RHIC/AP/91
April 11, 1996

# Sextant Test Ramps

D.P. Deng, J. Kewisch, B. Martin, J. Rose, S. Peggs

## 1 Introduction

This note has two goals. The first is to update the discussion of RHIC ramps in a previous RHIC/AP note [1]. For example, this paper invokes only 2 types of WFG, not 3 as before. The second goal is to record the consensus of the "Ramps Working Group" that met in early 1996, and which reported back to the "Controls Task Force".

For the sake of brevity, it is assumed that reader understands something of how the Real Time Data Link (RTDL), Wave Form Generator (WFG) channels, and Input Modules (IM) work. They are described in various references, should the reader be unfamiliar with them [1, 2, 3, 4, 5, 6].

## 2 What is a ramp?

A ramp is the synchronized change of the machine set levels to bring the machine from one state to another. For RHIC we identified the following independent ramps:

- The Reset ramp includes everything neccessary to make the machine ready to accept beam, i.e. cycling of magnets to compensate hysteresis effects, and ramp to the injection settings.
- The Injection ramp includes the necessary steps to fill RHIC.
- The Acceleration ramp includes the ramp of the beam energy and the transition jump.
- The Re-bucketing ramp. Other RF specific ramps will be added later.
- The Beta squeeze ramp.
- The Ring synchronization ramp.

| Increment name | Symbol | Units |
|---|---|---|
| <u>1996</u> | | |
| QD strength | $\Delta KLF$ | $\mathrm{m}^{-1}$ |
| QF strength | $\Delta KLD$ | $\mathrm{m}^{-1}$ |
| <u>1998</u> | | |
| Arc quad F | $\Delta KLF$ | $\mathrm{m}^{-1}$ |
| Arc quad D | $\Delta KLD$ | $\mathrm{m}^{-1}$ |
| Skew quad family 1 | $\Delta SQ1$ | $\mathrm{m}^{-1}$ |
| Skew quad family 2 | $\Delta SQ2$ | $\mathrm{m}^{-1}$ |
| Skew quad family 3 | $\Delta SQ3$ | $\mathrm{m}^{-1}$ |
| Sextupole F | $\Delta SF$ | $\mathrm{m}^{-2}$ |
| Sextupole D | $\Delta SD$ | $\mathrm{m}^{-2}$ |

Table 1: A list of knob increments that will be put on to RTDL by Input Modules. Knob variables are (usually) manually controlled. They are always active - either on top of a ramp as it plays or (most likely) in the absence of any ramps. The "units" are symbolic - actual RTDL variables will be 24 bit integers.

These ramps will be executed in sequence. However, the design will not inhibit the execution of the energy, beta squeeze and re-bucketing ramp in parallel after upgrades to the WFG firmware.

The basic tools used to execute ramps are WFGs. Many pieces of accelerator equipment, such as magnet power supplies and RF cavity phases, have their set levels controlled by WFGs. These set levels are changed in concert when the WFGs "play their tables".

For some of the RF set levels the control by WFGs is too slow. In these cases WFGs might be used to for the coarse set level and dedicated electronics for the fine tuning. Examples are the voltages of the cavities.

The current implementation of the WFG firmware has 64 sets of tables. A set of tables is selected by connecting the set to an event on the event line. The next occurrence of the event will start the play. In order to minimize the fill time it is a requirement that all ramps shall be downloaded into the WFGs before the filling (Jörg's Law). The "Super Ramp" can then be executed by a high level sequencer by sending the appropriate events on the event line.

To aid the optimization of ramps for minimum beam loss and emittance growth the operator must be able to slow down a ramp (or completely halt the ramp) at critical points so that measurements or corrections can be performed.

Another requirement is the possibility of "knobbing", i.e. the change of important system parameters ideally with a total response time of much less than 1 second. Knobbing allows the implementation of feedback loops, either directly or through the input of an operator. Table 1 lists those variables which may be incrementally knobbed in 1996, and in 1998.

# 3 System architecture

In order to synchronize ramps the WFGs are linked together with two different data busses: The event line and the RTDL line. In the present implementation the output value of a WFG is written as:

$$W = S_i \cdot V_i \cdot F_i(t) + S_j \cdot V_j \cdot G_j(V_g) + S_k \cdot V_k \cdot H_k(V_h) \tag{1}$$

where $S_i$, $S_j$, and $S_k$ are scale factors, $V_i$ et cetera are RTDL variables or "frames" delivered at 720 Hz, and time $t$ is locked to the 720 Hz event on the time-line. $F()$, $G()$ and $H()$ are linear interpolations of the WFG tables.

