Booster Beam Loss Monitor
Data Acquisition and Presentation
Specification

October 13, 2006

Craig Drennan

Table of Contents

Table of Contents..2
I. Introduction.. 3
II. BLM Digitizer Module Data Acquisition.. 3
III. Signal Processing: Computing of Various Sums..4
III.1 The Base 80 us Integration Samples... 5
III.3 The 1 ms Integration Samples...6
III.4 The 100 Second Moving Sums... 6
III.5 The 7.5 Hz Waveform Buffers...7
IV Serving Data for Console Display.. 7
IV.1 Snapshot Plot Support...7
Request Plot Class Information, type code 1...7
Setup A Snapshot, Type Code 7...8
V. Summary of the BLM DAQ Process.. 13

I. Introduction

This note is to specify and document how the Booster Beam Loss Monitor data is collected using the recently developed BLM Integrator/Digitzer VME modules. The following sections will describe how the BLM signals are digitized and the data managed. It will describe the different types of sums that are computed from this data. It will also describe how the front end crate processor that manages the ACNET requests for this data manages these requests.

II. BLM Digitizer Module Data Acquisition

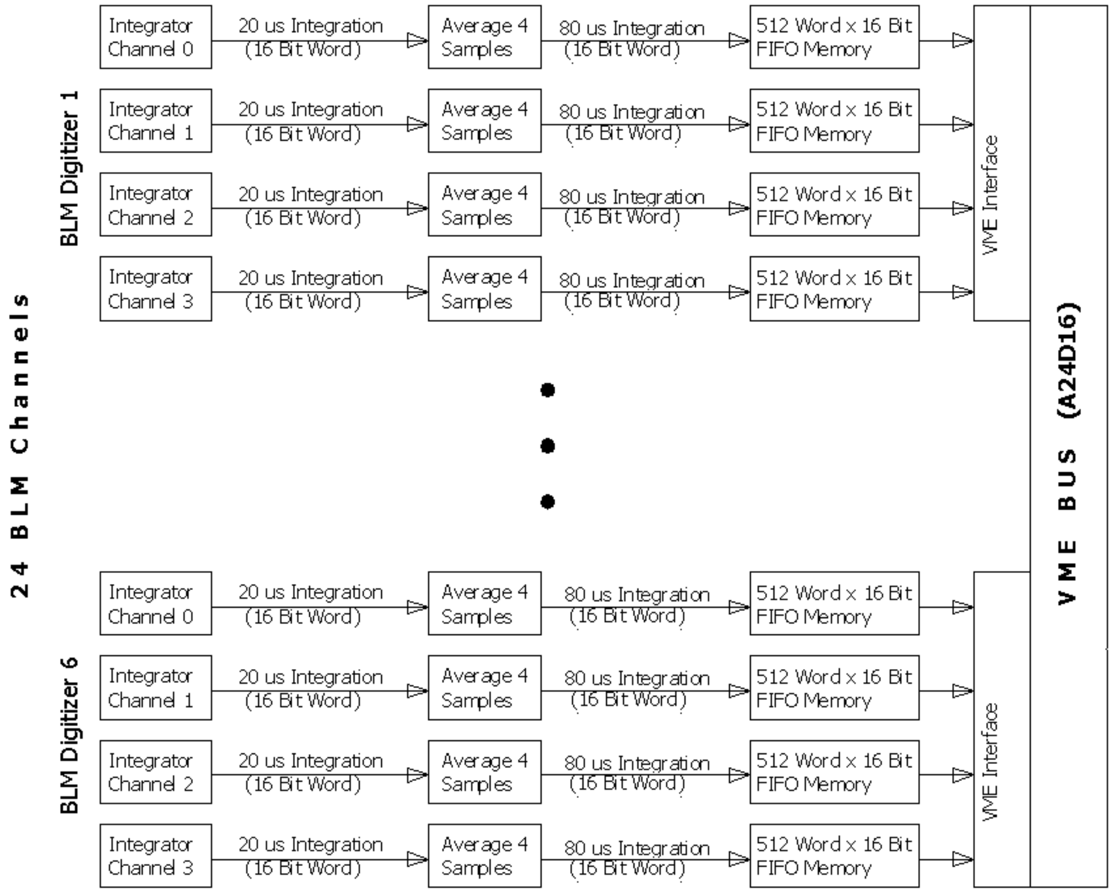
Integration and digitization of the loss monitor signals is performed by the BLM Digitizer Module. The data representing the digitized signal is buffered on the BLM Digitizer Module using FIFO memory. The output of this memory is accessible to the crate processor via the VME bus.

