAR system (Système de Transmission Adresse Rapide) developed by the Control Group of the CERN Proton Synchrotron Division.
characteristics of all its M individuals, then the specification of group J + 1, etc. However the number of individuals M may be different from one group to another and retrieving the characteristics of one single control variable would require, in general, knowledge of two pieces of information: the address (on disk) of its group characteristics and its item number within this group.

This arrangement is also rigid and would not allow, for example, giving different group characteristics to the same string of control variables (e.g., give it another name when various control elements of one group are combined into a control vector).

Consequently, one is led to a more flexible arrangement by storing into two different files the group characteristics and the characteristics of their individuals

\[
\ldots \text{GROUP}(J), \quad \text{GROUP}(J+1), \ldots \\
\text{INDIV}(J,3), \ldots \text{INDIV}(J,M); \quad \text{INDIV}(J+1,1), \ldots
\]

Each record of group characteristics includes a pointer to the address on disk of the individual characteristics of the first element of this group.

By arranging the record of group characteristics to be of fixed length (10 words) one is able to code both the address on disk of the group record of a control element and its item number into one piece of information. An example of such a coding will be given in a subsequent section and will be called SOFT-CODE (Software Code).

**Record length for group characteristics**

Characteristics one to eight inclusive have been defined as group characteristics. The "name" has been defined as four alphanumeric characters indicating the group followed by an item number within this group. This item number may be introduced in the code, called SOFT-CODE, which will allow retrieving the characteristics of the control variable. Therefore, it need not be stored on disk. Consequently, one will need in general two words of storage for the name of the group (alphanumeric characters are coded two characters per 16-bit word).

Furthermore, the record of group characteristics will include a pointer to the record of its first element (i.e., item 0).

By appropriate coding of other information such as decimal point position, busy indicator, and value of LSB, one can keep the length of the group record down to ten words (cf. Figure 1).

**Record length for individual characteristics**

Characteristics nine to sixteen, inclusive, have been defined as individual specifications. The total length of this record is effectively of 10 words, as is shown in Figure 1.

**Control vectors**

Control vector characteristics are imbedded within group characteristics of control elements. One distinguishes a vector from an element by its SOFT-CODE (see following paragraph).

To preserve the 10-word modular structure a vector of N elements is subdivided into subsets of four elements. The first word contains the number of elements; then four codes, SOFT-CODES, define the elements followed by their corresponding weighting factors \( a \). The last word points to the next following subset and is equal to \(-1\) (hexadecimal/FFFF) in the last subset. However, so far there are no such examples among the synchrotron injector control vectors.

**Estimate of the number of sectors on disk**

In the particular example of the synchrotron injector, an IBM 1800 process controller is used, with moving search head magnetic disks for mass storage. Every disk has 512 K of 16-bit words storage capacity, subdivided into 200 cylinders.

---

From the collection of the Computer History Museum (www.computerhistory.org)
of eight sectors each. Every sector has a capacity of 320 words. Hence 32 group characteristic records may be stored per sector. The number of groups of control variables for the synchrotron injector is estimated around 256, which corresponds to eight sectors. Another 24 sectors store the individual characteristics.

SOFT-CODE

Every control variable is defined by a SOFT-CODE retrieving the record of its characteristics in the data bank. The IBM 1800 process controller which is used by the synchrotron injector is a 16-bit machine; consequently the SOFT-CODE is a 16-bit word which is organized as follows:

- **(i) Most significant bit, sign bit (MSB)**
  - This bit is set to 1 for control vectors and is set to 0 for control elements.
- **(ii) Group number (G)**
  - In the case of the synchrotron injector 256 groups have been estimated; hence eight bits.
- **(iii) Item number (I)**
  - The remaining seven bits indicate the item number of a control element within its group.

This SOFT-CODE is effectively a pointer to the address on disk where the characteristics of its corresponding control variable are recorded. This address on disk is computed as follows:

\[
S_s + Q \left( \frac{G \cdot 10}{N_s} \right) + R \left( \frac{G \cdot 10}{N_s} \right),
\]

where

- \( S_s \) is starting address on disk of the data bank,
- \( G \) is the group number,
- \( Q \) is the quotient of the division between braces,
- \( R \) is the remainder,
- \( N_s \) is the number of words per sector (= 320 in the case of the IBM 1800);

- address of the record of individual characteristics:

\[
S_s + P + (I \cdot 10),
\]

where

- \( P \) is the pointer to the first item of the group (first word in the group characteristics),
- \( I \) is the item number.

DATA BANK HANDLING SOFTWARE SYSTEM

Every process control program has to refer to the data bank. However, one should prevent part or all of the data bank from being destroyed, for example by software bugs. In particular, the disks on which the synchrotron injector data bank is stored have a moving search head and the average access time is around 200 msec; moving the head can almost not be avoided because of the data bank structure (four cylinders or tracks; the first one contains group specifications, the remainder contains individual specifications). Therefore, some retrieving and updating operations need to be optimized to be short in time. As a general rule a programmer should not have direct access to the data bank, but only through an appropriate software system.

The various subroutines are written in machine language with FORTRAN compatible calling sequences. The error
indicator is always zero after successful termination of the subroutine.

All those subroutines are modular for simple program structure as is shown in Figure 2.

CONCLUSION

Implementing the data bank has proved very useful, especially during the commissioning of the various process components. For example, the calibration of the components required frequent modification of the definition of the least significant bit, both for control and for acquisition. This could be done in a flexible way through an interactive program displaying the current control variable specification on an alphanumerical display and returning the modification to the data bank.

All control programs have immediately the latest information in hand and need not to be reassembled and recompiled.

Furthermore only a rather small amount of information needs to be loaded in core for controlling part of the process.

However, the two-level structure is a limitation, particularly when vectors using similar elements are given different names for display purposes and also when various process variables only differ because of their name whereas all other group specifications are identical. In this case a three-level structure is more flexible but requires faster disk access.

ACKNOWLEDGMENTS

The author gratefully acknowledges the discussions he has had with H. Van der Beken, L. Burnod, and W. Remmer, which have benefited this work.

REFERENCES