RTDL variables are generated through an input module from one of the following sources: software running in the FEC, hardware (e.g. a DCCT on the dipole bus) or from the output of another WFG.

In RHIC two levels of WFGs will be used: A single first level WFG (one for most types of ramps) will produce a "pseudo-time" value which is distributed as a RTDL variable. The second level WFGs (one for each device) translate this pseudo-time into the device setting. Figure 1 illustrates this concept. Table 2 lists the pseudo-times for each ramp.

This two-level approach will serve two purposes: The speed of the ramp can be varied by programming just the pseudo-time WFG and the implementation of "meaningful" knob increments becomes easier.

If not for the knobbing all set levels could be calculated as a function of the pseudo-time in the console level computer (CLC) before the start of the ramp and downloaded as a single table. No arithmetic would be necessary on the WFG level. If the knobbing increments were in the same units as the WFG output the WFG arithmetic would contain just an addition of the table value and the increment.

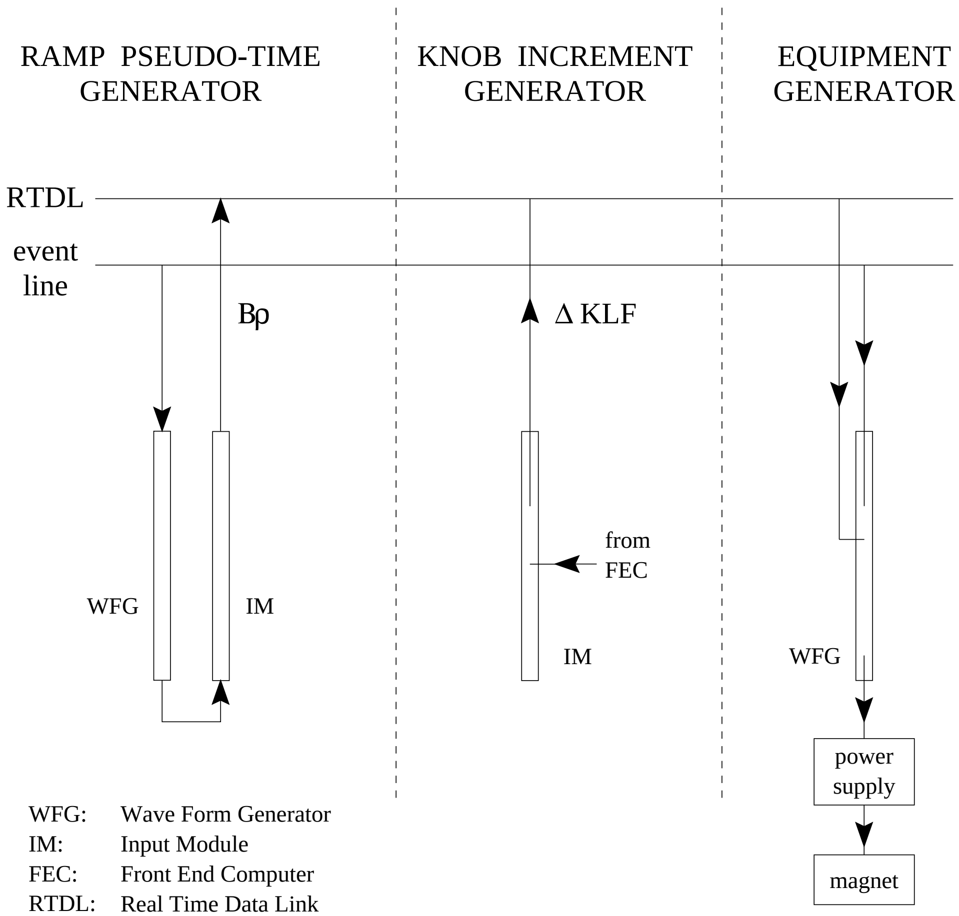


Figure 1: Conceptual relationship between RAMP PSEUDO-TIME wave form generators, KNOB INCREMENT generators, and EQUIPMENT wave form generators. Both ramp WFGs and knob IMs write parameter values onto the RTDL, perhaps (in an usual case) simultaneously. Only equipment WFGs read from the RTDL. In the example shown, the acceleration ramp is playing, and posting the rigidity pseudo-time, $B\rho$, onto the RTDL. At the same time, a quadrupole correction increment, $\Delta KLF$, is being manually added. The equipment WFG attached to a main arc quadrupole power supply reads both of these variables, and processes them appropriately.

| Ramp name | Pseudo-time | Symbol |
|---|---|---|
| 1996 | | |
| Reset | N/A | |
| Accelerate | rigidity | $B\rho$ |
| Re-bucket | time | $t$ |
| 1998 additions | | |
| Injection | | |
| Low beta squeeze | representative $\beta^*$ | $b^*$ |
| and more ... | | |

Table 2: A list of RHIC ramps, and associated pseudo-time parameters (if any). A single pseudo-time is put on to RTDL by a ramp generator WFG. It may be sped up, slowed down, or even halted, relative to true time.