The module performs the following functions.

1. Digitizes the results of a 20.08 us integration interval into a 16 Bit Word.
2. Every 80.32 us an average of 4 each 20.08 us integrations produces a 16 Bit Word.

   Note that this is not strictly an 80.32 us integrated value, but more approximately an 80.32 us integrated value divided by 4.
3. Data is collected for 40.16 ms per Booster cycle, resulting in 500 samples per cycle.
4. The conservative estimate for transferring one 16 Bit Word over VME is 1.0 us. This leads to a total time of 12 ms to transfer one cycle of data for 24 channels from the Digitizer modules to the MVME processor board.

   (500 samples / channel)*(1 accesses / sample)*(24 channels)*(1.0 us / access) = 12 ms



III. Signal Processing: Computing of Various Sums

Recall that within the Digitizer card the charge produced by the BLM ion chamber is integrated, or summed, over a 20 us interval and is then digitized to produce a number. These 20 us samples are summed into 80 us samples. The 80 us samples are divided by 4 to reduce the word size for transfer over the VME bus.

Within the crate processor the BLM data is stored as several different types of sums. Once the 80 us samples are transferred to the processor, they are summed to represent longer intervals of time, and they are summed in distinctly different manners to represent the accumulation of BLM charge (beam loss) in different ways.

III.1 The Base 80 us Integration Samples.

There is a 500 point buffer for each of the 24 channels of 80 us integrated data read from the BLM Digitizers over the VME bus. This data is used to produce the other forms of data described below.

III.2 The Full Cycle Sampled Accumulation.

Each cycle, further processing of the BLM integration data is performed and additional buffers are updated. The data from the previous cycle is summed into 500 *floating point values* of a continuously integrating signal. That is,

$$y[i]=y[i-1]+x[i] \; for\, i=1 \ldots 499$$

$$y[0]=x[0]$$

where $y[i]$ are the continuously integrating signal samples and $x[i]$ are the 80 us integration samples. There is a 500 point buffer of this kind for each of the 24 Booster channels. Typically it is the last sample in this accumulation that is reported if it is desired to retrieve a single value that represents the loss during a specific cycle.

III.3 The 1 ms Integration Samples.

Each cycle, the data from the previous cycle is summed into 40 each 1 ms sums. That is,

$w[0]=x[0]+x[1]+\ldots+x[11]$ (sum of 12 values)

$w[1]=x[12]+x[13]+\ldots+x[23]$ (sum of 12 values)

$w[2]=x[24]+x[25]+\ldots+x[35]$ (sum of 12 values)

o

o

o

$w[39]=x[488]+x[489]+\ldots+x[499]$ (sum of 12 values)

where w[i] are the 1 ms sums. These sums are double precision floating point values. There is a 40 point buffer of this kind for each Booster cycle type, for each BLM channel. That is 288 of this kind of buffer.

III.4 The 100 Second Moving Sums

For each booster cycle type, for each BLM channel, 100 second moving sums are maintained. The 100 second sum is the sum of 6 each 17 second sums. These 17 second sums are stored in a circular buffer, 6 values deep. Each time a new 17 second sum is added to the 100 second sum, the oldest 17 second sum in the circular buffer is subtracted off.

In order to compute the 17 second sums (which are actually 250, 15 Hz cycles) sum registers are maintained for each Booster cycle, for each BLM channel. When processing the data for a specific Booster cycle type the final value of the "Full Cycle Sampled Accumulation" (y[499] from equation XX above) is added to the 17 second sum value for that specific cycle type, for the specific BLM channel.

When a counter counting 15 Hz cycles reaches 250 (~17 seconds) the 100 second sums and their associated circular buffers are updated with the values in the 17 second sum registers. Then the 17 second sums are reset to zero.

