27-08-2020

KPV200 Users Guide version 1.3

Morten Vinggaard
KJAERULF PEDERSEN A/S

Table of Contents

Hardware setup.

Minimum requirements for use of the vibration sensors together with the KPV200 LabVIEW software is:

1. A KPV200 vibration sensor.
2. A power supply. The sensor can be sourced from 10V to 28V.
3. M12 cables. The sensor comes with a M12 connector.
4. A RS485 to serial converter. The software uses serial communication (COM port). Typically, a RS485 converter comes as RS485 to USB. The USB connection pop up as a virtual COM port on computer. An ADAM-4561 will work. It has good galvanic isolation. Many other converters will work too.
5. A PC. With COM port or USB ports. Depended on the RS485 to serial converter.
6. 120Ω termination resistors.

Connecting a sensor.

Several sensors can be mounted on the same Modbus connection. Each sensor has an unique address on the network. Use shielded twisted pair cable. Use termination in the farthest ends of the bus connection.

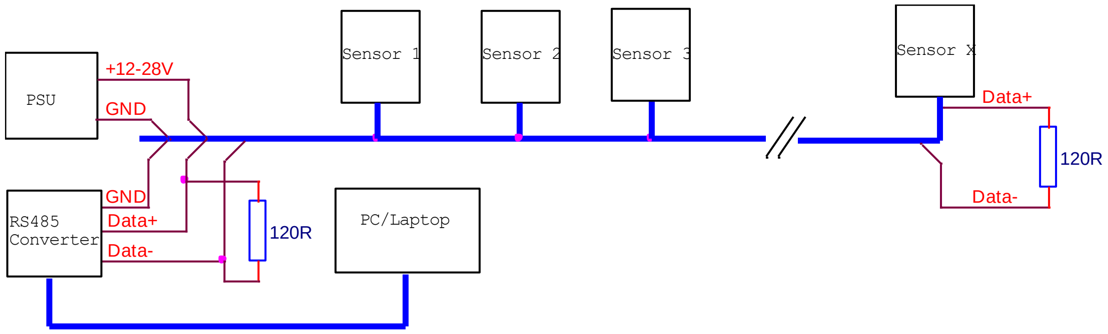


Figure 1 Typical installation with 120Ω termination in the farthest ends of the bus..

It is not recommended to make the RS485 setup as a star connection. It's better to daisy chain the sensors like Figure 1.

From the sensor to the RS485 bus a cable with M12 connector must be used. For this Users Guide a Lumberg cable is used, see Figure 2.

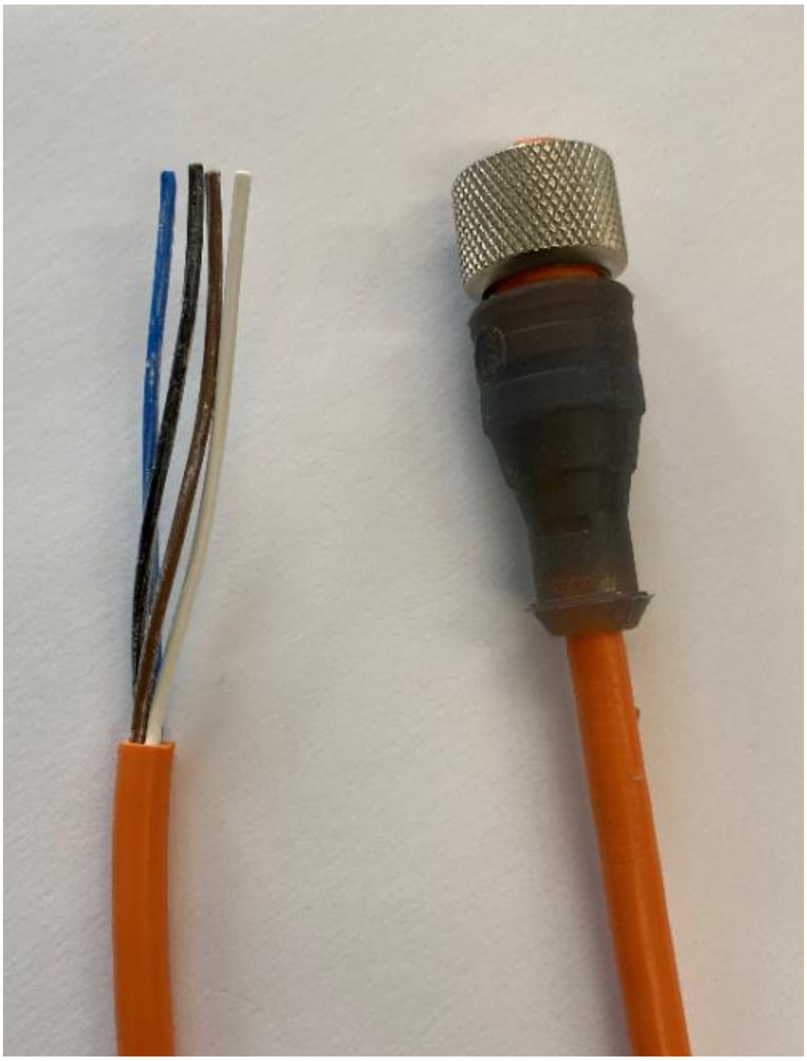


Figure 2 Lumberg cable.

