IEEE Transactions on Nuclear Science, Vol.NS-24, No.3, June 1977

FERMILAB MAIN RING POWER SUPPLY CONTROL PROGRAMS

H. Pfeffer and D. Jong
Fermi National Accelerator Laboratory*
Batavia, IL 60510

Summary

The Fermilab Main Ring Power Supply System consists of 60 3-MW SCR power supplies controlled and regulated by two linked minicomputers.[1] This paper describes the computer operation - the algorithms, programming techniques and timing.

General Considerations

The Main Ring power supplies are controlled by computer rather than hardware because of the complexity of the waveforms necessary to operate the system. The Magnet Current program itself includes multiple flattops, variable slopes and parabolic curves within its typical 12-second cycle. Also, since the SCR supplies have insufficient gain-bandwidth to regulate completely in real time, a complicated, self-correcting voltage waveform must be learned over many machine cycles.

The computers are interrupt driven at a 720-Hz rate; this is an optimum speed since it is the fastest response time of the 12-phase SCR supplies. The 720 Hz is line-locked, so that analog data is sampled at a fixed phase with respect to line-related ripple. The essential regulating calculations are done at high priority levels, and are completed within every 720-Hz cycle. Remaining time is devoted to less important functions such as plotting and operator interfacing.

Two computers are used (Lockheed MAC 16 16-bit minicomputers) because one CPU is not fast enough to do all of the calculations required in a 720-Hz interval. Both CPU's are presently using 95% of their available time, which makes improvements difficult. We will soon be purchasing a 3-4 times faster mini to replace both MAC's, and to increase our timing margin.

The Regulation System

Figure 1 describes the regulation system. The processes described in it are repeated for each of the 3 magnet busses (bend plus 2 quads).

The disc is the starting point of activity; each 720-Hz interval it supplies a program, disc profile and update. The program is simply the desired magnet current for that time slot. The disc profile represents the total voltage that the power supply system should be generating, and the update is a learned correction to the disc profile. The disc profile and update are summed to form the working profile. This number is then converted into power supply phasing and transmitted to the appropriate power supplies.

Feedback is then applied. The measured magnet

*Operated by Universities Research Association, Inc., Under Contract With the U. S. Energy Research and Development Administration.

