11-19-2018

RS485 Integrators Guide

VTB-SENSOR MODBUS DEFINITIONS

MACHINE SAVER INC| 9788 Clarewood Dr. STE 302 Houston, TX 77036

Table of Contents

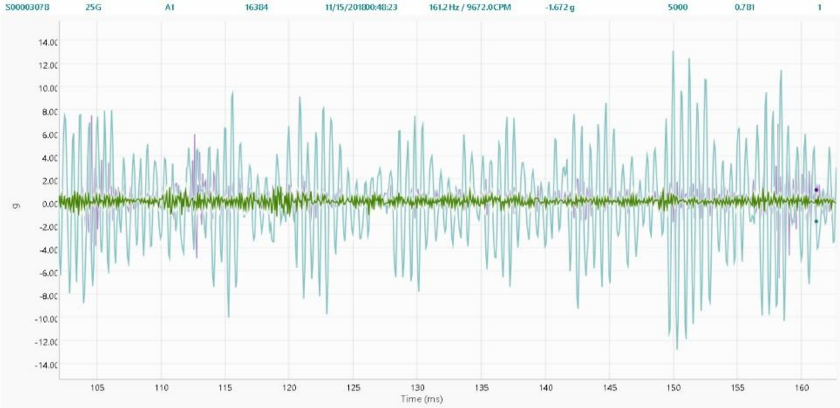
Vocabulary:

Overall Vibration Data – processed data which can be used for machine protection. Trending this data on good machines and making decisions based on the changes in this data is how most operators use the VTB Sensor.

Dynamic Data Clip – Raw ADC Count values typically used for machine condition analysis purposes. When prompted by a Modbus Master, the sensor captures this data and stores it in an internal buffer on the sensor. Once stored, the data can be retrieved from the sensor by reading a special set of Modbus Registers. Those values can then converted to amplitude. Once converted, they can be displayed as a Time Waveform (amplitude/time) and/or a Fast Fourier Transform can be performed on the converted data to produce a Spectrum (amplitude/frequency). Vibration Analysts or Machine Learning Algorithms can parse this data to detect specific fault conditions (such as misalignment, out of balance, bearing faults or gear-mesh faults) of machinery based on the machine's geometry and running speed.

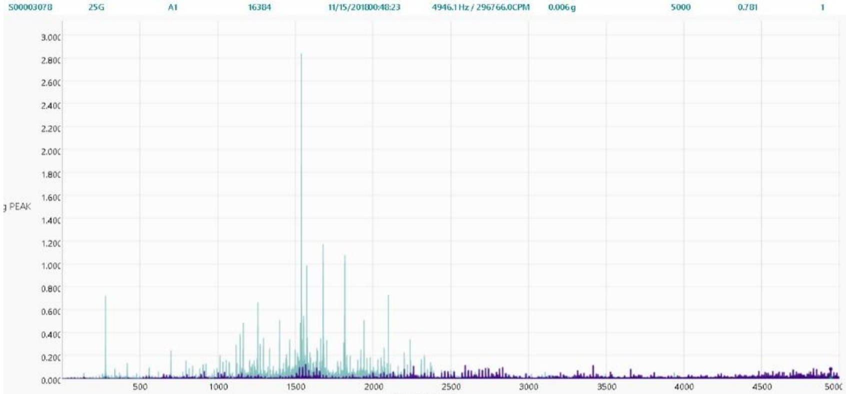
Time Waveform – a charted display of a machine's dynamic data in the time domain.

Serial	Type	Axis	Samples	Date/Time	Frequency	Value	Freq Max	FFT Res	Avg
S00003078	25G	A3	16384	11/15/201800:51:39	161.2 Hz / 9672.0CPM	0.096 g	5000	0.781	1
S00003078	25G	A2	16384	11/15/201800:50:12	161.2 Hz / 9672.0CPM	1.051 g	5000	0.781	1
S00003078	25G	A1	16384	11/15/201800:48:23	161.2 Hz / 9672.0CPM	-1.672 g	5000	0.781	1



Spectrum – a charted display of a machine's dynamic data in the frequency domain. This can be thought of as an EKG of the machine's internal components which a vibration analyst or machine learning algorithms could parse to diagnose the underlying causes of excess vibration on a machine.

Serial	Type	Axis	Samples	Date/Time	Frequency	Value	Freq Max	FFT Res	Avg
S00003078	25G	A3	16384	11/15/201800:51:39	4946.1 Hz / 296766.0CPM	0.007 g	5000	0.781	1
S00003078	25G	A2	16384	11/15/201800:50:12	4946.1 Hz / 296766.0CPM	0.084 g	5000	0.781	1
S00003078	25G	A1	16384	11/15/201800:48:23	4946.1 Hz / 296766.0CPM	0.006 g	5000	0.781	1



Fast Fourier Transform – is an algorithm that samples a signal over a period of time (or space) and divides it into its frequency components. This is how the data is changed from the time domain (Time Waveform) into the frequency domain (Spectrum).

Machine Saver's Web Portal – (SAAS) a remote server which takes overall vibration data, temperature, alarm data and dynamic vibration data and presents it in a useful way for operators and analysts.

Vibration Alarms – Four typical methods for alarms based on VTB Sensor data.

1st Machine Saver's sensors in conjunction with a cellular gateway and data logger which monitors overall vibration and temperature and sends text message and email alerts to operators and analysts when there are significant changes in the data.

2nd Machine Saver's Internal Sensor Alarm Channels. These are alarms set inside the VTB Sensor by writing to certain configuration registers. Once configured, other readable registers get filled with data related to the levels set internally. These can be monitored and logged by an external system.

3rd Overall vibration and temperature data can be polled continuously and alarms can be handled by an external Modbus Master or PLC. Often there is a warning level and a shutdown level based on this overall data. This data can be monitored and logged by an external system.

4th Dynamic Data Clips taken at intervals and converted to acceleration and then changed to the frequency domain. Once in the frequency domain, a system could look at the fault frequencies specific to the machinery that is being monitored. When amplitudes at known fault frequencies increase in the spectra, maintenance should be notified to look at the specific component which shows increased amplitudes. Traditionally, this work was done by vibration analysts which would walk around with a portable data collector to gather data on a weekly or monthly basis and then spend time reviewing the data they collected manually.

Trip Delay – The amount of time after a signal has crossed an alarm threshold before the sensor registers the exceeded alarm state as an actual alarm.

Hysteresis/Deadband – The range on top of an alarm threshold that a signal must pass in order for a second alarm to be registered.

RTU Number - Remote Terminal Unit number, also known as the "Slave ID", is the identifying number that a Modbus Master addresses to communicate with a VTB-Sensor.