Just to reiterate, there is a 17 second sum register and a 6 deep circular buffer of 17 second sums for each of the 12 Booster cycle type, for each of the 24 BLM channels.

In addition to maintaining these 100 second moving beam loss values, a 100 second moving count of the occurrence of the specific Booster reset events (those triggers which initiate the different Booster cycle types) are maintained. These are also updated by maintaining 17 second counts of the Booster reset events and 6 deep circular buffers of the 17 second counts. In this case there are only 12 sets of counts and circular buffers. One set for each Booster reset event.

III.5 The 7.5 Hz Waveform Buffers

Requests may be made from ACNET applications to receive data for a specific set of channels on the BLM front-end processor at a 7.5 Hz update rate. Since the Booster cycles at a 15 Hz rate, two cycles worth of data are returned at the 7.5 Hz rate. The data returned is for the specified channel with no distinction with regard to the type of Booster cycle the data was collected over or whether there was even beam in the Booster during the interval. In addition to BLM data channels, there are channels that report the specific Booster reset events that may or may not have occurred over the last 133 ms (inverse of 7.5 Hz). Also along with the data is included the specific "cycle counts" for the two cycles of data in the update response. The cycle count information can be used to correlate the Booster reset event information with the data taken during the cycle the reset event triggered.

The BLM channel data will be the 500 point Full Cycle Sampled Accumulation waveform. For each BLM channel, 2 of these waveforms are transmitted to the requesting ACNET application every 133 ms.

IV Serving Data for Console Display

The front end processors in the BLM crates will respond to ACNET Fast Time Plot and Snapshot Plot requests as specified in ACNET Design Note No. 49.9. Details of the communication protocols are given below for the specific case of the Beam Loss Monitor data.

IV.1 Snapshot Plot Support

Request Plot Class Information, type code 1
Before making requests to setup a continuous or snapshot plot, the plotting application will make a request to the front ends involved to retrieve plot class information for the devices about to be plotted. The plotting application uses the class codes to retrieve a structure which contains information about the device's plotting capabilities. ACNET Design Note No. 49.9 describes the format of the request to retrieve plot class information and the format of the reply. For the sake of the new BLM devices a new Snapshot plot class will be created. The information for this new class is given in Table IV.1.1 below.

Table IV.1.1 Information for Class 28, “Booster Instrumentation 1”

Parameter	Value
Supports Type 7 Requests	TRUE
Returns Timestamps	FALSE
Fixed Sample Rate	FALSE
Max Sample Rate	12.45 kHz
Maximum Points per Plot	4096

Setup A Snapshot, Type Code 7
This type code replaces type code 3 which is being retired. To setup a snapshot plot a console plotting task sends a “multiple-reply” request to the appropriate front ends. The format for this request with comments related to the new Booster Instrumentation class is given in Table IV.1.2.

Upon receipt of a snapshot setup request, the front end will first cancel any previous continuous or snapshot request from the task making the requesting. Then the front end will need to ensure that resources are available to service the request. If any of the requested devices can be accommodated, the entire request is considered successful.

Periodic status replies will be issued from the front end to the task that made the plot request. The format for these replies is given in Acnet Design Note 49.9 and here with comments in Table IV.1.3. Within the reply the front end may change various parameters in the request such as Sample Rate or the Arm and Trigger Selection Word to match the capabilities of the BLM data acquisition.

These periodic replies will report the state of the device plot service. This status is either snapshot pending waiting for the arm event, the time delay or completion of the data collection, or the status will report that data collection is complete.

