METRIX
Experience Value

Application Note

Understanding SETPOINT™ Waveform Collection and the i-value Algorithm

Q1: How does Metrix define static data?

A1: Static data is an attribute of a waveform, such as the peak-peak amplitude, the 1X filtered amplitude, the gap voltage, the 1X phase lag, etc. As such, a waveform may have numerous static values associated with it. Static data is also generally understood to include status information, such as whether a measurement is in an alarm condition, NOT OK condition, etc.

Q2: How does Metrix define dynamic data?

A2: Dynamic data is simply a waveform. It is a time record of the instantaneous changes in the data, sampled at or above the Nyquist criteria to ensure aliasing does not occur. Waveforms are generally understood to be a time-domain representation of the signal. Frequency-domain representations (i.e., spectra) are computed from the time-domain. Data in the Metrix SETPOINT system is always stored in the time-domain format (time series of samples). The frequency-domain representations are computed on-demand from the time-domain data; data is not stored in frequency-domain format.

Q3: How does Metrix define Δrpm data?

A3: Δrpm data is collected at fixed rpm intervals. The Metrix data collection algorithm used in SETPOINT collects data based on changes in numerous variables, not just rpm. See Q11 for additional details.

Q4: How does Metrix define synchronous waveform data?

A3: Synchronous waveform data consists of samples equally spaced in phase (i.e. shaft rotative angle). To collect such data, a once-per-turn reference signal is required in addition to the signal being sampled. This reference signal is referred to as a phase trigger. Synchronous data does not use an anti-alias filter.

Q5: How does Metrix define asynchronous waveform data?

A5: Asynchronous waveform data consists of samples equally spaced in time, without reference to any external phase trigger. As such, asynchronous waveforms do not require a phase trigger. Asynchronous data uses an anti-alias filter.

Q6: Does the SETPOINT system collect both asynchronous and synchronous waveform data?

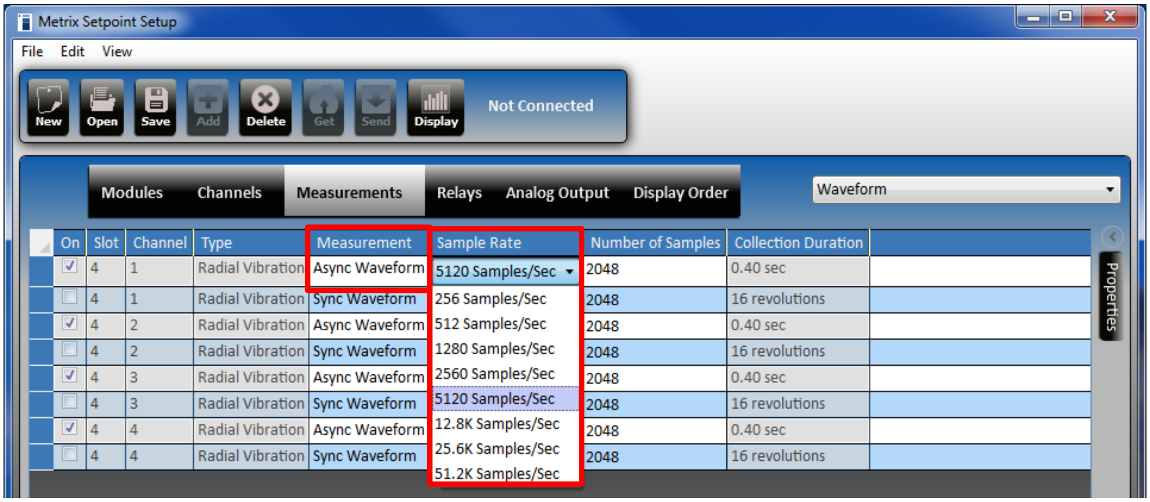
A6: Yes. UMM_{CM} channels can return asynchronous waveforms, regardless of whether a phase trigger has been associated with the channel or not. In addition to the asynchronous waveforms, synchronous waveforms can also be returned if a phase trigger is present and has been associated with the channel in the configuration software. The user may select asynchronous, synchronous, or both. See Figures 1 and 2 on the following page.

Q7: What is the basic communications architecture in the SETPOINT system?