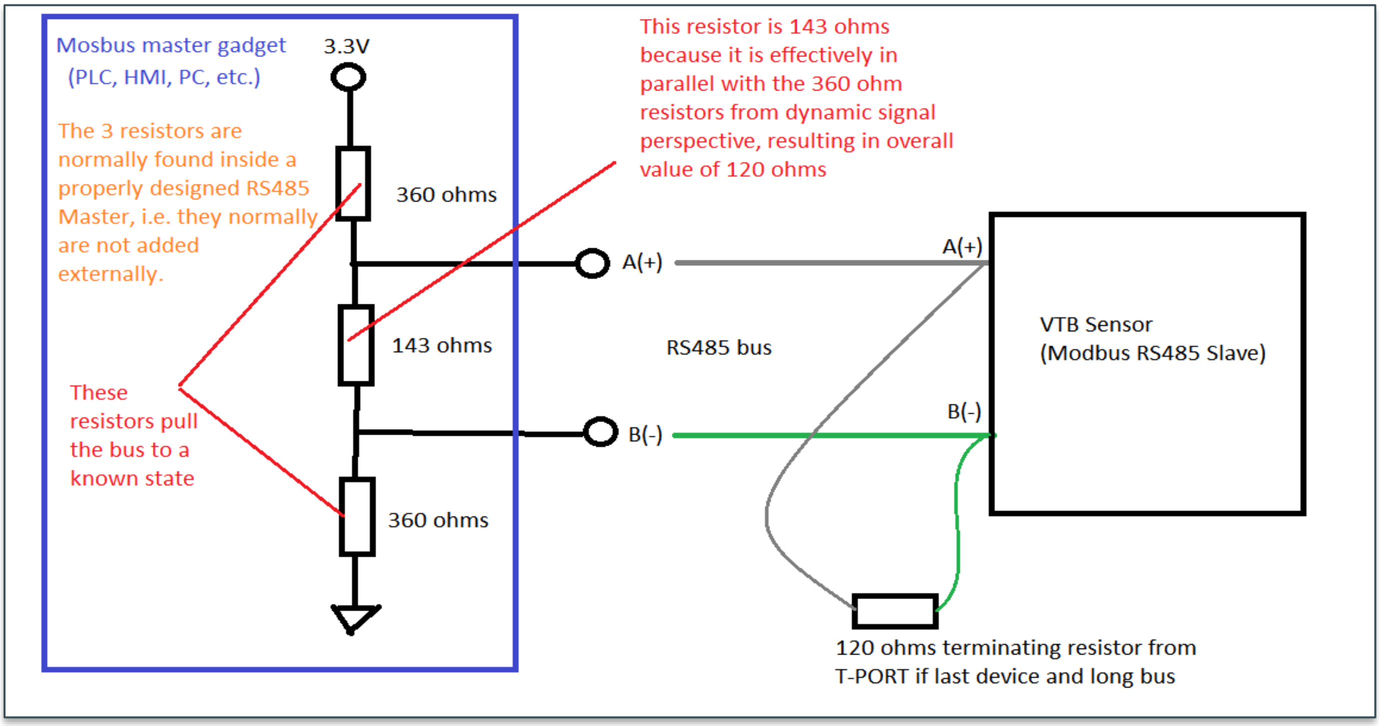
Sensor Mode – Since the VTB-Sensor is a digital device it can be programmed for different internal processing algorithms which provides the user with different information. The two which will be illuminated in this document are vibration (standard) and impact mode (mechanical looseness monitoring).

Sensor Type – The VTB Sensor comes in several different "flavors" depending on the application. Each internal accelerometer has a frequency and an amplitude range.

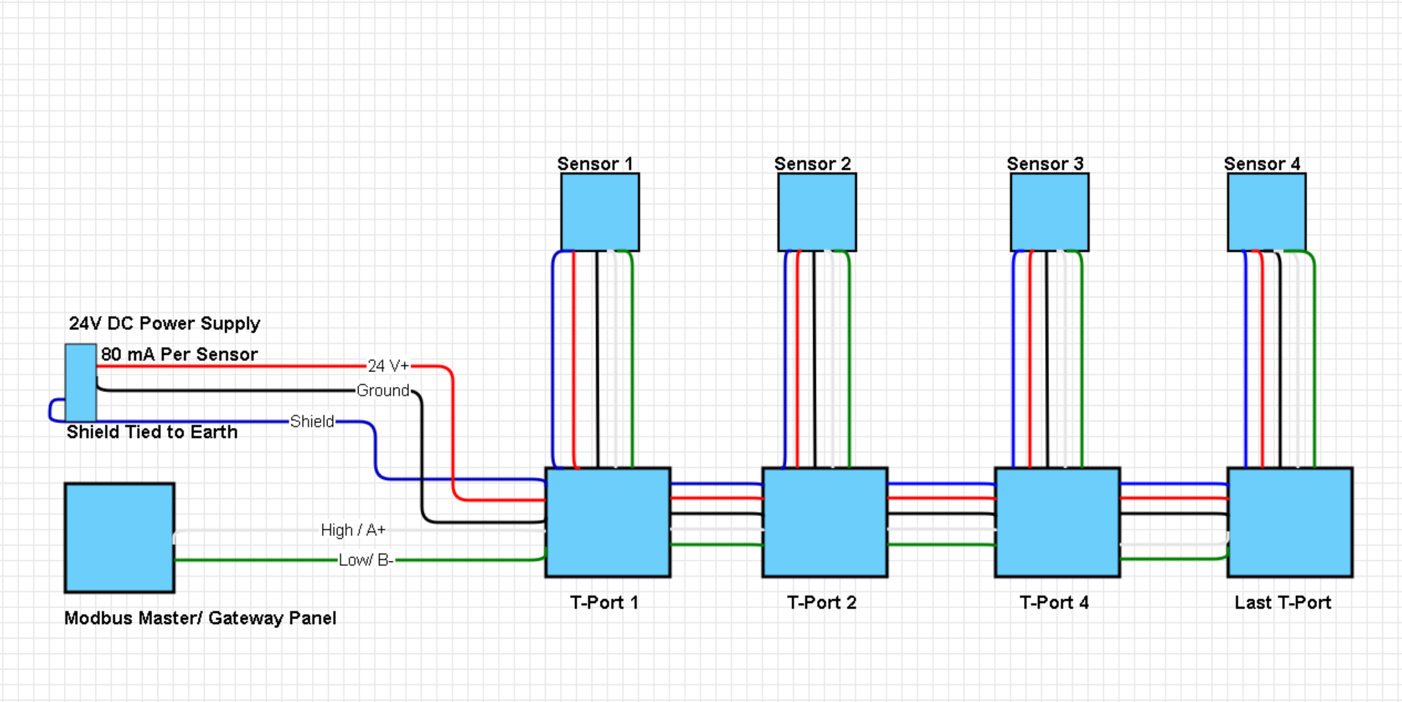
Sensor Amplitude Range	Corresponding Frequency Range
2G	1000 Hertz
4G	1000 Hertz
10G	1000 Hertz
25G	5000 Hertz
100G	5000 Hertz

Communicating with the VTB Sensor

Modbus Master Electrical Requirements



Daisy Chain Wire Diagram



Connect Power Wires to the Sensor

Apply power to the sensor using 24-36V DC.