Table IV.1.2 Setup Snapshot Request Format

Length (words)	Field Name	Description
1	type code (=7)	Identifies the type of request.
2	Requesting Task Name	Name of plotting task making the request.
1	Number of Devices Requested	As implied. There will be this many "Snapshot Request Packets" included at the end of this message.
1	Arm and Trigger Selection Word	For the Booster Instrumentation class this value is expected to be 0x0042(hex). This declares that the 1. Arming source is the logical OR of the arm clock events. 2. The value "N" specified in this request is interpreted as the number of microseconds into the device data buffer, of the cycle the plot data collection is first armed, that data is copied from the device data buffer into the plot data buffer. 3. Data is sampled from the device data buffers and copied into the plot data buffers at intervals that emulated an analog signal being sampled at the Data Sampling Rate specified in this request.
1	Priority	A low priority request should be terminated if the resource it holds are needed by an incoming higher priority request.
2	Data Sample Rate "Q" in Hz	
2	Arm Delay "N" in microseconds	
4	Arm Clock Events (8 bytes)	This word specifies the Tclk events that are logically OR'd to arm the data acquisition. The word has one event per byte. The hex value FF is used to designate no event.
2	Sample Trigger Clock Events	Not used by the Booster Instrumentation Class
2	Number of Points	This is the requested number of data points to return to the requesting plotting task.
2	Device Arm DI/PI	Ignore This Field – Cryo use only
2	Device Arm Offset	Ignore This Field – Cryo use only
4	Device Arm SSDN	Ignore This Field – Cryo use only
2	Device Arm Mask	Ignore This Field – Cryo use only
2	Device Arm Value	Ignore This Field – Cryo use only
4	Zeros for expansion	Reserved space
10	Snapshot Request Packet 1	See Acnet Design Note 49.9 for the format of this packet.
10	Snapshot Request Packet 2	
:		
10	Snapshot Request Packet 'n'	

Table IV.1.3 Format of type code 7 reply to setup a Snapshot.

Length (words)	Field Name	Description
1	ACNET format return status	
1	Arm and Trigger Selection Word	For the Booster Instrumentation class this value is expected to be 0x0042(hex). This declares that the 1. Arming source is the logical OR of the arm clock events. 2. The value "N" specified in this request is interpreted as the number of microseconds into the device data buffer, of the cycle the plot data collection is first armed, that data is copied from the device data buffer into the plot data buffer. 3. Data is sampled from the device data buffers and copied into the plot data buffers at intervals that emulated an analog signal being sampled at the Data Sampling Rate specified in this request.
2	Data Sample Rate in Hz	This is going to be the value = (12450 / IntegerPart[12450/Q]) where Q is the data sample rate requested in the Setup Snapshot Request message.
2	Arm Delay in microseconds	Does the plotting routines need for us to recompute the actual delay in microseconds used ?? This would be = 80 * IntegerPart[N / 80], where N is the arm delay in microseconds requested in the Setup Snapshot Request message.
4	Arm Clock Events (8 bytes)	This word specifies the Tclk events that are logically OR'd to arm the data acquisition. The word has one event per byte. The hex value FF is used to designate no event. This is a copy of what was in the Setup Snapshot Request message.
2	Number of Points	This is the number of data points that will be returned to the requesting plotting task.
9	Device reply packet for Device 1	See ACNET Design Note 49.9 for the format of this packet. The fields in this packet will be all zeros except for the first word which is the status for the device and will have one of the following values 0 Snapshot data collection is complete FEPNDA Snapshot pending, waiting for arm event FEPNDB Snapshot pending, waiting for time delay. FEPNDC Snapshot pending, collecting data.
:		
9	Device reply packet for Device n	As above.

When a Setup Snapshot Request message is received, plot data buffers (one for each device to be plotted) will be established with the requested length (number of points). These plot data buffers will begin being loaded with device data from the front end device buffers after the *completion* of a Booster cycle that was triggered by one of the specified Arm Clock Events (Tclk reset event). After this, the plot data buffers will continue to be loaded each 15 Hz cycle until the plot data buffers are filled.

The Arm Delay "N" specified in this request is interpreted as the number of microseconds into the device data buffer that data is copied from the device data buffer into the plot data buffer, for the 15 Hz cycle the plot data collection is first armed. An illustration of what is meant is given in Figure IV.1.1. The integer value "M" is the number of 80 us samples into the device data buffer of the initial cycle from which we start copying data into the plot data buffer. The value of "M" is the integer part of the quotient (N / 80).