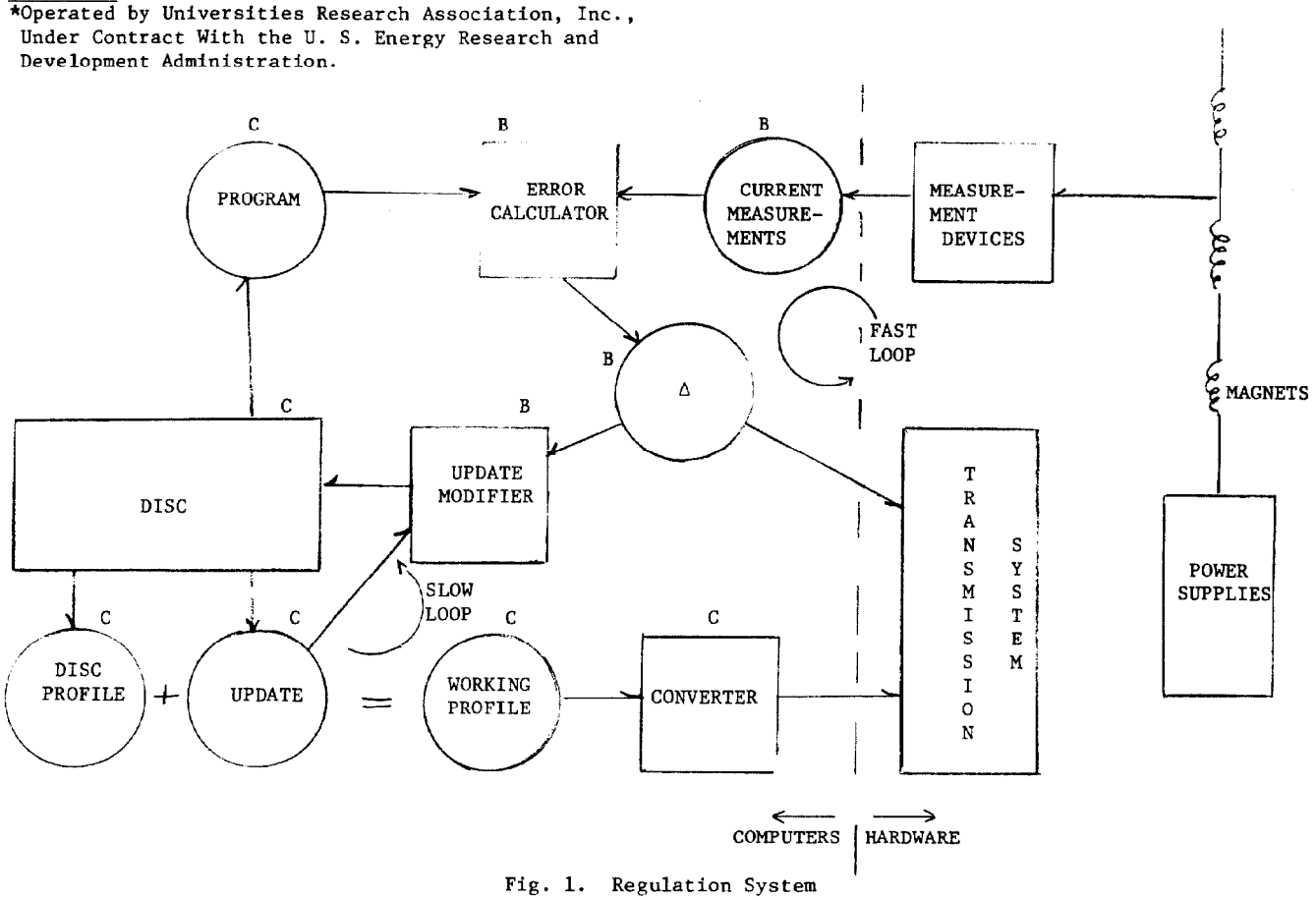

Fig. 1. Regulation System

current is compared to the disc-generated program, and an error signal (Δ) is produced. Δ is used in two ways: it is transmitted to a power supply to provide fast feedback correction, and it also modifies the update which is then stored back onto the disc for use in the next ramp cycle. The updates change slowly, from cycle to cycle, and constitute a self-correcting function generator.

The two minicomputers are called MAC-B and MAC-C. Labels in Fig. 1 indicate the sharing of functions between them.

Elements of the System

A) The Disc

A 1-megabyte disc is attached to MAC-C. It can store the data for six 12-second ramp files, although in normal operation we run continuously on one file. Data is transferred from the disc to 320-word length buffers in MAC-C core via the Multiplexed Data Channel (MDC). The transfer is initiated by the CPU, then controlled automatically by the MDC while the CPU resumes normal regulation calculations. Two buffers, A and B, are loaded alternately (see Fig. 2). The CPU regulates on the data in buffer A for 444 ms (320 points x 1/720 sec) while buffer B is being loaded. Then the CPU operates on buffer B while buffer A's modified updates are sent back to the disc and new data is loaded into the buffer. The process then repeats throughout the cycle. A third buffer - the R buffer - resides in core and is used immediately after a reset pulse starts the new ramp cycle. The R buffer contains the data for the first 444 ms of the ramp. It must be kept in core because the reset pulse is not synchronous with the buffer transfer timing.

Each 320-word length buffer contains nine parallel segments - (program+profile+update)x(bend+2 quads) - so is actually 2,880 words in size.

B) The Converter

To convert the working profile into power supply control information, we divide the profile by the nominal supply voltage (900 V), and get a quotient (Q) and remainder (Rem). Q indexes a power supply "turn-on order" table to find the address of the indicated supply; Rem indexes a 450-word "phase" table to find the appropriate phase angle. The data is then transmitted to the power supply.

The profile increment is limited to 200 volts-per-720 Hz interval, so that each power supply in the turn-on order must be addressed in sequence, and none can be skipped over. When the profile increases to address the next supply in the turn-on order, the previously addressed supply is turned full-on.

C) Update Modification

If the error signal (Δ) indicates excess current in a given time slot, the corresponding update is decreased, so that during the next ramp cycle there will be less voltage and hence less current. The formula for this correction is:

$$Up^{N+1}(t) = Up^{N}(t) + \Delta \cdot UpG$$

- Superscript refers to the number of ramp cycles
- (t) refers to the 720 Hz count within a cycle.

UpG = update gain

Up = update.

This corrects in the right direction, but is not a stable system. Because the magnet bus is inductive, the current at a given time depends upon the voltage in previous time slots as well as the time slot in which the correction is made. Instability shows up as an alternate time slot blow-up of the updates, in which the odd-slot updates grow in the positive direction and the even ones grow negatively. Each update tries to regulate its own time slot, but adjacent updates fight each other.

Damping is added to the calculation in the form of "neighbor gain". Each update is corrected slightly towards the line joining its two neighboring updates (see Fig. 3).