The 18 Gauge Red Conductor is V+

The 18 Gauge Black Conductor is Ground

The 24 Gauge Grey Conductor is Shield (to be connected to Earth Ground on the panel/Master side of the bus)

Connect Communication Wires to the Sensor

The 22 Gauge Green Conductor is LOW / B-

The 22 Gauge White Conductor is HIGH / A+

Make sure that the jumper which enables the 120 Ohm resistor is installed on the Last T-Port (ONLY)

Finding your Sensor's RTU Number

When shipped from the Machine Saver factory, the sensor's RTU number is set as the last 2 digits of its serial number. The Serial Number can be found on the side of the VTB-Sensor white labeling:

This can also be found by polling the sensor at Modbus Register Address 40027-40028.

RTU Numbers must be unique on a single bus line (Do not duplicate/repeat an RTU number on the same bus line). RTU Numbers must be integers between 2-245.Traditionally, 0,1 and 255 are reserved for the Modbus Master (Do not set a VTB-Sensor to any of these reserved numbers).

Default Communication Settings

Baud Rate: 115,200

Stop Bits: one

Parity: none

Handshakes: None

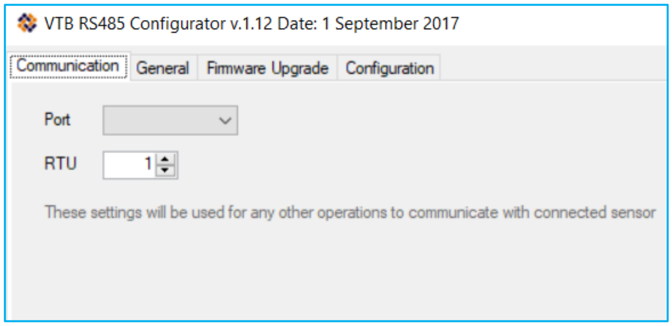
Data Bits: 8

(Optional) Change the RTU Number of a VTB-Sensor

Using RS485 Configurator (Windows-based Utility)

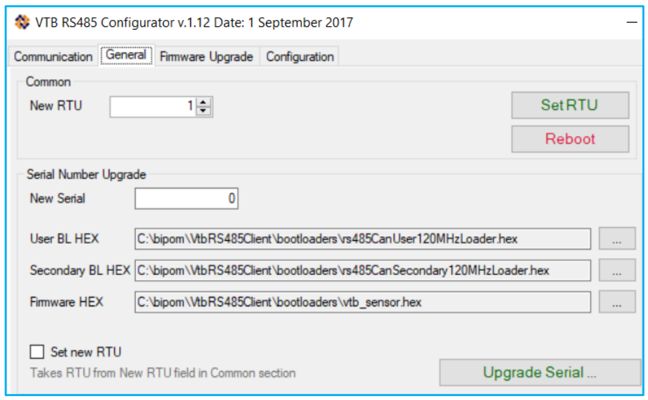
Set up the RS485 to USB Converter Cable [Reference Document]

Open and Install VTB RS485 Configurator from the USB



Select the COM Port associated with the RS485 to USB Converter Cable

Enter the current sensor RTU



Select the “General” tab

Enter the new RTU Number next to “New RTU”

Select “Set RTU” (The sensor will reboot, after the sensor reboots the RTU Number will be updated)

Using Modbus Commands

Reference Document [ChangingRTUNumberUsingModbus.docx]

1. Unlock Configuration Registers
   A. Write 45555 to System Control Register 40002
2. Assign new RTU by writing to the related Configuration Register
   A. Write new RTU number to Configuration Register 40368
3. Save to NVM and lock all Configuration Registers
   A. Write 45556 to System Control Register 40002
4. Check the status
   A. Read System Status Register 40003 to check the remote command is executed without problems (1-done).
5. Restart the sensor (After the sensor reboots the RTU Number will be updated)
   A. Write 1 to System Control Register 40002

(Optional) RS485 to USB Converter Cable Installation and Setup

Reference Document [RS485toUSBCableSetup]

Connect the sensor cable to the T-Port

Connect the T-Port to a 24VDC Power supply

Connect the RS485 to USB Converter Cable to the T-Port's White and Green Terminal

Connect the RS485 to USB Converter Cable to a computer's USB Port

Run the program file () to update the RS485 to USB Converter Cable Driver

Reading Overall Vibration from the Sensor

Each VTB Sensor has 6 internal accelerometers. 2 on each axis. The sensor can tell internally when the useful amplitude of an internal accelerometer is exceeded. When the useful amplitude is exceeded for the lower amplitude sensor registers for Overall Vibration with be switched to the higher amplitude accelerometer.

Example: If the 4g sensor of a 2g/4g sensor begins reading anything above 2 gs, the value put into the overall vibration register will be the reading from the 4g sensor.

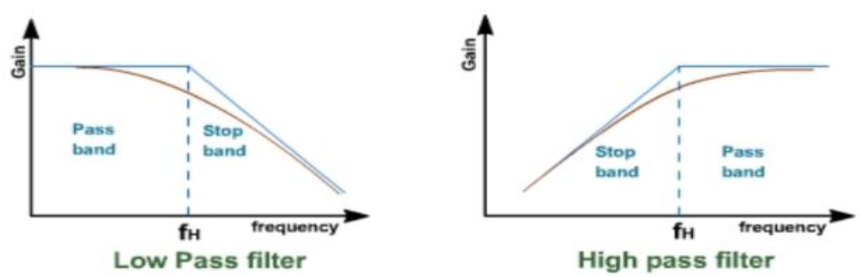
You also have the option to read the vibration information directly from one of the internal accelerometers.

If you read the board type (Register 4XXXX) and it is 25G2G.

The 25G sensor is Internal Sensor 1

The 2G sensor is Internal Sensor 2

Filtering Out Frequencies from Overall Vibration



Filter Registers

Register	Name	Description	Type	Unit
40376	Low Pass Filter	Frequency filter for overall vibration values multiplied by 10 (all frequencies below this value will be included in overall vibration)	16-bit Unsigned Integer	Hertz
40377	High Pass Filter	Frequency filter for overall vibration values multiplied by 10 (all frequencies above this value will be included in overall vibration)	16-bit Unsigned Integer	Hertz

Internal Sensor Alarm Setups

Foreword about Alarm Channels

Within the VTB-Sensor there are four alarm channels which are designated 1-4. When the VTB-Sensor only had a vibration mode, these four channels would be setup to so that each Axis could have an alarm based on either overall acceleration or overall velocity readings. The 4th channel could be configured for a temperature alarm or configured to alarm on one of the axis using the unused measurand (such as velocity when the other channels were already set to alarm on acceleration or vice versa). As the sensor grew to include new algorithms for special applications, the same four alarm channels could be used for a host of other applications such as Impact Monitoring (for detecting mechanical looseness on reciprocating compressors) and Detonation Monitoring (for detecting explosions inside the power cylinders of large natural gas engines).