For an efficient tuning of the machine the operator should not have to knob the low level set point of the device but a higher level quantity such as the tune or chromaticity. The set-points of the quadrupoles and sextupoles are then calculated from the knob values. Because the WFGs execute the ramp independent of the CLC only a part of this calculation can be done on the CLC, the rest must be performed by the WFGs.

As an example we consider the QD quadrupoles during the beta squeeze ramp. The quantity the operator wants to change is the horizontal tune $\Delta\nu_H$ of the machine. The magnet current is calculated from $\Delta\nu_H$ in four steps:

First, the change in quadrupole strength $\Delta k$ is calculated by multiplying with a scale factor derived from the integrated beta function in the QD quadrupoles. $\Delta\nu_H$ is calculated on the CLC and is send as a RTDL variable to the WFG (see Figure 1).

Next, $\Delta k$ is received by the WFG and added to the strength $k$ which was downloaded as a function of the pseudo-time $\beta^*$ before the start of the ramp.

Next, the field gradient of the quadrupole is calculated by multiplying the quadrupole strength with the rigidity of the beam. The rigidity is available as the pseudo-time of the acceleration ramp.

Last, the magnet current is calculated from the field gradient. As long as the saturation of the magnet can be neglected this calculation involves only a constant multiplier. The currently implemented formula for the WFG output is capable of performing these calculations in real time.

If the saturation of the magnet can not be neglected the second step uses the transfer function of the magnet. This function is available as a table from the "magbase" data base and could be downloaded into one of the WFG tables. In order to use the WFG in this way the formula for the WFG output must be modified to allow indexing of a WFG table by the result of an intermediate calculation rather than a RTDL variable. This modification is planned for the sextant test.

An additional complication is introduced by the wiring of the interaction region quadrupoles. In order to minimize the heat loss through the cable penetration in the heat shielding the quadrupoles are wired in a matrix where power supplies affect more than one magnet and magnets are powered by more than one power supply. Figure 2 of an interaction region shows power supplies as boxes, penetrations as triangles and quadrupoles as ovals. $ID$ and $IF$ indicate the global bus for the focusing and defocusing arc quadrupoles.

In order to maintain the above described way of calculation each WFG will be associated with a magnet and a power supply at the same time. The WFG calculates first the magnet current and then the power supply current. To do so, the WFG needs the magnet currents of the other magnets driven by the power supply. For example, the current of power supply P1 is calculated by:

$$I_{P1} = I_{Q1} - I_{Q2} - I_{Q3} - I_{Q4} - I_{Q7} - I_F \tag{2}$$

Different ways of implementing this cross-talk between WFGs have been discussed. Since all WFGs of one interaction region are in the same VME crate a simple and inexpensive way would be a task running on the front end computer CPU at a rate of 720 Hz which copies the magnet currents between the "Score board" memory of the WFGs. The $I_F$ and $I_D$ currents would be distributed as RTDL frames.

Independent of the cross-talk method this calculation requires intensive changes of the WFG firmware which can not be completed before the sextant test.

# 4 WFG software implications

WFG software includes the WFG firmware that is downloaded from the flash PROM on the V115 controller card and executed by the Intel 80960-CA microprocessor, as well as the wfgRamp ADO methods executed by the FEC, and