A7: Each UMM_{CM} and TMM_{CM} in the rack sends its data to the rack's SAM via a backplane network known as "CM-NET." TMM_{CM} channels collect only static data, updated every 500 ms and sent to the SAM. This collection and transmission rate is not user-configurable. UMM_{CM} channels collect both static and dynamic (waveform) data. A UMM_{CM} channel's static data is updated every 80 ms and sent to the SAM. This collection and transmission rate is not user-configurable. Refer to Tables 1 and 2 in sections 1.5.1 and 1.5.2 in SETPOINT MPS manual 1079330 for a list of static data returned by each channel type. Depending on channel type, as few as 1 or as many as 12 static data types per channel can be returned. Channel alarm status is also returned.

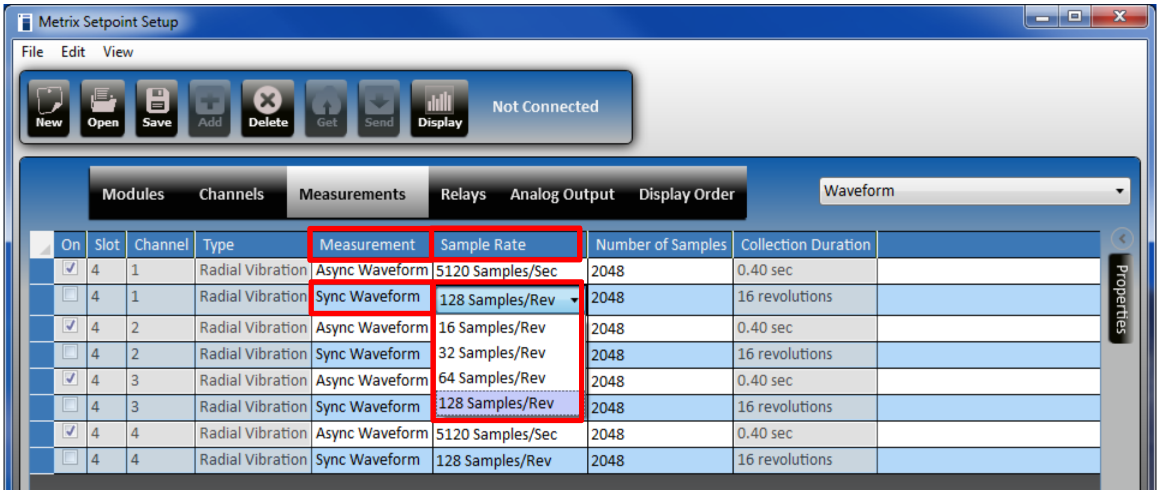
UMM$_{CM}$ Waveform Sampling			
		Asynchronous	**Synchronous**
Configurable Settings	**Number of Samples**	1024 or 2048	1024 or 2048
	Sample Rate	256 512 1280 2560 5120 12.8 K 25.6 K 51.2 K	16 32 64 128

Unlike static data, the rate at which dynamic (i.e., waveform) data is sampled and sent is user-configurable. While every cycle of every waveform is collected continuously by each UMM$_{CM}$ channel, this data is decimated inside the UMM so that only waveforms representing sufficient change above a threshold value are sent. This is a more efficient means of processing and transmitting data than if all samples were sent to the SAM and/or software. The sample rate of the synchronous waveform is independently configurable from the asynchronous waveform as shown in the table at right and in the rack configuration software screen captures (Figures 1 and 2) below.



On	Slot	Channel	Type	Measurement	Sample Rate	Number of Samples	Collection Duration
☑	4	1	Radial Vibration	Async Waveform	5120 Samples/Sec	2048	0.40 sec
□	4	1	Radial Vibration	Sync Waveform	256 Samples/Sec	2048	16 revolutions
☑	4	2	Radial Vibration	Async Waveform	512 Samples/Sec	2048	0.40 sec
□	4	2	Radial Vibration	Sync Waveform	1280 Samples/Sec	2048	16 revolutions
☑	4	3	Radial Vibration	Async Waveform	2560 Samples/Sec	2048	0.40 sec
□	4	3	Radial Vibration	Sync Waveform	5120 Samples/Sec	2048	16 revolutions
☑	4	4	Radial Vibration	Async Waveform	12.8K Samples/Sec	2048	0.40 sec
□	4	4	Radial Vibration	Sync Waveform	25.6K Samples/Sec	2048	16 revolutions
					51.2K Samples/Sec		