Upon shipment of the sensor these alarm channels are not configured and in order to use them must be configured using either Modbus Commands or using the windows-based utility called RS485 Configurator both of these processes are described below.

Configuring Alarm Channels using Modbus Registers:

Alarm Type, Axis, Low Limit, High Limit, High High Limit, Hysteresis must be configured for each desired alarm channel.

Register	Name	Description	Type	Unit
Alarm #1 Config				
40314	Type	This register contains information about the type of alarm for this alarm channel. Alarm1 Type: 0 – Not Configured 1 – Velocity 2 – Acceleration 3 – Displacement (not supported for now) 4– Temperature 5 – Impact Alert 6 – Impact Danger 7 – Detonation These are NOT bit numbers. Write the actual value shown to the register to achieve the desired effect.		
40315(High Word)/ 40316(Low Word)	Low Limit	High 16 bits of alarm 1 low level/ Low 16 bits of alarm 1 low level		g RMS, ips RMS, mils PK-PK, C
40321(High Word)/ 40322(Low Word)	Hysteresis	High 16 bits of alarm 1 hysteresis/ Low 16 bits of alarm 1 hysteresis		g RMS, ips RMS,

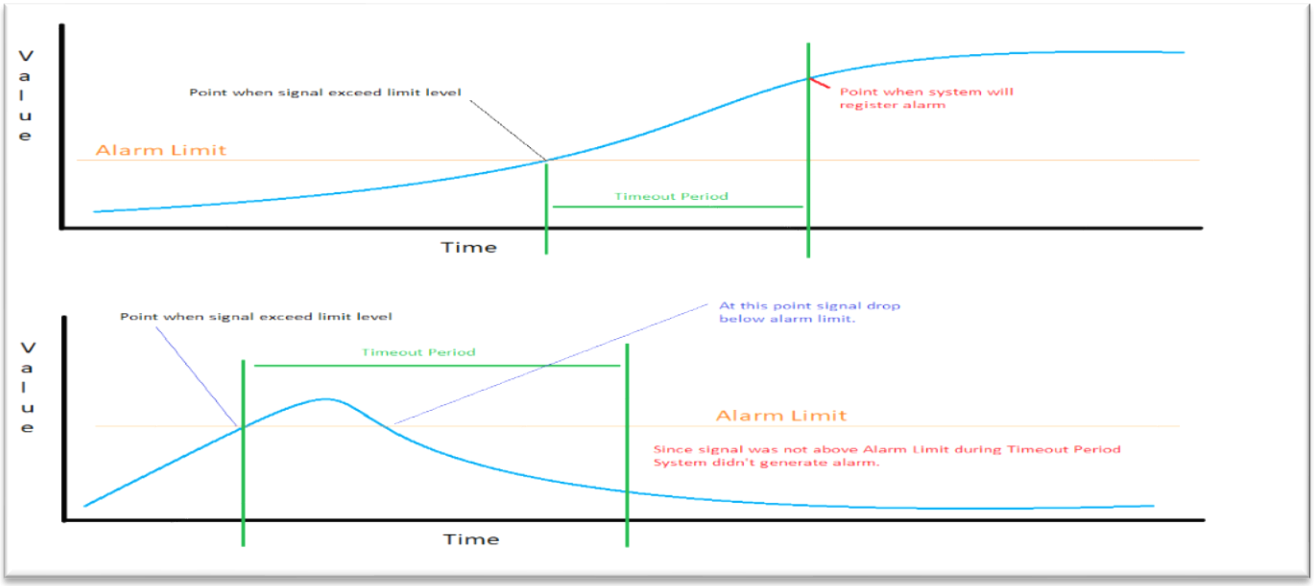
				mils PK-PK, C
40319(High Word)/ 40320(Low Word)	HighHIgh Limit	High 16 bits of alarm 1 high high level/ Low 16 bits of alarm 1 high high level		g RMS, ips RMS, mils PK-PK, C
40317(High Word)/ 40318(Low Word)	High Limit	High 16 bits of alarm 1 high level/ Low 16 bits of alarm 1 high level		g RMS, ips RMS, mils PK-PK, C
40313	Axis	This register contains information about the axis for this alarm channel. Alarm2 Axis: 0 – Not Configured 1 – Axis 1 2 – Axis 2 3 – Axis 3 4 – Temperature 5 – Not Used 6 – Not Used 7 – Detonation Axis 1 8 – Detonation Axis 2 9 – Detonation Axis 3 10 – Impact Threshold Axis 1 11 – Impact Threshold Axis 2 12 – Impact Threshold Axis 3 13 – Impact Count Axis 1 14 – Impact Count Axis 2 15 – Impact Count Axis 3 These are NOT bit numbers. Write the actual value shown to the register to achieve the desired effect.		
Alarm #2 Config				
40324	Type	This register contains information about the type of alarm for this alarm channel. Alarm1 Type: 0 – Not Configured 1 – Velocity 2 – Acceleration 3 – Displacement (not supported for now) 4– Temperature 5 – Impact Alert 6 – Impact Danger 7 – Detonation These are NOT bit numbers. Write the actual value shown to the register to achieve the desired effect.		
40325(High Word)/ 40326(Low Word)	Low Limit	High 16 bits of alarm 1 low level/ Low 16 bits of alarm 1 low level		g RMS, ips RMS, mils PK-PK, C

40331(High Word)/ 40332(Low Word)	Hysteresis	High 16 bits of alarm 1 hysteresis/ Low 16 bits of alarm 1 hysteresis		g RMS, ips RMS, mils PK-PK, C
40329(High Word)/ 40330(Low Word)	HighHIgh Limit	High 16 bits of alarm 1 high high level/ Low 16 bits of alarm 1 high high level		g RMS, ips RMS, mils PK-PK, C
40327(High Word)/ 40328(Low Word)	High Limit	High 16 bits of alarm 1 high level/ Low 16 bits of alarm 1 high level		g RMS, ips RMS, mils PK-PK, C
40323	Axis	This register contains information about the axis for this alarm channel. Alarm2 Axis: 0 – Not Configured 1 – Axis 1 2 – Axis 2 3 – Axis 3 4 – Temperature 5 – Not Used 6 – Not Used 7 – Detonation Axis 1 8 – Detonation Axis 2 9 – Detonation Axis 3 10 – Impact Threshold Axis 1 11 – Impact Threshold Axis 2 12 – Impact Threshold Axis 3 13 – Impact Count Axis 1 14 – Impact Count Axis 2 15 – Impact Count Axis 3 These are NOT bit numbers. Write the actual value shown to the register to achieve the desired effect.		
Alarm #3 Config				
40334	Type	This register contains information about the type of alarm for this alarm channel. Alarm1 Type: 0 – Not Configured 1 – Velocity 2 – Acceleration 3 – Displacement (not supported for now) 4– Temperature 5 – Impact Alert 6 – Impact Danger 7 – Detonation These are NOT bit numbers. Write the actual value shown to the register to achieve the desired effect.		
40335(High Word)/ 40336(Low Word)	Low Limit	High 16 bits of alarm 1 low level/ Low 16 bits of alarm 1 low level		g RMS, ips RMS, mils