The BLM data is sampled at 12.45 kHz. If the requested Data Sample Rate "Q" is greater than this the reply to the setup request will report that the 12.45 kHz rate will be used. If the requested Data Sample Rate "Q" is less than 12.45 kHz then the closest sample rate that can be derived from the 12.45 kHz data will be computed and reported in the reply to the setup request. A value "R" will be computed such that by copying every Rth value from the device data buffers into the plot data buffers we can return data to the plotting task that represents this new sample rate. The value "R" will be equal to the integer part of the quotient (12450 / Q). The resulting sample rate reported back to the plotting task in the setup reply will be (12450 / R).

Listing IV.1.1 further illustrates how the plot data is copied from the device data.

Listing IV.1.1

```
Plot data [0]     = Cycle 1 Device Data[M]
Plot data [1]     = Cycle 1 Device Data[M + R]
Plot data [2]     = Cycle 1 Device Data[M + 2*R]
 o
 o
 o
Plot data [J]     = Cycle 1 Device Data[M + J*R]

Plot data [J + 1] = Cycle 2 Device Data[0]
Plot data [J + 2] = Cycle 2 Device Data[R]
Plot data [J + 3] = Cycle 2 Device Data[2*R]
 o
 o
 o
```

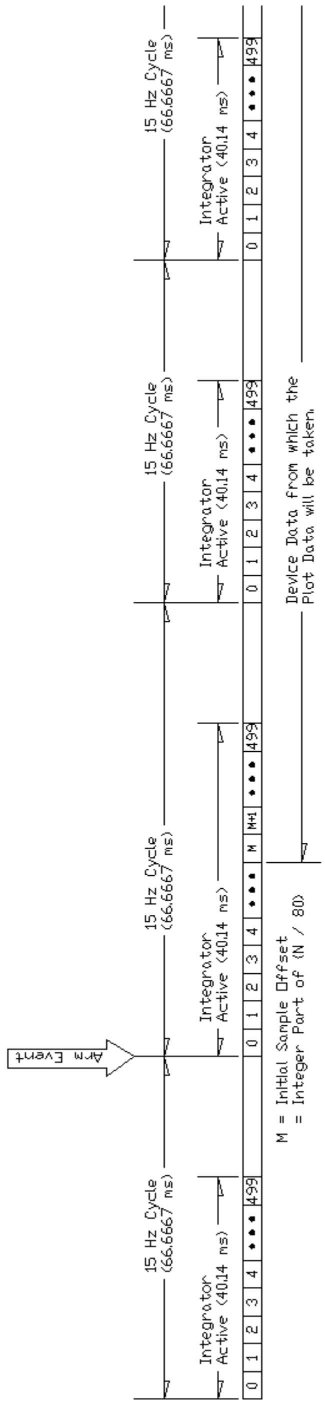


Figure IV.1.1 Illustration of how the Arm Delay is implemented.

V. Summary of the BLM DAQ Process

Steps in the Process:

1. Signal digitization and collection is triggered every 15 Hz cycle on event $11.
2. The BLM signals are digitized at a rate of 12.45 kHz. Each sample represents an 80.32 us integration interval.
3. After being triggered by event $11, the BLM signals are integrated and digitized for 40.14 ms, producing a FIFO buffer of 500 data points for each device.
4. The front end processor begins collecting data from the BLM Digitizer modules over the VME bus approximately 40 ms after event $11.
5. It is expected that the front end processor will have collected the data from 24 BLM channels by 52 ms after event $11.
6. At this point in the 15 Hz cycle 24 each 500 point buffers have been filled with the BLM data for the *current* cycle. This is the “Base 80 us Integrated Data”.
7. 14 ms remain before the end of the 15 Hz cycle and 54 ms remain before the next VME bus transfer must begin again.
8. From the 24 buffers of 80 us Integrated Data the 24 buffers of “Full Cycle Sampled Accumulation” data are computed.
9. From the 24 buffers of 80 us Integrated Data the “1 ms Integration Samples” buffers are computed.
10. Using the final values of the “Full Cycle Sampled Accumulation” buffers the 17 second sums are updated and every 250^{th} 15 Hz cycle the “100 Second Moving Sums” are updated.
11. In the remaining time before the front end is required to begin the VME bus transfers with the BLM Digitizer modules, the front end must service the ACNET requests for data. These include the data logging requests for the “1 ms Integrated Samples”, the comfort display requests for the “100 Second Moving Sums” data, and Snapshot or other plotting application requests.