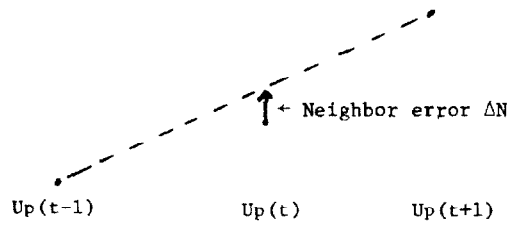

Fig. 3. Update neighbor correction.

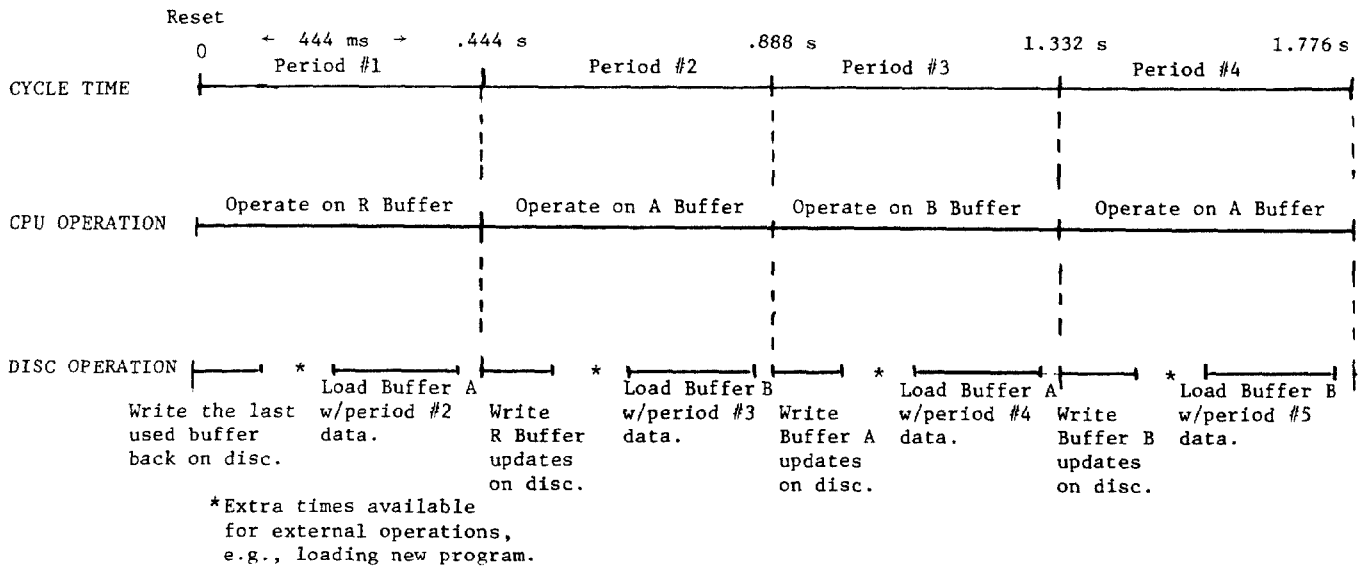

Fig. 2. Disc operation.

Up(t) is corrected by adding a fraction of ΔN. The total update modification can now be written as:

$$Up^{N+1}(t) = Up^{N}(t) + \Delta \cdot UpG + ([Up^{N}(t+1)-Up^{N}(t)]-[Up^{N}(t)-Up^{N}(t-1)]\cdot UNG$$

(UNG = Update Neighbor Gain)

Damping the system reduces the bandwidth, as can be seen by noting the resemblance of the neighbor term to the 2nd derivitive of the update function (Δt's implicit), and rewriting the formula as:

$$Up^{N+1}(t) = Up^{N}(t) + \Delta \cdot UpG + Up^{N}{}''(t) \cdot UNG$$

When the learning process reaches equilibrium, $Up^{N+1} = Up^{N}$. Thus,

$$\Delta \cdot UpG = -U^{N}{}''(t) \cdot UNG$$

and

$$\Delta = -U'' \cdot \frac{UNG}{UpG}$$

The residual error is proportional to the 2nd derivitive of the update function; the update can't make high frequency corrections. For best operation, one wants the highest UpG and lowest UNG consistent with stability. We have found these empirically in our system.

The update bandwidth is extended by "making a profile", which means storing the working profile back as the disc profile and clearing the updates. The updates can then start learning again. This process is typically done 3-5 times while learning a new ramp waveform. It cannot be repeated indefinitely without encountering stability problems. The modified disc profile is a valuable "fallback file"; if anything disturbs subsequent updating, the updates can be cleared to return to a reasonably accurate profile.