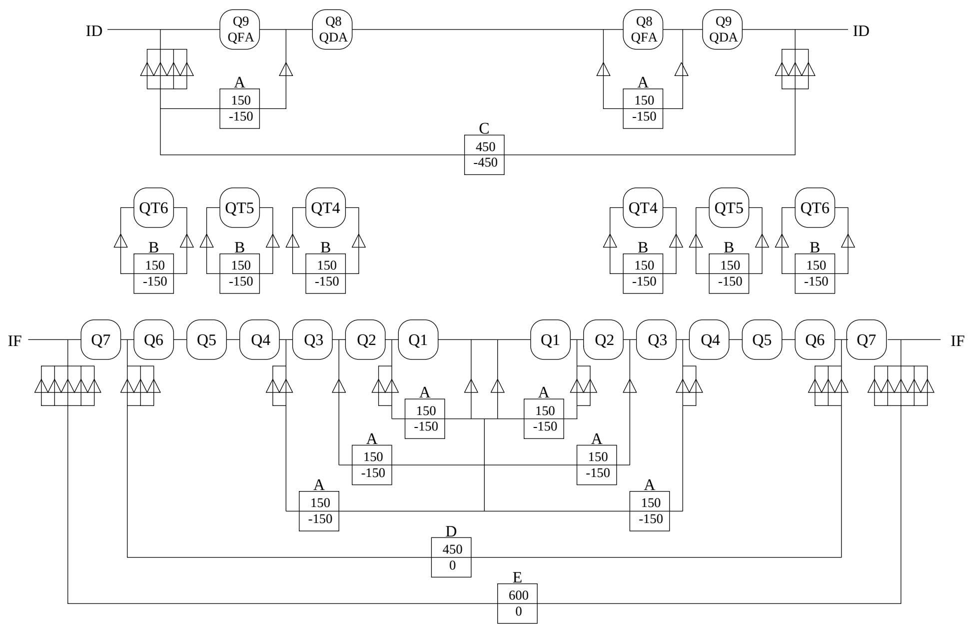


Figure 2: Power supply configuration for IR quadrupoles (copied from the RHIC Design Manual).

| ISSUE | 1996 | future options |
|---|---|---|
| Type of arithmetic | 32 bit integer | scaling/floating point in software |
| Shifting subexpressions | new feature | - |
| Formula type | as is | specifiable |
| Operation order | fixed | specifiable |
| Index tables by intermediate variables | new feature | - |
| Interpolation $(F, G, H)$ | linear | quadratic, cubic |
| Table spill-over | use new table | larger or variable table length |
| Behavior after event | new table | continue in table |
| Mag. transfer function | in WFG | - |
| Engineering units | in ADO | - |
| Internal clock rates | 720, 1000 Hz | 5, 10 kHz |
| External clock rates | 720 Hz | 5, 10 kHz |

Table 3: Software issues and options. Choices for 1996 will be implemented for the sextant test. Other options will be considered for circulating beam.

some of the applications and utility programs run on CLCs. While the focus is generally on the firmware itself, some of the issues are best dealt with by defining the computational and real-time requirements and treating the distribution of functionality as merely an issue of software architecture.

The existing WFG software (firmware and ADO) provided AtR only with DC ramps to set points determined by magnet applications programs, but it is capable of downloading any arbitrary tables for F, G, and H. Modifications now in progress expand the number of tables (by a factor of four), introduce shift-factors to preserve precision, and allow certain other values to be used as multipliers or independent variables.

Each of the items identified below as a "1996 design choice" will be incorporated in the WFG software in time for the Sextant Test, and performance testing will be done to determine maximum clock rate. Table 3 illustrates the major decisions pertaining to the design of software for controlling the power supplies.

### 4.0.1 Type of arithmetic

The WFG processor uses 32-bit integer arithmetic, internally, to produce the 24-bit output value. Shift counts are being added to WFG formula, for proper scaling of RTDL values and to preserve more precision over the dynamic range of the calculations. We see no problem with this for the sextant test.

While floating point software is possible, it would greatly damage the real-time performance of the WFG. Alternatively, function editor programs, running on processors with floating point, may be able to produce sufficient granularity in the downloaded tables to achieve the desired ramp characteristics. This needs to be determined for 1998.

### 4.0.2 Operation order and formula type

As described in Chapter 3 and 4 a more flexible formula for the output of the WFGs is necessary for efficient ramps. A improved but fixed formula will be available for the sextant test. Future options include a fully programmable formula.

### 4.0.3 Interpolation

Linear interpolation between rows of a table, as now performed by the WFG, will be adequate for the 1996 Sextant Test. A discussion weather it will be necessary for circulating beam operation to use quadratic or cubic interpolation within the WFG itself is continuing. It may be possible to show that the function editor software is capable of building tables which can be downloaded to achieve the desired ramp characteristics, without going beyond linear interpolation.

### 4.0.4 Spill-over from $F$ table

In the sextant test, it may be necessary for a ramp to "spill over" from one table to another (that is, to require more than 128 rows). This can be done with an additional event or with a simple firmware change (to continue when a table is exhausted). In the long run, it is desirable to keep to one ramp description. Since the WFG software was recently changed such that there is now a pointer to each table (due to PPM enhancements and compiler limitations), it is rather straightforward to allow each table to be variable in length and thereby make more efficient use of the available memory. (The downside is that the firmware would become more complex and require "garbage-collection" and de-fragmentation algorithms.)