Figure 1: Asynchronous sampling configuration options



On	Slot	Channel	Type	Measurement	Sample Rate	Number of Samples	Collection Duration
☑	4	1	Radial Vibration	Async Waveform	5120 Samples/Sec	2048	0.40 sec
□	4	1	Radial Vibration	Sync Waveform	128 Samples/Rev	2048	16 revolutions
☑	4	2	Radial Vibration	Async Waveform	16 Samples/Rev	2048	0.40 sec
□	4	2	Radial Vibration	Sync Waveform	32 Samples/Rev	2048	16 revolutions
☑	4	3	Radial Vibration	Async Waveform	64 Samples/Rev	2048	0.40 sec
□	4	3	Radial Vibration	Sync Waveform	128 Samples/Rev	2048	16 revolutions
☑	4	4	Radial Vibration	Async Waveform	5120 Samples/Sec	2048	0.40 sec
□	4	4	Radial Vibration	Sync Waveform	128 Samples/Rev	2048	16 revolutions

Figure 2: Synchronous sampling configuration options

Q8: What is the bandwidth of CM-NET?

A8: CM-NET has been sized such that one synchronous and one asynchronous waveform from every channel in the rack can be returned every 5 seconds (in addition to 80ms static data). For a 16-slot rack with 14 UMM$_{CM}$ modules, this represents 1344 waveforms per minute (14 UMMs x 4 channels/UMM x 2 waveforms/channel x waveform/5 sec x 60 sec/min). This rating is conservative; we have not yet tested CM-NET to higher speeds, but more throughput may be feasible. We do not, however, believe that continuous streaming of every waveform from every channel is practical (see Q9 below).

Q9: If Metrix is actually sampling each and every waveform continuously, why don't you send all of this data to the SAM and then on to external software where it would/could be processed?

A9: The primary reason for not returning every waveform to the SAM is that sending every waveform would require substantial bandwidth, processing power, and system memory; it is more efficient to perform the decimation algorithm inside each UMM, allowing it to deciding whether a waveform has changed sufficiently from a prior sample to merit transmission to the SAM. To put this into perspective, consider a rack with 14 UMM$_{CM}$ modules (56 channels) monitoring a machine train running at 10,000 rpm with an asynchronous sample rate of 51,200 samples/sec and a synchronous sample rate of 128 samples/rev. If all data were streamed and stored at the SAM or in the host software, this would correspond to about 130 Mbit/sec, or 58 GB/hour per the table below. Note further that this considers only waveform data, not the additional bytes required by 80ms static data which although considerably smaller, still amounts to ~120 MB/hr.

Async samples/sec	51,200 samples/sec x 56 channels = 2,867,200 samples/sec	
Sync samples/sec	128 samples/rev x 10,000 rpm/60 x 56 channels = 1,194,667 samples/sec	
TOTAL	Samples/sec	**4,061,867 samples/sec**
	Bits/sec	**130 Mbit/sec** (4,061,867 samples/sec x 32 bits/sample = 129,979,744 bits/sec)
	GB/hr	**58.49 GB/hr** (129,979,744 bits/sec x 1 byte/8 bits x 3600 sec/hr)

Q10: What if a customer wants to do special processing on the raw waveform data, such as computation of eigenvalues?

A10: For the reasons noted in Q9, the data throughput that would be required if we returned every waveform from all channels continuously is not practical from a processing standpoint in our SAM (which uses an Intel ATOM processor) or from a storage perspective (approximately 1 GB/min or 1.4 TB/day). Eigenvalues could be computed from either the waveforms currently returned (2 waveforms / channel at rates up to every 5 seconds) or Metrix could conceivably investigate running the Eigenvalue computations inside UMM firmware on the embedded DSP in each monitor module. This effectively splits the computations among 14 separate DSP processor cores (1 for each UMM) rather than trying to do this centrally in the SAM or a PC.

Q11: If you are returning only one pair of waveforms from each channel every 5 seconds, how are you determining which waveforms to send?