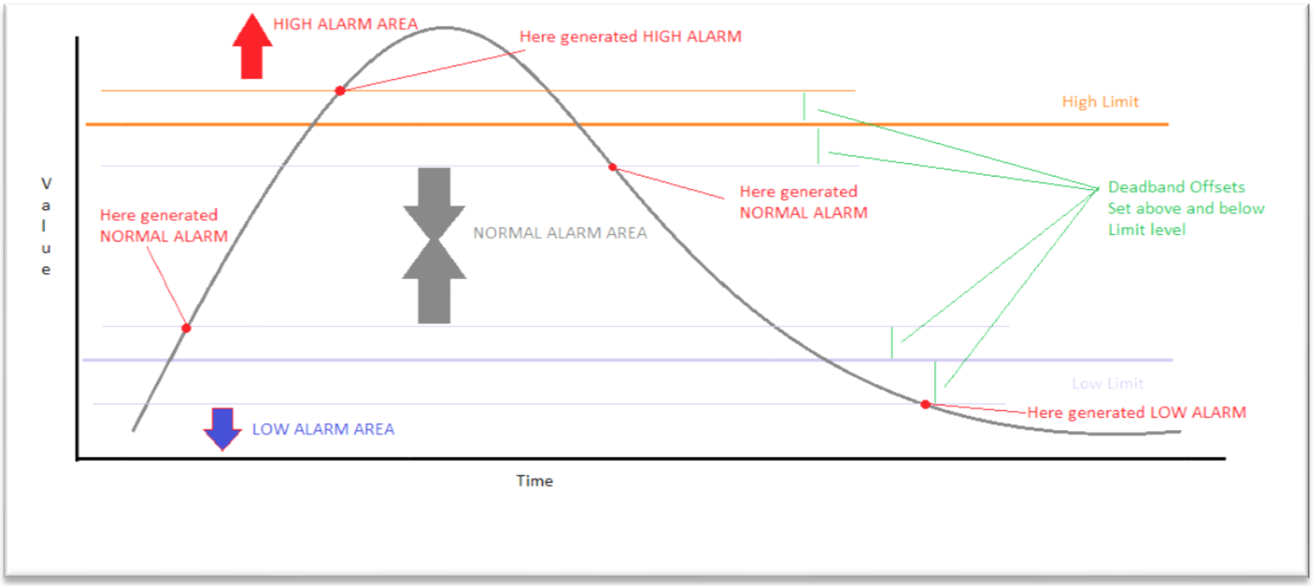
				PK-PK, C
40341(High Word)/ 40342(Low Word)	Hysteresis	High 16 bits of alarm 1 hysteresis/ Low 16 bits of alarm 1 hysteresis		g RMS, ips RMS, mils PK-PK, C
40339(High Word)/ 40340(Low Word)	HighHIgh Limit	High 16 bits of alarm 1 high high level/ Low 16 bits of alarm 1 high high level		g RMS, ips RMS, mils PK-PK, C
40337(High Word)/ 40338(Low Word)	High Limit	High 16 bits of alarm 1 high level/ Low 16 bits of alarm 1 high level		g RMS, ips RMS, mils PK-PK, C
40333	Axis	This register contains information about the axis for this alarm channel. Alarm2 Axis: 0 – Not Configured 1 – Axis 1 2 – Axis 2 3 – Axis 3 4 – Temperature 5 – Not Used 6 – Not Used 7 – Detonation Axis 1 8 – Detonation Axis 2 9 – Detonation Axis 3 10 – Impact Threshold Axis 1 11 – Impact Threshold Axis 2 12 – Impact Threshold Axis 3 13 – Impact Count Axis 1 14 – Impact Count Axis 2 15 – Impact Count Axis 3 These are NOT bit numbers. Write the actual value shown to the register to achieve the desired effect.		
Alarm #4 Config				
40344	Type	This register contains information about the type of alarm for this alarm channel. Alarm1 Type: 0 – Not Configured 1 – Velocity 2 – Acceleration 3 – Displacement (not supported for now) 4– Temperature 5 – Impact Alert 6 – Impact Danger 7 – Detonation These are NOT bit numbers. Write the actual value shown to the register to achieve the desired effect.		

40345(High Word)/ 40346(Low Word)	Low Limit	High 16 bits of alarm 1 low level/ Low 16 bits of alarm 1 low level		g RMS, ips RMS, mils PK-PK, C
40351(High Word)/ 40352(Low Word)	Hysteresis	High 16 bits of alarm 1 hysteresis/ Low 16 bits of alarm 1 hysteresis		g RMS, ips RMS, mils PK-PK, C
40349(High Word)/ 40350(Low Word)	HighHIgh Limit	High 16 bits of alarm 1 high high level/ Low 16 bits of alarm 1 high high level		g RMS, ips RMS, mils PK-PK, C
40347(High Word)/ 40348(Low Word)	High Limit	High 16 bits of alarm 1 high level/ Low 16 bits of alarm 1 high level		g RMS, ips RMS, mils PK-PK, C
40343	Axis	This register contains information about the axis for this alarm channel. Alarm2 Axis: 0 – Not Configured 1 – Axis 1 2 – Axis 2 3 – Axis 3 4 – Temperature 5 – Not Used 6 – Not Used 7 – Detonation Axis 1 8 – Detonation Axis 2 9 – Detonation Axis 3 10 – Impact Threshold Axis 1 11 – Impact Threshold Axis 2 12 – Impact Threshold Axis 3 13 – Impact Count Axis 1 14 – Impact Count Axis 2 15 – Impact Count Axis 3 These are NOT bit numbers. Write the actual value shown to the register to achieve the desired effect.		

Setting up Alarm Timeout/Trip Delay



Setting up Alarm Hysteresis/Dead-band



Reading Alarm Information from the Sensor

How to read current or latched alarm states from the VTB Sensor:

Register	Name	Description	Type	Unit
40312	Trip Delay	Alarm Trip Delay Applies across all 4 Alarm Channels	16-bit Unsigned Integer	Seconds
40239	ALARM CONTROL	A Special Register which must be written to in order to change alarm settings.	16-bit Unsigned Integer	
40227	Alarm #1 State	Current state of Alarm Channel #1	16-bit Unsigned Integer	
40240	Alarm #1 NORMAL Value		32-bit Floating Point	
40231	Alarm #1 Highest Latched Value		32-bit Floating Point	
40256	Alarm #1 HI Value		32-bit Floating Point	
40264	Alarm #1 HI HI Value		32-bit Floating Point	
40228	Alarm #2 State	Current state of Alarm Channel #2	16-bit Unsigned Integer	
40242	Alarm #2 NORMAL Value		32-bit Floating Point	
40233	Alarm #2 Highest Latched Value		32-bit Floating Point	
40258	Alarm #2 HI Value		32-bit Floating Point	
40266	Alarm #2 HI HI Value		32-bit Floating Point	
40229	Alarm #3 State	Current state of Alarm Channel #3	16-bit Unsigned Integer	
40244	Alarm #3 NORMAL Value		32-bit Floating Point	
40235	Alarm #3 Highest Latched Value		32-bit Floating Point	
40260	Alarm #3 Hi Value		32-bit Floating Point	
40268	Alarm #3 HI HI Value		32-bit Floating Point	
40230	Alarm #4 State	Current state of Alarm Channel #4	16-bit Unsigned Integer	
40246	Alarm #4 NORMAL Value		32-bit Floating Point	
40237	Alarm #4 Highest value Latched		32-bit Floating Point	
40262	Alarm #4 HI Value		32-bit Floating Point	
40270	Alarm #4 HI HI Value		32-bit Floating Point	

External Alarm Suggestions

Programming External Alarms based on Overall Vibration

Overall Acceleration (g RMS)

A useful measurand for higher frequency signals.