### 4.0.5 Magnet transfer function in the WFG

As described in chapter 3, the (inverse) magnet transfer functions will reside permanently in one of the $H$ tables of magnet power supply WFGs.

### 4.0.6 Engineering units

Engineering units and conversions are handled in the ADO. There is no need to involve the WFG, which is really a device controller and should deal only with "raw" values.

### 4.0.7 Clock rates

All WFGs installed for power supplies in the sextant test will read RTDL frames at 720 Hz, and generate output at 720 Hz. Where the time-line is available, the WFG will use the 720 Hz event as an external trigger for timing; in rare cases where it is unavailable, an internal clock may be used. Other clock rates (such as 1000 Hz either internally or externally) are feasible (and may be of interest to the RF Group), but none is contemplated for controlling power supply ramps during the sextant test. The WFG may be driven at other rates, either by another external event or by the internal clock which has a resolution of 5 MHz. Rates of 1 kHz or faster should be no problem. Pending performance evaluation, it is expected that 5 kHz will be achievable, and rates up to 10 kHz may be possible.

## 5 Global save/restore: stepping stones

Suspension of disbelief is not necessary to imagine the following discourse, by a discouraged operator or commissioner:

> "Something has gotten worse since last Friday. Maybe it was our closed orbit correction yesterday, or maybe some other shift did something. Or maybe something is **really** wrong. Let's restore everything to the way it was last Friday."

This "global restore" of the machine parameters is not possible at Fermilab or at the AGS, where save/restore is done in a very piecemeal fashion. However, at those machines such as CESR where "global save/restore" is available, it is found to be very powerful, and life without it is hard to imagine.

In order for RHIC to have this capability, attention must be paid at the (present) design stage of the control system in general, and the ramps in particular. A full implementation of global save/restore goes beyond capturing the state of WFGs, and those aspects of the beam optics not controlled by WFGs, to include instrumentation settings. Nonetheless, this document focuses on WFG save/restore, probably the largest and hardest part of the whole problem.

Each table has many rows (up to 128), each of which pairs an argument value with an output value. When a table plays, its output is found by interpolation between neighboring rows. A small number of rows - perhaps 5 or

10 - correspond to "stepping stones", which are the focus of attention of ramp applications and managers at the console level. For example, in the low beta squeeze, there may be six stepping stones, corresponding to optics lattices in which $\beta^* = 10$, 8, 5, 3, 2, and 1 meter. In the abstract, a ramp plays from beginning to end by jumping from one stepping stone to the next.

These stepping stones exist as special rows in the equipment WFG tables, whether or not the ramp in question is mediated by a pseudo-time variable. **The "finite state" of the optics that is saved or restored corresponds to an assembly of one stepping stone value from each and every one of the equipment WFGs**. This "save set" is saved with a unique name. One ramp consists of a set of several such save sets - plus WFG row generation rules.

Note that this requires that ALL equipment WFGs have stepping stones defined at identical "times" of a ramp. This requirement introduces some inconveniences. A user cannot, for example, create new stepping stones for finer control of the chromaticity during the acceleration ramp, without also creating new stepping stones for all other variables. A specialized ramp editor such as the re-bucketing ramp application, which only affects a small number of WFG channels, must nonetheless align its stepping stones with those of all other equipment. However, it seems that these inconveniences need not be major, especially with a well designed ramp manager.

The layout of RHIC slots along the beamline, and the method of generating intermediate table rows from the stepping stones, are both necessary to reconstruct a complete model of a RHIC ramp from a number of save sets. Fortunately it is not necessary to save all the "implicit details" such as these with each save set, except as a date stamp or revision number. These rules change only rarely - it is (presumably) not very often that an extra magnet is added to RHIC, and it is not very often that the first step of the acceleration ramp (for example) is changed from quadratic to exponential.

# 6 Managers, Applications and the Sequencer

Especially when RHIC is in a luminosity production mode, it will be re-filled under the control of a high level "sequencer", which automates the complex dance of running many large and small application codes. In principle - when things are running smoothly - human operator intervention will hardly be necessary between the end of one data taking fill, and the beginning of the next.

After a successful commissioning phase, in which a thorough knowledge of the machine is gained, it might even be possible to refill RHIC using a "super-ramp", a concatenation of all basic ramps included in the fill procedure. Such a ramp would be executed after a start event without further interference of the high level programs.