A11: Metrix uses an algorithm that quantifies the "interestingness" of a waveform; this is called the waveform's "i-value." The i-value is computed by looking at the relative change in static parameters corresponding to a given waveform as detailed below. The SETPOINT CMS manual (doc 1176125) does not use the term "i-value" and instead refers to this as the "Dynamic Collection % Change Threshold" where it is briefly discussed on pages 34-35. However, the terms can be used interchangeably and will be referred to as *i-value threshold* in this document due to its more compact form. The i-value is used as follows:

11.1. The waveform's static parameters are calculated, and include the following[1]:

11.1.1. Overall amplitude, which is the waveform's pk, pk-pk, or RMS AC amplitude
11.1.2. Gap voltage[2] (or bias voltage) which is the waveform's DC offset
11.1.3. The machine's rotative speed[3] which corresponds to the waveform's 1X period
11.1.4. The waveform's filtered 1X amplitude
11.1.5. The waveform's filtered 2X amplitude
11.1.6. The waveform's filtered nX amplitude (where n is user-configurable)
11.1.7. Up to 2 additional bandpass region amplitudes configured for the channel.

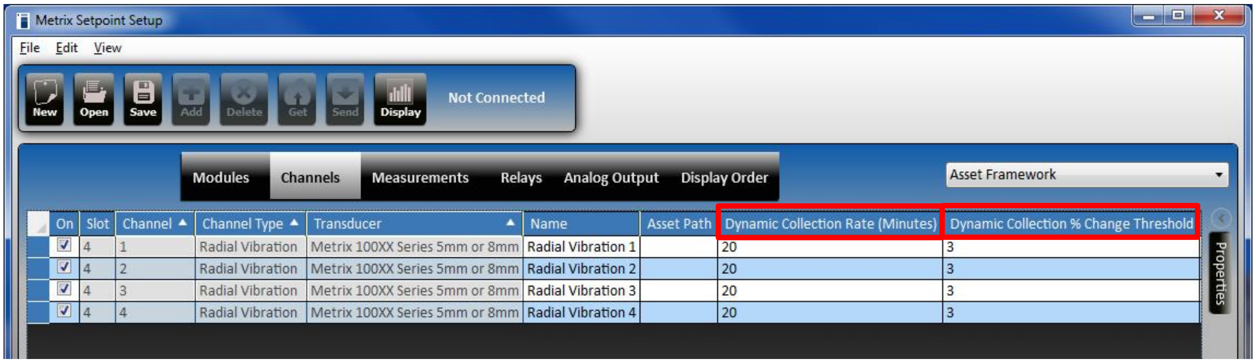
11.2. When a waveform sample is initially stored upon powering up or resetting the system, its characteristic static parameters become the baseline values, and the *i-value* is calculated. During any given 5 second interval thereafter, if any of the above static parameters change by more than 3% of the DANGER setpoint[4], a new waveform and baseline is stored. If the change is less than 3%, it is assumed that the baseline waveform adequately characterizes the current vibration and nothing is stored. Thus, the vibration is continuously sampled and compared to a relevant baseline, storing waveforms when the change detection warrants. This is depicted in Figure 4.

11.3. The change in i-value[5] required to force storage of a new waveform and corresponding baseline is set to 3% by default, but can be changed to any value between 0.1% and 1000% (see Figure 3 below). 3% has been found empirically to be adequately sensitive without flooding the system with "uninteresting" data.

11.4. If multiple waveforms for a channel within a given 5-second interval exceed the 3% i-value threshold, only the waveform with the largest change in i-value will be stored. This is an absolute value calculation; thus, if one waveform represented a +3.6% change in i-value while another waveform represented a -4.8% change in i-value, only the waveform corresponding to a -4.8% i-value change would be saved and a new associated baseline assigned.

NOTES:

1) The static parameters available for use in the i-value computation will depend on the channel type and configuration. See Table 1 in section 1.5.1 of the *SETPOINT MPS Operation and Maintenance Manual* (doc 1079330).
2) For purposes of i-value computation, the gap voltage has a full scale range of 48 volts; future versions of the configuration software may allow the user to adjust this value.
3) For purposes of i-value computation, the speed has a full scale range of 2000 rpm; future versions of the configuration software may allow the user to adjust this value.
4) If a DANGER setpoint has not been established for the static parameter in question, the full scale value is used instead.
5) The i-value associated with a channel is returned as a separate static variable type that can be trended and displayed. Metrix used these trends of i-values as part of its testing during various machine conditions (steady-state, startup, shutdown, malfunction, etc.) to empirically determine an optimal sensitivity.