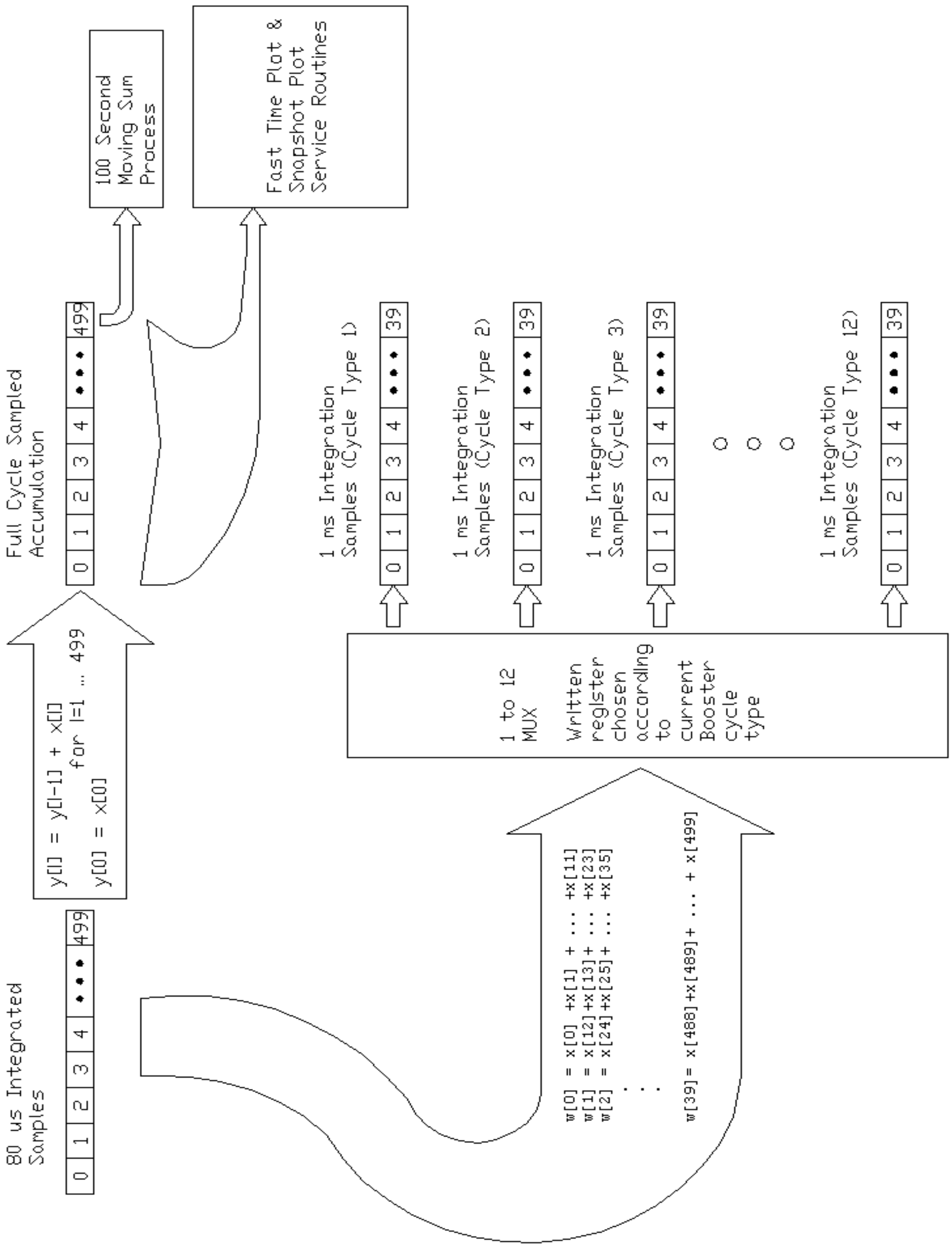
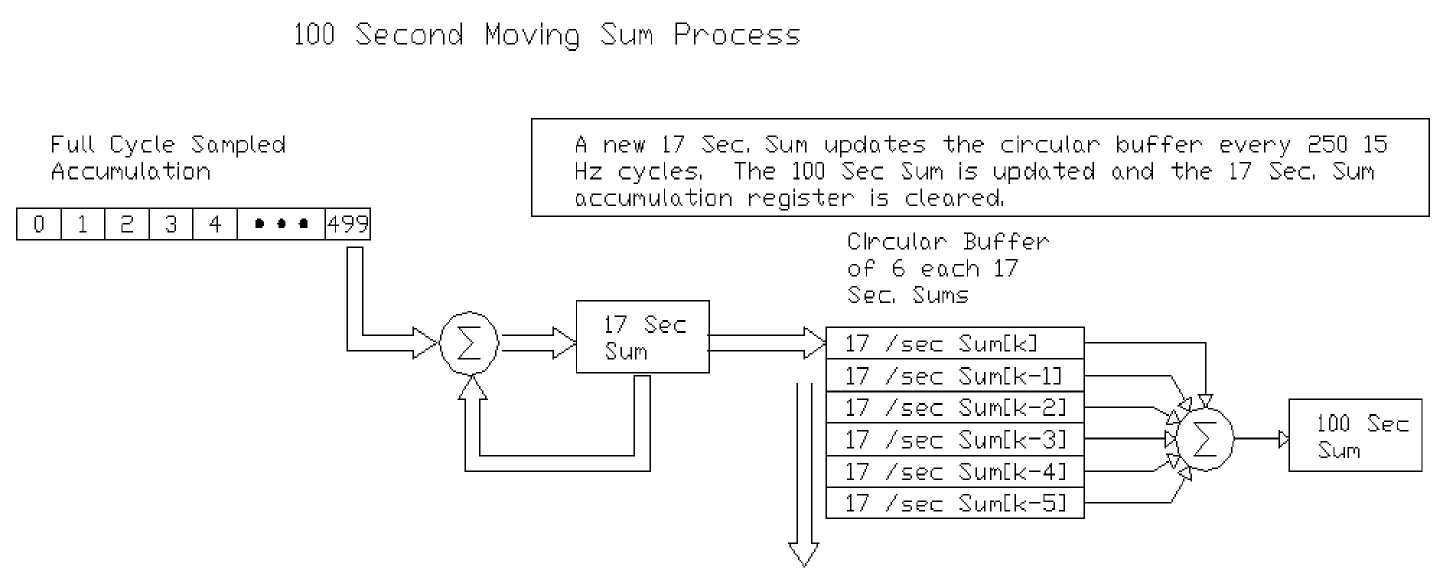