Until that time the **sequencer** program will aide the operator by supervising all processes involved in the ramp process, performing error checking and comparrision of recorded machine behavior with the real machine. The sequencer will interact with "manager" processes. The title "manager" indicates that a process is persistent (always running) and manages the hardware resources (WFG tables, RTDL variables, events...).

The managers need as input a definition of the upcoming ramp and they output log files of the past ramp. The data flow between the processes supporting the ramps is shown in Figure 3.

A principle of the system design is the attempt to separate different components of set levels:

- The design set level is the part which results in an ideal machine in the desired behavior.
- The Trim set level is the change which is applied to counter the imperfections of the machine

Keeping the two components separate helps the operator to focus on what he wants to accomplish.

For example, the design set level for the quadrupoles is given by the optics design (tunes, betas) of the machine. If the measured tunes are different from the design, the operator will change the quadrupole trim settings to compensate for the machine imperfections. A large (compared to the design level) adjustment should alert the operator that something is wrong.

The tool to describe the "design ramp" is the **design ramp editor**. The tasks of this editor are:

- the definition of the pseudo-time vs. time function.
- the definition of stepping stones and interpolation method to generate intermediate values.
- the setting of ramp specific machine parameters, e.g. timing of the transition jump.
- the setting of some global parameters such as tunes, chromaticities etc.
- the definition of knobbing variables which will be active from the start of the ramp to the start of the next ramp.