On	Slot	Channel	Channel Type	Transducer	Name	Asset Path	Dynamic Collection Rate (Minutes)	Dynamic Collection % Change Threshold
☑	4	1	Radial Vibration	Metrix 100XX Series 5mm or 8mm	Radial Vibration 1		20	3
☑	4	2	Radial Vibration	Metrix 100XX Series 5mm or 8mm	Radial Vibration 2		20	3
☑	4	3	Radial Vibration	Metrix 100XX Series 5mm or 8mm	Radial Vibration 3		20	3
☑	4	4	Radial Vibration	Metrix 100XX Series 5mm or 8mm	Radial Vibration 4		20	3

Figure 3: Configuration screen showing user-adjustable settings for Dynamic Collection Rate (DCR) and Dynamic Collection % change threshold (i.e., i-value change threshold)

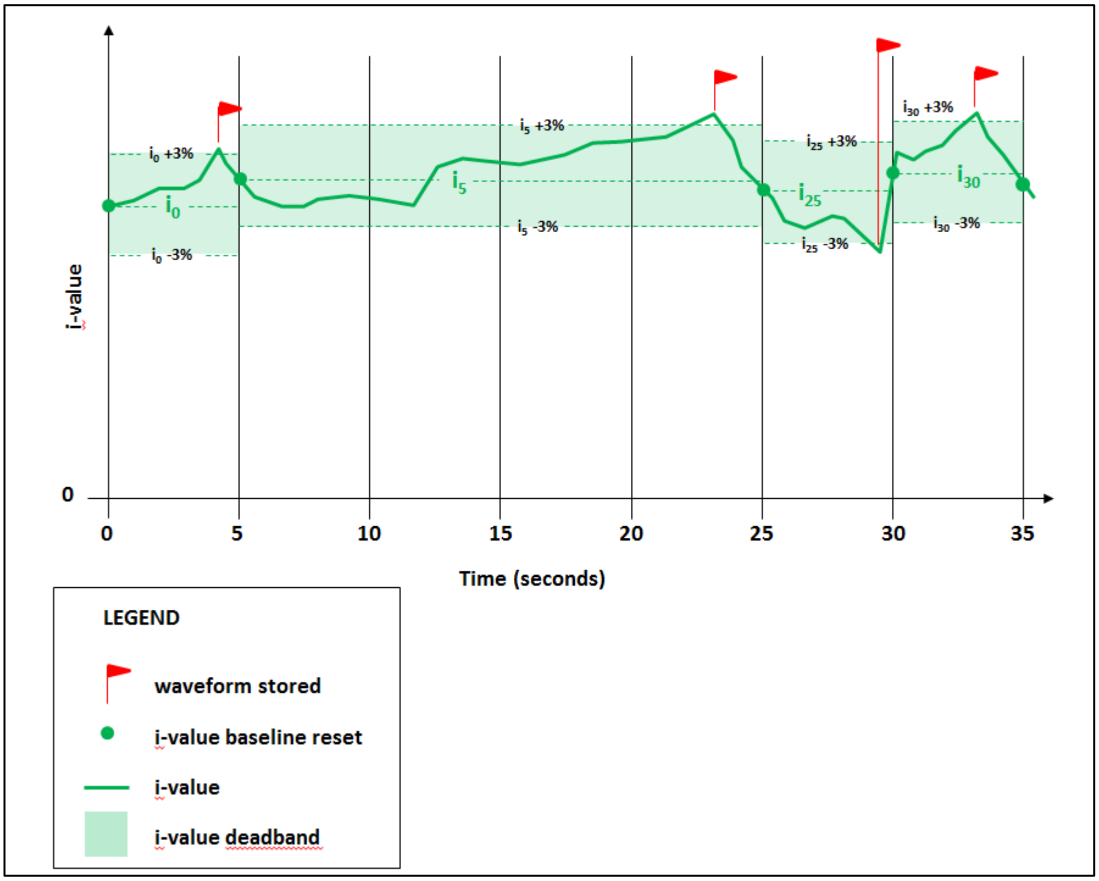


Figure 4: Example showing changing i-value and waveform storage triggers

11.5. If multiple waveforms for a channel within a given 5-second interval exceed the 3% i-value threshold, only the waveform with the largest change in i-value will be stored. This is an absolute value calculation; thus, if one waveform represented a +3.6% change in i-value while another waveform represented a -4.8% change in i-value, only the waveform corresponding to a -4.8% i-value change would be saved and a new associated baseline assigned.