Overall Velocity (ips RMS)

A useful measurand across a broad range of frequency signals.

Temperature (Celsius)

Another good indication of issue is increases in temperature.

Dynamic Data Capability

What can the data tell you?

Vibration Analysts have used both timewave form (amplitude vs time) and spectral (amplitude vs frequency) information for years to successfully identify fault conditions of rotating and reciprocating machinery before the failures become catastrophic.

How many samples do you need? (Nyquist Theorem)

The Nyquist Theorem states that in order to resolve a sine wave you must sample at at least 2X the highest frequency that you want to see. In practice, the actual sample rate should be 2.7X the highest frequency that you want to resolve.

Reference document [VTB Sensor Reading Dynamic Data Clips for Analysis Updated 11-7-2018]

EX. If you want to resolve a 2350 Hertz sine wave you would need 6345 Samples per 1 Second Period.

So if you wanted dynamic data clip looking at 2000ms (Time Capture Register: 40036) and wanted to resolve the same 2350 Hertz frequency you would need to request at least 12690 Samples (Number of Samples Register: 40038) from the sensor.

Conceptual Video Links

An Introduction for Machine Vibration Analysis (first 16 minutes recommended):

https://www.youtube.com/watch?v=Vj1xmze3GlE

Understanding the Time Waveform (Composed of Individual Sine Waves):

https://www.youtube.com/watch?v=r18Gi8lSkfM&t=561s

Collecting Dynamic Data Clips

Reference Document:

[VTB Sensor Reading Dynamic Data Clips for Analysis Updated 11-7-2018.docx]

Software Components & Items to be Included on USB (Optional)

Modbus Client

.RTU Project File for Modbus Client

RS485 Configurator

RS485 to USB Driver

RS485 Installation Procedure

[VTB Sensor Reading Dynamic Data Clips for Analysis Updated 11-7-2018.docx]

[Dynamic Data Spreadsheet (Includes Conversion from Raw Data to Gs]

Overall Vibration Registers

Register	Name	Description	Type	Unit
40173(High Word)-40174(Low Word)	Axis 1 Acceleration	Selects between internal sensor 1 and internal sensor 2 appropriate acceleration value	32-bit Floating Point	g RMS
40175(High Word)-40176(Low Word)	Axis 2 Acceleration	Selects between internal sensor 1 and internal sensor 2 appropriate acceleration value	32-bit Floating Point	g RMS
40177(High Word)-40178(Low Word)	Axis 3 Acceleration	Selects between internal sensor 1 and internal sensor 2 appropriate acceleration value	32-bit Floating Point	g RMS
40179(High Word)-40180(Low Word)	Axis 1 Velocity	Selects between internal sensor 1 and internal sensor 2 appropriate acceleration value	32-bit Floating Point	ips RMS
40181(High Word)-40182(Low Word)	Axis 2 Velocity	Selects between internal sensor 1 and internal sensor 2 appropriate acceleration value	32-bit Floating Point	ips RMS
40183(High Word)-40184(Low Word)	Axis 3 Velocity	Selects between internal sensor 1 and internal sensor 2 appropriate acceleration value	32-bit Floating Point	ips RMS

Other Useful Registers

Register	Name	Description	Type	Unit
40028	Sensor Serial Number	The serial number of the sensor	16-bit Unsigned Integer	N/A
40005	System State 2 (Internal Error register)	If the sensor has a problem with an internal process this register will return an error code greater than 0 **Reference Appendix A for Error Codes**	16-bit Unsigned Integer	N/A
40015	Mode register	1 = Vibration 2 = Impact	16-bit Unsigned Integer	N/A
40032	Temperature	Temperature in Celsius multiplied by 10. EX. Sensor reads 253. The temperature is 25.3˚C.	16-bit Signed Integer	˚C

Modbus Examples

Float Format for MODBUS Floating-Point Values

When reading some float or long value in Modbus request, the values are represented as sequence of 4 bytes. To convert these bytes to a valid number, the byte order should be known.

There are 4 possible Byte Orders

1. No swap
2. Byte swap
3. Word swap
4. Byte and word swap

For floating-point values, we always use Bytes and Words Swap logic.

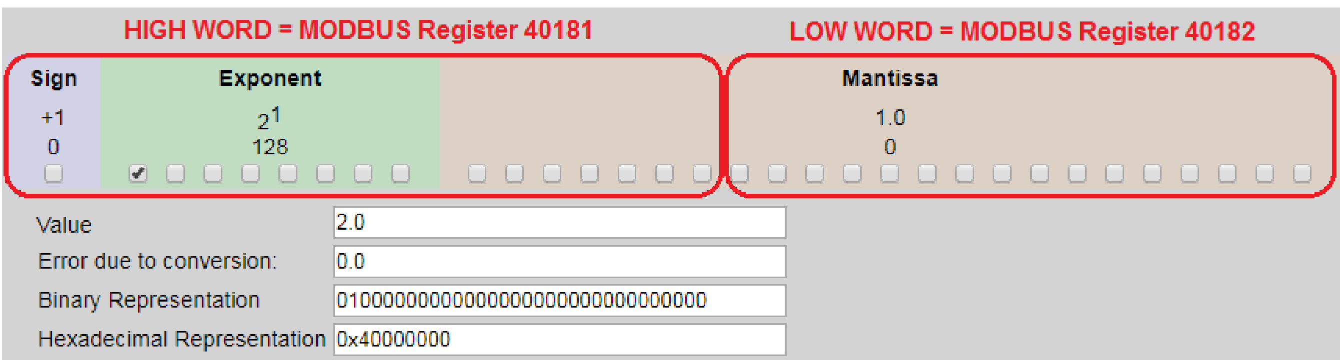
Example:

Bytes in response (from first to last):
0x01 0x02 0x03 0x04
Where 0x01 – first received byte and 0x04 – last received byte.
So in Byte and word swap logic, it means:
HEX: 0x04030201
DEC: 67305985
FLOAT (SINGLE): 2.38793926E-38

Example 1: Reading a Velocity Value on Axis 2

The sensor is in Vibration mode as shipped from the factory. So if the sensor mode has not been changed, there is no need to write to the SYSTEM MODE register. Simply read the velocity for Axis 2. For this, we read the OVERALL VELOCITY AXIS 2 register (40181). This register returns the instantaneous velocity on Axis 2 as an IEEE-754 floating point number in registers 40181 and 40182.