D) Fast Regulation Loop

The error signal is added to an "offset" level which biases the regulation supply at 85% of its full voltages. The error signal causes small variations (±100 volts) around this level.

CPU Timing

MAC's B and C run synchronously, and pass data back and forth every 720-Hz cycle via a parallel transmitter with a first in, first out stack. Figure 4 shows the timing of the computers.

The emphasis in the programming of both computers has been on minimizing execution times. The programs are written in LEAP Assembler language. Repeated routines are written out rather than looped, since indexing takes an additional 1 μs per instruction. Data tables are kept on the same 512-word page as the calculations referencing them, to avoid indirect or base page addressing. Mathematical routines, especially double precision ones, are constructed in specific rather than universal formats, to conform to knowledge of the numerical inputs. For example, if both inputs to a particular multiplication are always positive numbers, sign-checking routines are skipped.

These techniques and the great care with which the code is written, allow us to complete the numerous operations required to regulate the power supply system.

Acknowledgement

The programs were written under the guidance, and following the ideas of, Dick Cassel - presently at Princeton's Tokamak.

Reference

[1]R. Cassel; "A Multiple Flattop Control System for the Fermilab Main Accelerator Power Supplies," IEEE Trans. on Nucl. Sci., NS-22, p. 1247 (1975).

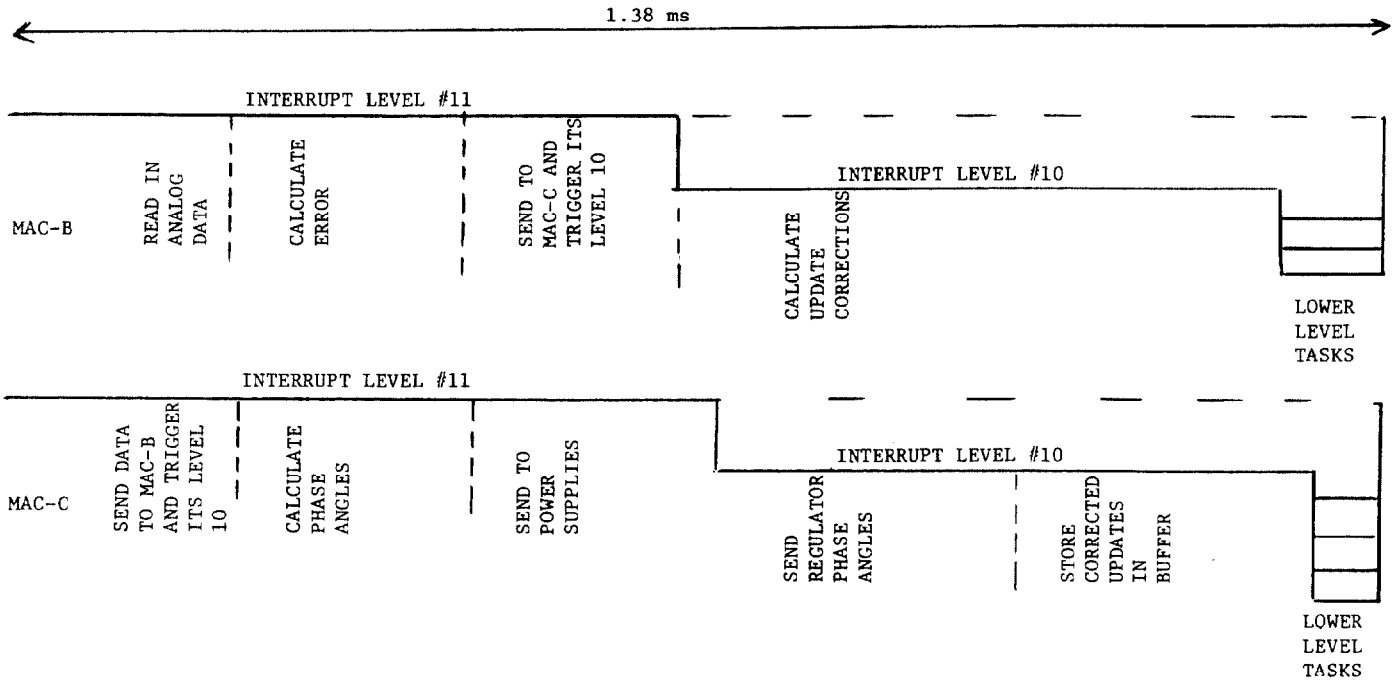

FIG. 4. CPU Timing.