80 us Integrated Samples
Full Cycle Sampled Accumulation
y[I] = y[I-1] + x[I] for I=1 ... 499
y[0] = x[0]
100 Second Moving Sum Process
Fast Time Plot & Snapshot Plot Service Routines
1 ms Integration Samples (Cycle Type 1)
1 ms Integration Samples (Cycle Type 2)
1 ms Integration Samples (Cycle Type 3)
1 ms Integration Samples (Cycle Type 12)
1 to 12 MUX
Written register chosen according to current Booster cycle type
w[0] = x[0] +x[1] + ... +x[11]
w[1] = x[12]+x[13]+ ... +x[23]
w[2] = x[24]+x[25]+ ... +x[35]
w[39]= x[488]+x[489]+ ... + x[499]

Figure ###6


100 Second Moving Sum Process
Full Cycle Sampled Accumulation
0
1
2
3
4
499
A new 17 Sec. Sum updates the circular buffer every 250 15 Hz cycles. The 100 Sec Sum is updated and the 17 Sec. Sum accumulation register is cleared.
Circular Buffer of 6 each 17 Sec. Sums
17 Sec Sum
17 /sec Sum[k]
17 /sec Sum[k-1]
17 /sec Sum[k-2]
17 /sec Sum[k-3]
17 /sec Sum[k-4]
17 /sec Sum[k-5]
100 Sec Sum