11.6. The method used by Metrix has the added advantage of not relying on hardware alarms to trigger waveform collection. Hardware alarms are typically set somewhat conservatively to eliminate false or nuisance trips of critical machinery. The i-value relies indirectly on a DANGER setpoint if available, or simply on full scale value if a DANGER setpoint is not configured. The system is thus capable of collecting meaningful data even when vibration levels may fall well below established hardware alarm limits, without collecting data that reflects little or no change.

Q12: Can you collect data under changing speed conditions?

A12: Yes. Referring to 11.1.2 above, waveform storage is collected at so-called "Δrpm" intervals in addition to any other changes in the i-value that exceed the 3% "deadband". A change in rpm of more than 60 rpm (3% of 2000 rpm – see footnote 3, page 4) will trigger storage of a new waveform. Version 2.0 of SETPOINT CMS software is scheduled for release in December 2013 and will allow the user to display static data in polar format and frequency-domain dynamic data in waterfall and cascade formats. In fact, for users with installed CMS software, the data is already present in our database; we are merely providing the tools to display it. Bode plots – another format for viewing data versus machine speed – were released in 1.0 software.

Q13: What is the fastest rate at which you can currently collect and store data?

A13: As noted in Q7, this is currently 80 ms for all static data, and two waveforms (one synchronous and one asynchronous) from each channel every 5 seconds.

Q14: What happens if the i-value is not changing enough to trigger waveform capture? Can the system be configured to return a waveform anyway so that only an acceptable maximum amount of time is allowed to elapse between waveforms?

A14: Yes. This is referred to in the CMS manual as the "Dynamic Collection Rate" (DCR) and is discussed on pages 33-34 of the manual. The DCR is a user-configurable interval (see configuration screen in Figure 3) that ensures the maximum amount of time that will be allowed to elapse between waveforms being stored, even if the i-value has not changed appreciably (i.e., 3% or more). The DCR can best be thought of as a countdown timer. Assume the user has configured the DCR for 2 hours. Further assume that the machine runs for 2 hours at steady-state conditions and therefore incurs only very small fluctuations in vibration, well below the 3% i-value threshold. Each and every waveform during this 2-hour interval is analyzed and the waveform with the largest change in i-value from the baseline is saved in a constantly-updating buffer. Once 2 hours have elapsed, the buffer is published, even if it does not represent at least a 3% change from the baseline. Under worst-case conditions, the maximum elapsed time between two waveforms will therefore be 2X the DCR (assumes first waveform collected in DCR interval #1 has largest i-value and last waveform collected in DCR interval #2 has largest i-value).

For a depiction of the DCR, refer to Figure 5 on the following page. In the figure, we assume that the DCR has been set for 10 minutes (600 seconds). The DCR countdown starts immediately following the last waveform capture and would work as follows:

- In the time interval t=0 to t=5, the largest excursion of the i-value from the baseline value occurs at t=2. Because this excursion exceeds 3%, a waveform is stored. At the end of this interval (t=5), the DCR timer begins counting down.

- During the interval t=5 to t=605, the change in i-value never exceeds 3%. The largest excursion occurs at t=8 and is approximately -2%. Because no waveform has been stored during this 10-minute period, the DCR countdown timer runs out and it selects the waveform with the largest change in i-value (t=8, i-value= -2%). The baseline is reset to the value at t=605, as depicted by the green dot.

- The DCR timer resets and begins counting down at t=605. During the next 10-minute interval (t=605 to t=1205), the i-value again does not change by 3%, so no waveform is saved by virtue of excessive i-value change. Instead, the DCR timer runs out after 10 minutes at t=1205 and the system selects the waveform with the largest excursion from the baseline value during this interval. This occurs at t=1202 and represents an i-value change of approximately +2%. Thus, waveforms are saved at t=8 and at t=1202. The elapsed time between waveforms stored in this example is approximately twice the 10-minute DCR, or 19 min. 54 sec.

- The maximum change in i-value that could occur under worst-case conditions would happen when the i-value excursion was -2.99% at t=8, the new baseline was reset to i-value = +2.99% at t=605, and the i-value continued rising another +2.99% above the baseline value at t=1202. This would give a *change* in i-value (Δi) of approximately 9% between the waveform stored at t=8 and that stored at t=1202.