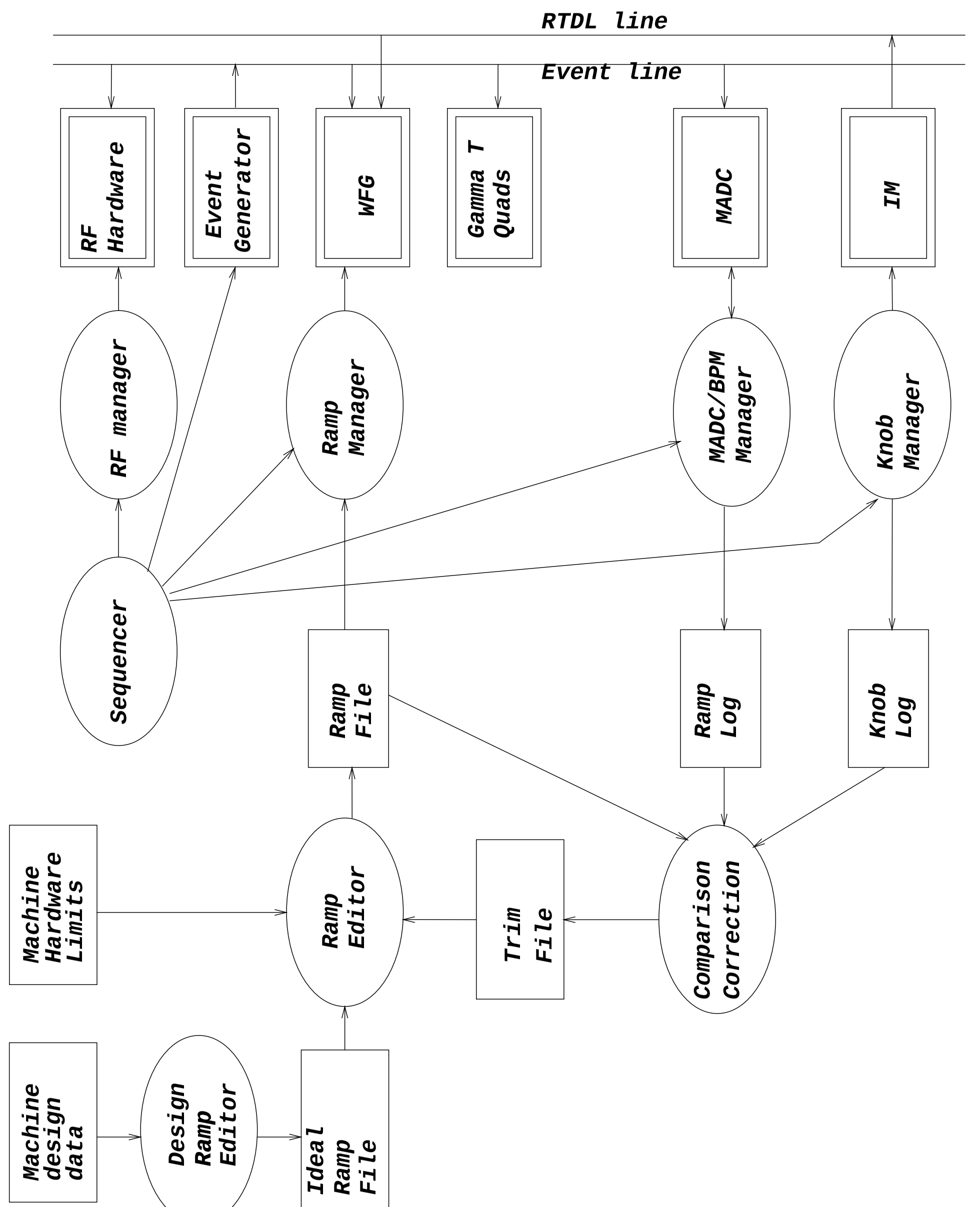


Figure 3: Data flow in the RAMP software system. Ovals, boxes and double-boxes represent processes, data files and hardware devices, respectively.

Input for the design ramp editor is the machine design information which is available from the SYBASE optics database.

Ramps are very different from each other. For example, the acceleration ramp is relatively complex, and involves all WFGs in the ring, while the re-bucketing ramp is relatively simple, directly involving only a small number of RF related WFG channels. This implies that, although there may be some exceptions, it is natural to develop one ramp editor module for each different kind of ramp.

- The Reset ramp is special because it is executed without beam. Synchronization and knobbing is not necessary; a pseudo-time is not defined. All magnets are cycled from minimum to maximum current a number of times and then all devices are set up for injection. The minimum, maximum and injection settings are obtained from a database. The operator specifies the number of cycles and has the possibility to restrict the minimum and maximum magnet currents to prevent "sick" magnets from quenching.

- Acceleration ramp is most complex. The pseudo-time is $B\rho$ of the dipoles. The ramp of amplitude of the $\gamma_t$ quadrupoles is part of the acceleration ramp but has its own pseudo-time to allow moving the jump point relative to the energy of the beam. The actual jump of the $\gamma_t$-quadrupole polarity is caused by a separate event whose delay from the start of the ramp must be defined. A third event tells the RF cavities to adjust their stable phases from $\phi_s$ to $\pi - 2\phi_s$.

  Other ramp specific parameters are the synchrotron tune and the bucket area. The required gap voltage can be calculated from these parameters and the change of the magnetic field[7].

  Although the phase of the vector sum of the gap voltage is controlled by a hardware feedback loop, the WFGs will be used to control the individual cavity phases. The design ramp editor is used to specify these phases to allow the operation of the cavities in normal and counter-phase mode.

- The Beta Squeeze ramp changes quadrupoles to modify the $\beta^*$ in the interaction regions. The pseudo-time is the percentage of $\beta$ modification.

- The Re-bucketing ramp has three steps:

  1. The RF phase is moved to position the unstable fixed point of the bucket to the center of the bunch. The bunch will lengthen.
  2. The stable fixed point of the bucket is moved back to the center of the bunch. The bunch rotates in phase space and becomes short.
  3. The acceleration cavities are turned off, the storage cavities are turned on.

The design ramp editor is used to set the timing of these events. The standard WFGs are not fast enough to perform this ramp. Specific electronics will be developed to perform this task For this reason there is also no pseudo-time.

- The Ring Synchronization ramp adjusts the relative phases of the Blue and Yellow ring so that the beams collide in the design interaction point. The exact mechanism of this ramp needs to be determined.

The result is stored in the "ideal ramp file".

The **ramp editor** adds the trims to the ideal ramp file. Initially all trim values are zero. New trim files with non-zero values will be generated during operations by correction programs. The output of the ramp editor program is the "ramp file" which contains all information necessary to execute the ramp.

The ramp editor program has also the task to scale existing trim files to fit new designed ideal ramp files. It has to make sure that limitations of the machine are not exceeded, i.e. maximum beam energy, B-dot etc.

The design ramp editor and the ramp editor program are used off-line to prepare the ramp. Under the supervision of the sequencer the manager programs are used to execute the ramp:

The **ramp manager** has the task to load the ramp tables into the WFGs assuring that current state of the machine is identical to the first stepping stone. It assigns RTDL variables to the knobbing parameters and connects the start of tables to different events. The ramp manager monitors the progress of the ramp and informs the sequencer when the ramp is completed.

The existing **power supply manager** (which was used to commission the AtR) will co-exist with the new manager during the sextant test. It will be appropriate for "DC" uses, such as steering the beam to the sextant dump.

The **MADC manager** and the **BPM manager** set up the hardware to start recording on the "start of ramp" event. At the end of the ramp these managers upload the recorded data and store them in a log file.

The **knobbing manager** accepts knobbing commands from the operator or other programs. It controls the input module, and records all knobbing actions in a log file.

