# Booster IRM Timing

*Analysis of synchronization*

Fri, Apr 7, 2000

The IRMs that serve as control system front ends for Booster HLRF use a timing delay of 35 ms to deliver the interrupt that begins their 15Hz cycle activity. This delay is measured from the occurrence of the 15Hz reset clock event, which occurs 2 ms before BMIN, which is the time of Linac beam injection into the Booster. This means that the interrupt time occurs 33 ms after BMIN. But Booster beam extraction occurs at the time of BMAX, which is 33.5 ms after BMIN.

The IRM interrupt therefore occurs about 0.5 ms ahead of beam extraction. When updating the data pool, the IRM reads the most recent full set of 64 channels whose readings are found in the circular memory buffer that holds the most recent half second of digitizations. Since the 1KHz digitizations of the 64 channels are asynchronous with everything, the data that is captured may therefore come from 1–2 ms earlier than the time it is sampled. From Data Access Table timing diagnostics, the 64 channels are captured from the circular hardware during 0.2–0.3 ms after commencement of Data Access Table processing, which itself begins about 0.1 ms after the 15Hz interrupt. The delay before accessing the circular buffer memory is due to sampling the digital data before the A/D readings are sampled.

***Extraction time squeeze***

There is a time squeeze that is quite tight. Part of the digital data that is read includes a bit to indicate whether the RF is high enough to qualify as a valid RF signal. This bit is sampled after the interrupt; it is not recent history as in the analog case. This bit will be present only as long as RF is present. The IRM interrupt is timed to come at beam extraction time, after which the RF system could choose to turn off the RF, which would turn off this bit. By waiting until beam extraction time, we have only a brief window of time during which we can expect to find valid RF status.

Beyond the question of the short time between extraction and the demise of the RF status signal, there is also the fact that the IRM is not purely interrupt driven. It receives an interrupt that signals the start of its cycle, but its activities are scheduled to run at task level. All tasks have the same priority, and time slicing is not used, so any currently executing task must complete before the Update task can begin processing the Data Access Table to update its data pool. If a complicated data request is received just before the 15Hz cycle interrupt occurs, then this can introduce additional delay before the usual cyclic activities commence.

If we operate the Booster HLRF nodes at an earlier time in the cycle, most data sources will be valid. Many RF signals are sampled midway through the RF cycle. As long as signals are not needed that can only be valid late in the cycle, this would work well enough. But one RF system is used for connecting to beam budget monitoring. Its signals are not available until some time shortly after beam extraction. That station must be triggered to operate later than 35 ms, perhaps 40 ms.

The problem with nodes operating asynchronously within a project shows up when server/consolidator support is considered. If one of the contributing nodes to a data request is running a few ms later than the rest, including the server node, there should be no serious problem. The late node's contributions to a data request will arrive a few ms later than the rest, but it will be long before the 40 ms deadline that the server node imposes. The problem arises when the late node is a server to a given request that includes data from other nodes. Replies from contributing nodes will arrive earlier than expected, and perhaps even before the server node's cycle interrupt. This means that the server node will think that the contributing node replies are missing for that cycle, because the current cycle's data arrived late on what is considered by the server to be the previous cycle.

Acnet devices are defined to be associated with a source node. To provide for support for correlated data, all Acnet devices within a project are defined to be sourced from one of the nodes of that project. That node is considered the server node for Acnet requests made for data from that project. So the choice of server node is under control of the DABBEL user.

For Classic protocol, however, things are more fluid. For a list of devices on a parameter page, for example, the first device might determine the server node for a request for all data on the page. So the server node could be any node in the project. There may be cases for which a server node is selected that is a late node, so that errors will occur as a result of the server node missing replies from the others.

By testing with the Macintosh parameter page, the errors (type "7" errors) occur only when the data request period is 15Hz. This is because the tardy status is assigned only if the data is old by at least the reply period. If the reply period is two cycles, the data is not considered old if it is one cycle old. But at 15Hz, when correlated data is sought, one will get errors in this case.