If the Velocity on Axis 2 is 2.0 ips, this would be represented as an IEEE-754 floating point number as shown below:



As can be seen from this example, the IEEE-754 floating point number is represented as a mantissa, exponent and a sign bit. The sign bit, exponent and part of the mantissa is contained in the HIGH WORD of OVERALL VELOCITY AXIS 2 (register 40181) and the rest of the mantissa is contained in the LOW WORD of OVERALL VELOCITY AXIS 2 (register 40182).

The following hexadecimal values would appear in the MODBUS registers:

40181 = 4000 hex (16384 decimal)

40182 = 0

Many MODBUS masters such as PC programs or PLC’s have the ability to represent two 16-bit integer MODBUS registers as a IEEE-754 floating point number. The formula is:

$$-1^{S} \times (1.0 + 0.\text{Mantissa}) \times 2^{\text{Exponent}-127}$$

In our example, sign bit is 0 (showing a positive number). -1Sign becomes -10 which is 1. This means the sign bit does not affect the number. So we effectively have:

$$(1.0 + 0.\text{Mantissa}) \times 2^{\text{Exponent}-127}$$

In our example, Mantissa is 0 and Exponent is 128. So the result is calculated as:

$(1.0 + 0.0) \times 2^1$

which is

$(1.0) \times 2$

which is 2.0 ips.

Example 2: Configuring a Velocity Alarm on Axis 3

The sensor is in Vibration mode as shipped from the factory. So if the sensor mode has not been changed, there is no need to write to the SYSTEM MODE register.

To capture an alarm and read the velocity value corresponding to the alarm, we first set an alarm threshold. For example, we want to trigger an alarm any time the velocity value exceeds the alarm threshold of 1.0 ips.

We will use Alarm Channel 1 to program the alarm. Alarm configuration registers are non-volatile. In other words, they retain their values even if the power to the VTB-RS485 sensor is cycled. Non-volatile registers need to be unlocked first before writing to them and then after the write, they need to be locked. Locking and unlocking is done through SYSTEM CONTROL (register 40002).

a) Unlock the non-volatile writes

Write a value of 42074 to SYSTEM CONTROL (register 40002). 42074 is a specially designated value for UNLOCK operation. Keep reading register 40003 and wait until it is 0 (idle).

b) Select the Alarm Axis:

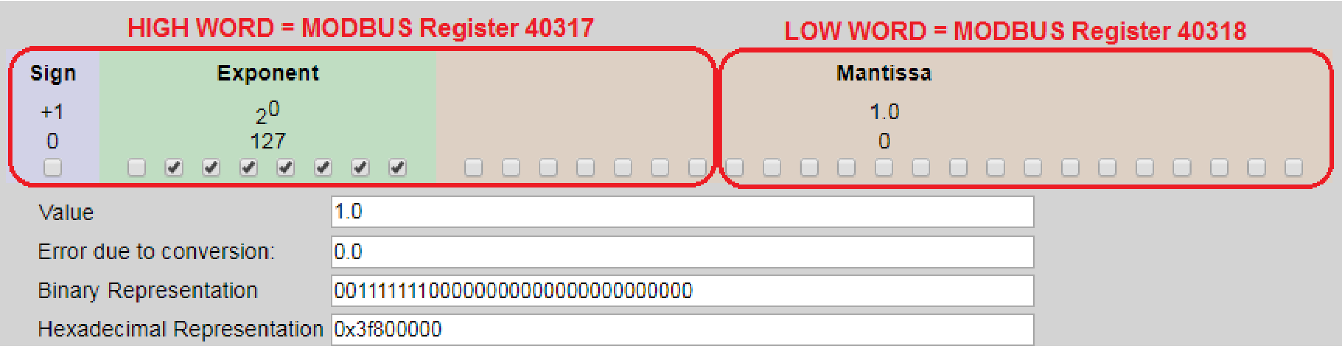
Write a value of 8 to ALARM1 AXIS (register 40313). 8 is Bit 3 which denotes Axis 3.

c) Select the Alarm Type:

Write a value of 2 to ALARM1 TYPE (register 40314). 2 is Bit 1 which denotes Velocity.

d) Decide if we want LOW, HIGH or a HIGH HIGH alarm. For now, we will use only a HIGH alarm.

Write the HIGH HIGH alarm threshold to ALARM1 HIGH LEVEL registers (HIGH WORD is 40317 and LOW WORD if 40318). Alarm threshold is IEEE-754 floating-point number that is represented as two 16-bit MODBUS registers. Many MODBUS masters such as PC programs and PLC's do this conversion automatically.



e) Lock the non-volatile writes

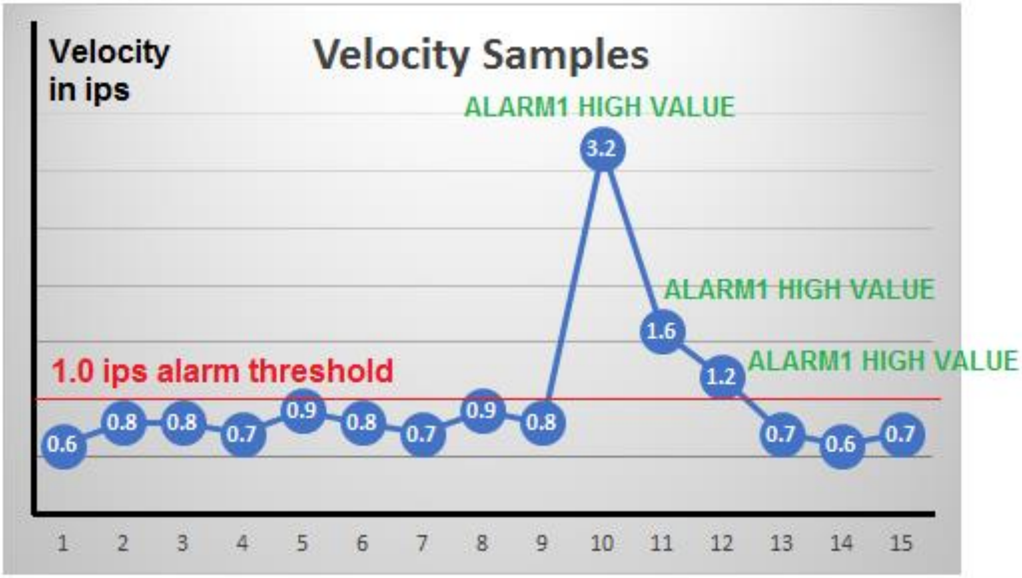
Write a value of 42075 to SYSTEM CONTROL (register 40002). 42075 is a specially designated value for LOCK operation. Keep reading register 40003 and wait until it is 0 (idle).