The **RF manager** controls the part of the RF system that is not controlled by WFGs.

| Use | Count |
|---|---|
| RAMP WFGs | |
| Acceleration | 1 |
| | |
| EQUIPMENT WFGs | |
| Main power supplies (5 kA) | 2 |
| Shunt PS (50 V, 300 A) | 1 |
| Transition quad PS | 1 |
| Dipole corrector power supplies (50 A) | 33 |
| Spare corrector power supplies (50 A) | 7 |
| RF voltage (accelerate and storage) | 2 |
| RF phase (accelerate and storage) | 2 |
| Radial loop offset | 1 |
| | |
| TOTAL | 50 |

Table 4: Wave Form Generator channels that will be installed for the 1996 sextant test.

The **comparison/correction** program compares the newly generated log files with the design behavior of the machine as well as previously generated log files. It also contains correction algorithms which produce new trim files to improve the next ramp.

# 7 What is used in the Sextant test?

A single pseudo-time WFG will be installed for the sextant test, to generate the $B\rho$ pseudo-time RTDL frame. As shown in Table 4, the majority of equipment WFG channels will be connected to magnet power supplies, with only 5 channels attached to "active" RF devices. (Note that there are two channels on a single WFG card.) In fact, the number of fully installed WFG channels is limited by the number of power supplies available - there will be an excess of WFGs. This leaves open the possibility of making additional ramps available "in software" for the sextant test, using excess WFGs not connected to real equipment.

The nominal long range plan is to install a single RTDL line to serve both Blue and Yellow rings, although it may be possible/desirable/necessary to add

| RTDL variable name | Symbol | Units | Source |
|---|---|---|---|
| Measured current, main PS 1 | $I_1$ | A | Magnet PS |
| Measured $dI/dt$, PS 1 | $\dot{I}_1$ | $\mathrm{As}^{-1}$ | Magnet PS |
| Measured current, main PS 2 | $I_2$ | A | Magnet PS |
| Measured $dI/dt$, PS 2 | $\dot{I}_2$ | $\mathrm{As}^{-1}$ | Magnet PS |

Table 5: Variables that will be put on to RTDL from other sources, in 1996. The "units" are symbolic - actual RTDL variables are 24 bit integers.

one or more extra lines. Each line carries as many as 255 frames, with a guaranteed repetition rate of 720 Hz. In addition to ramp pseudo-times and knob increments, RTDL variables may be injected from other sources. In the sextant test, and as shown in Table 5, each of the two main power supplies will each insert $I$ and $\dot{I}$ frames. These are the only additional frames that are foreseen.

RTDL variables are very powerful in allowing software testing, with busses, et cetera, inactive. Fermilab makes much use of its "RTDL" line in this way. For this and other reasons, it is likely that many more RTDL frames will be added in the long run.

# 8 Conclusions

The number of conceptually distinct WFGs layers has been reduced to 2 from the 3 quoted in a previous discussion of ramps. A simpler way to provide "knobbing" capabilities, during and after ramping, has been devised.

It is possible and highly desirable to perform global save/restore of the ramp "stepping stones", in which the device set point of every WFG channel is captured simultaneously.

The specific 1996 plans that are described, above, form a large scale "road map" for ramping activities in the sextant test. When sextant test experience is under our belt(s), reasonable modifications of the general scheme - for example, quadratic or cubic table interpolation - will probably be necessary.

## Acknowledgments

We would like to thank many people for the enthusiastic contributions that they have made to this paper. In particular, the other members of the ramp working group - Leif Ahrens, Fulvia Pilat, and Todd Satogata - made many useful comments. George Ganetis, Bob Lambiase, and Mike Brennan gave us significant insight to their plans. Last, but not least, Heather Hartmann and Brian Oerter kept us up to speed with the state of the WFG hardware.

## References

[1] T. Clifford, V. Mane, S. Peggs, "RHIC Ramps: a proposal", RHIC/AP/16, December 1993.

[2] H. Hartmann, "Specifications for the RHIC Real Time Data Link transmitter system", October 1993.

[3] H. Hartmann, B. Oerter, S. Peggs, "RHIC Real Time Data Link System", October 1993.

[4] T. Kahn, "RHIC Power Supply Waveform Generator Board Application Notes", revised October 1994.

[5] T. Kahn, "Power Supply Waveform Generator Programmer Manual", March 1995. October, 1994.

[6] For more WFG information, see http://www.rhichome.bnl.gov:80/RHIC/Controls/Documentation

[7] D. P. Deng, "BBAT: bunch and bucket analysis tool", Particle Accelerator Conference 1995.