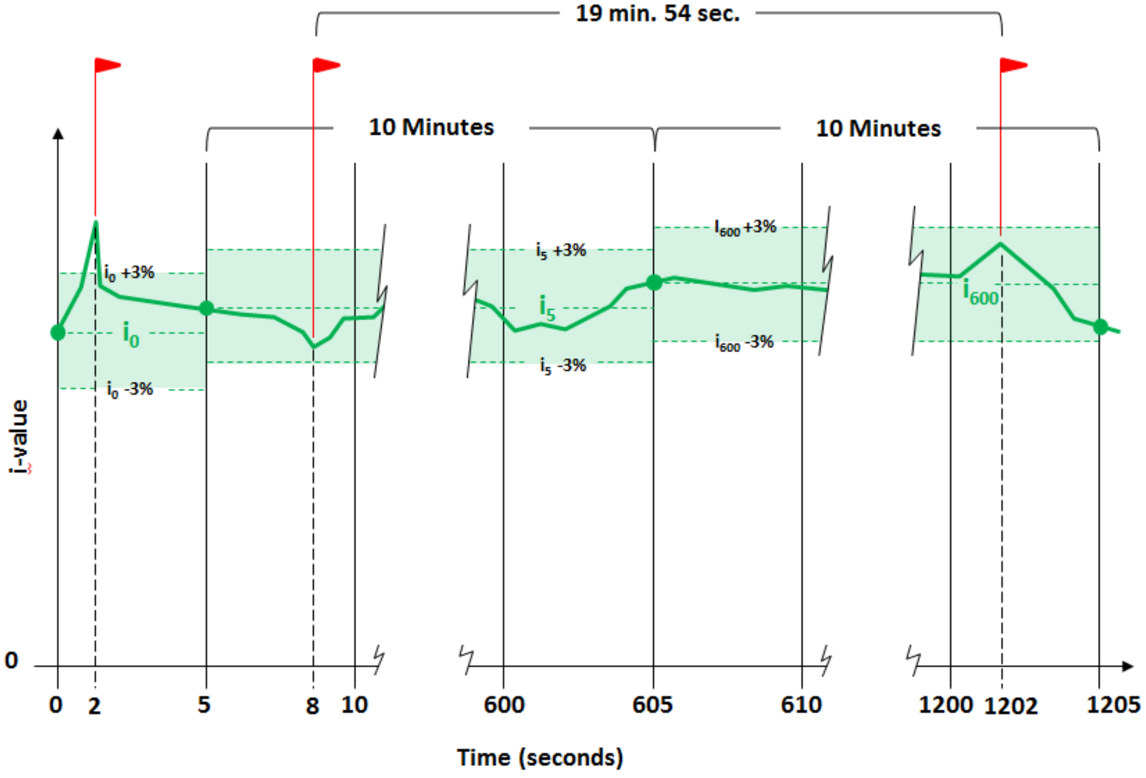


Figure 5: Dynamic Collection Rate (DCR) example

Q15: What about the time stamps on data?

A15: Although the fastest rate at which data can be saved from the system is currently one synchronous and one asynchronous waveform during a given 5-second interval, any two sync or async waveforms saved could actually be adjacent to one another. Assume that a machine is running at 10,000 rpm with a synchronous sample rate of 128 samples per rev and a sample length of 2048 samples. A waveform will thus consist of 2048 samples and encompass 16 shaft revolutions (2048 samples / 128 samples per rev). The time stamps on successive waveforms would be 6 ms apart (16 revs at 10,000 rpm = 6 ms). If the waveform with the largest i-value during interval #1 is the last waveform, and the waveform with the largest i-value during interval #2 is the first waveform, they would be successive and would thus reflect time stamps that are only separated by 6 ms. In contrast, the worst-case scenario would occur when the first waveform of interval #1 and the last waveform of interval #2 were the largest i-values. In this case, the time stamps on the saved waveforms would be approximately 10 seconds apart.

Q16: What is the protocol used at the DAC (Data Acquisition Computer) port on the SAM?

Q16: The Application Programming Interface (API) used in the SETPOINT rack is available to any customer wishing to develop their own database / display environment. The API is not documented in the product manual; however, it will be provided upon request in a separate document. When used with the OSIsoft™ PI System, a special application (*SETPOINT-to-PI Adapter*) is used to convert the rack's native API to the OSIsoft environment.