Now we have set the alarm threshold to a value of 1.0 ips, any time the RMS value of velocity on Axis 3 exceeds the alarm threshold of 1.0 ips, we should detect an alarm.

How do we know that we have an alarm ? This is achieved by monitoring the MODBUS register ALARM1 STATE (register 40227). This register shows which type of alarm we detected. Since we configured the Alarm Channel 1 to detect only HIGH alarms, we can expect an alarm only when velocity goes from NORMAL to HIGH or when it returns from HIGH to NORMAL. ALARM1 STATE would show 2 when a HIGH alarm is present and would show 4 when the velocity returned to normal levels.

The alarm value is read from the MODBUS registers ALARM1 HIGH VALUE (HIGH WORD is 40256 and LOW WORD is 40257). This is a velocity reading that is an IEEE-754 floating-point number and is in ips. For example, is the velocity spike that caused the alarm was 3.2 ips, ALARM1 HIGH VALUE would contain a floating-point number that represents 3.2.

The chart below shows the various velocity samples that are measured and calculated by VTB-RS485 sensor:



1.0 ips is the programmed alarm threshold for Alarm Channel 1. First 9 samples do not exceed this threshold.

Sample 10 exceeds the 1.0 ips threshold. So ALARM1 HIGH VALUE becomes 3.2. If the MODBUS Master reads the ALARM1 HIGH VALUE registers just after sample 10 is calculated, the ALARM1 HIGH VALUE would contain 3.2.

Sample 11 is 1.6 ips. It is still above the 1.0 ips threshold. So ALARM1 HIGH VALUE is updated to 1.6 ips after sample 11 is calculated.

Sample 12 is 1.2 ips. It is still above the 1.0 ips threshold. So ALARM1 HIGH VALUE is updated to 1.2 ips after sample 12 is calculated.

This way, the instantaneous value of the velocity that causes an alarm can be tracked by continuously reading MODBUS register ALARM1 HIGH VALUE.

So in this example, once the alarm channel has been configured, the strategy to monitor for alarms:

Forever

{

Read MODBUS register 40227

If the alarm type is HIGH

{

Read MODBUS registers 40256 and 40257 to see the velocity value that caused the alarm.

Take some action based on alarm, for example sound a buzzer, activate a relay, etc.

}

}

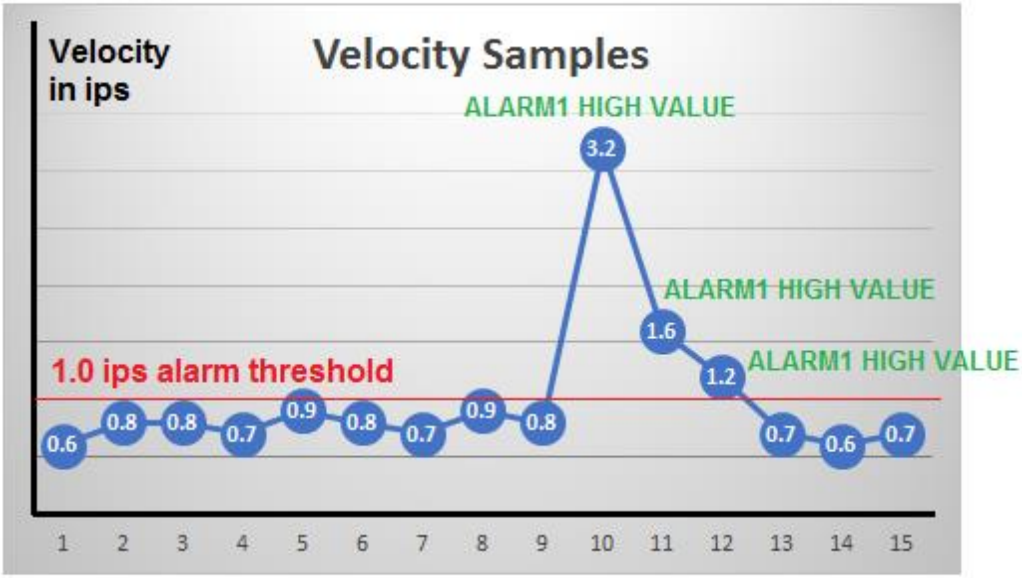
Example 3: Reading Latched Alarm Values

This is like Example 2 except that instead of reading instantaneous alarm values that exceeded the alarm threshold, we read the captured (latched) highest alarm value. For this we read the ALARM1 HIGHEST LATCHED VALUE (registers 40231 and 40232).

The difference between ALARM1 HIGH VALUE and the ALARM1 HIGHEST LATCHED VALUE:

- ALARM1 HIGH VALUE always contains the LAST alarm value that exceeded HIGH alarm threshold.
- ALARM1 HIGHEST LATCHED VALUE contains the HIGHEST alarm value that exceeded HIGH alarm threshold.

Using the values from Example 2:



ALARM1 HIGH VALUE would first contain 3.2, then 1.6 and then 1.2.

ALARM1 HIGHEST LATCHED VALUE would contain 3.2 and keep 3.2 because 3.2 is the highest HIGH alarm value ever calculated. So ALARM1 HIGHEST LATCHED VALUE would forever latch the highest alarm value until it is explicitly cleared by using the ALARM CONTROL (register 40239). Writing 1 to register 40239 clears the ALARM1 HIGHEST LATCHED VALUE.

Appendix A: System Error Bit Definitions

POWER ERROR	0	If this bit is 1, supply voltage is outside expected range and/or main regulator is not working.
RAM ERROR	1	If this bit is 1, dynamic RAM did not pass the power-up self-tests.
ADC ERROR	2	If this bit is 1, vibration voltage reading is within 1% of positive or negative voltage rail of measurement window.
NVM ERROR	3	If this bit is 1, non-volatile memory failed a WRITE operation
TEMPERATURE ERROR	4	If this bit is 1, the on-board temperature sensor did not respond.
CALIBRATION ERROR	5	If this bit is 1, the sensor was not calibrated. Contact Machine Saver.
COLLECT ERROR	6	If this bit is 1, the sensor had problems calculating vibration. Possibly some MODBUS register could not be read. Internal Error.
IMPACT ERROR	7	If this bit is 1, the sensor had problems calculating impacts. Possibly some MODBUS register could not be read. Internal Error.
ALARM ERROR	8	If this bit is 1, an alarm channel was configured incorrectly.
ALARM LEVEL ERROR	9	If this bit is 1, an alarm condition is being detected. This is not an error but regular alarm operation of the sensor. This is kept for backwards compatibility with older versions of VTB software.
COLLECT2 ERROR	10	Reserved for now.
SIN TABLE ERROR	11	Only firmware version 90 uses this. Reserved for now.
DETONATION ERROR	12	If this bit is 1, the sensor had problems calculating detonation. Possibly some MODBUS register could not be read. Internal Error.
BOARD TYPE ERROR	13	If this bit is 1, sensor detected incorrect board type. Contact Machine Saver.