The cable is used from the sensor and to the RS485 bus. On Figure 3 the connection setup can be seen.

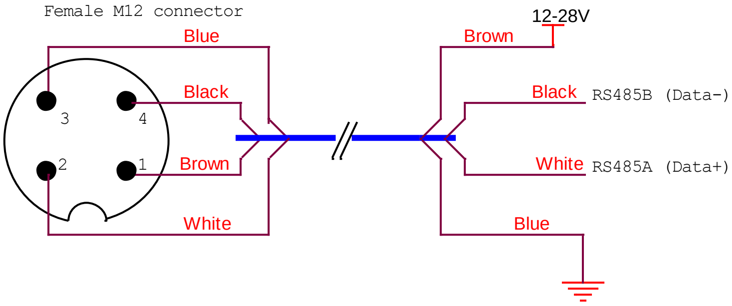


Figure 3 Connections of the Lumberg cable.

KPV200 Software.

The software is developed in LabVIEW 2018.

The software comes on an USB stick. When received it is owned by the company and can be used for all your own installations. The software cannot be sold to third parties.

Installation.

To use the software, you need to have administrator rights on the computer. Or at least, one with administrator rights needs to install the software before it can be used.

The software will run on Windows 7 and up.

It has been tested on:

OPR: WIN7 Proff. 64 Bit

CPU: I5, 2.80GHz.

6 GB RAM.

And on:

OPR: WIN10 Proff. 64 Bit

CPU: I7, 1.80GHz.

32GB RAM.

No issues found.

Find the install.exe on the USB stick.

Run the installation. Follow the recommendations from the installation software . The software will install all the needed National Instruments drivers, install the KPV200 software and create a Library for the KPV200 software and install a setup file in this library. See “The setup file.” for more information.

First run.

Connect the RS485 converter to PC and to the sensor. Connect the power supply to the sensor and run the software.

When the KPV200 software is started first time, no data will be seen on the screen.

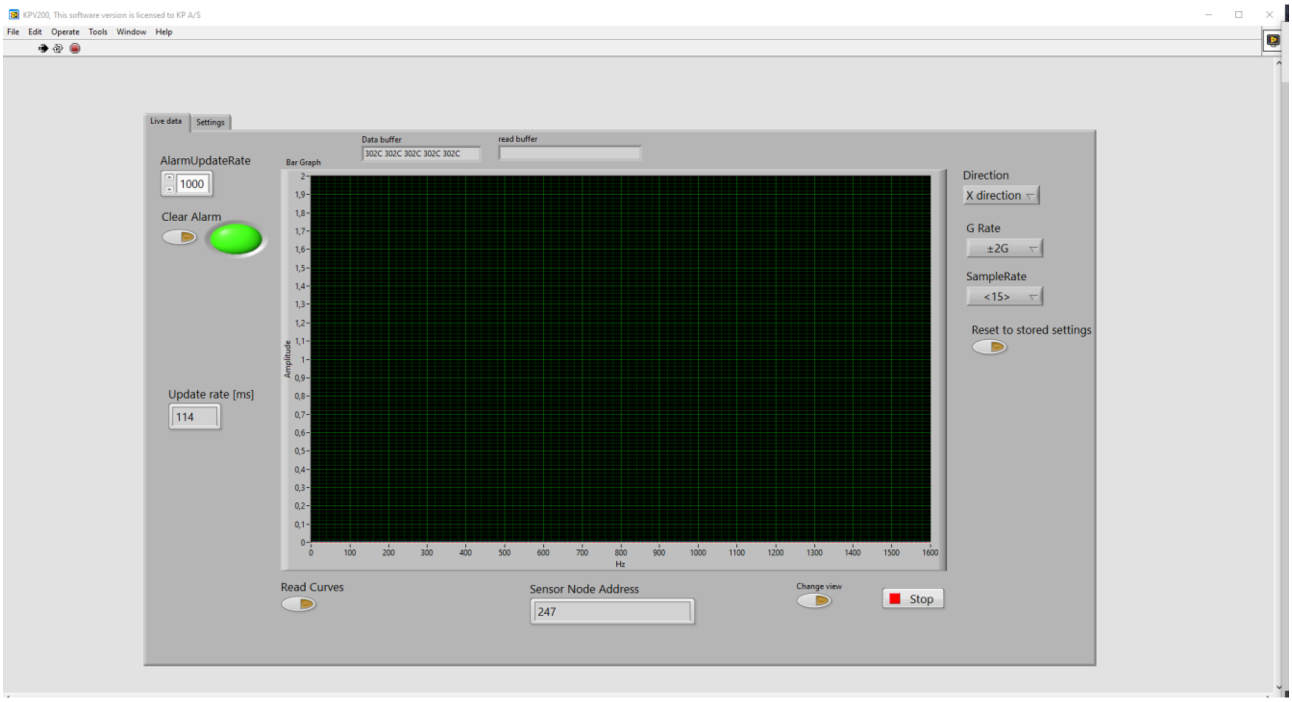


Figure 4 First time run.

Click on the settings tab. No PC comport is selected.

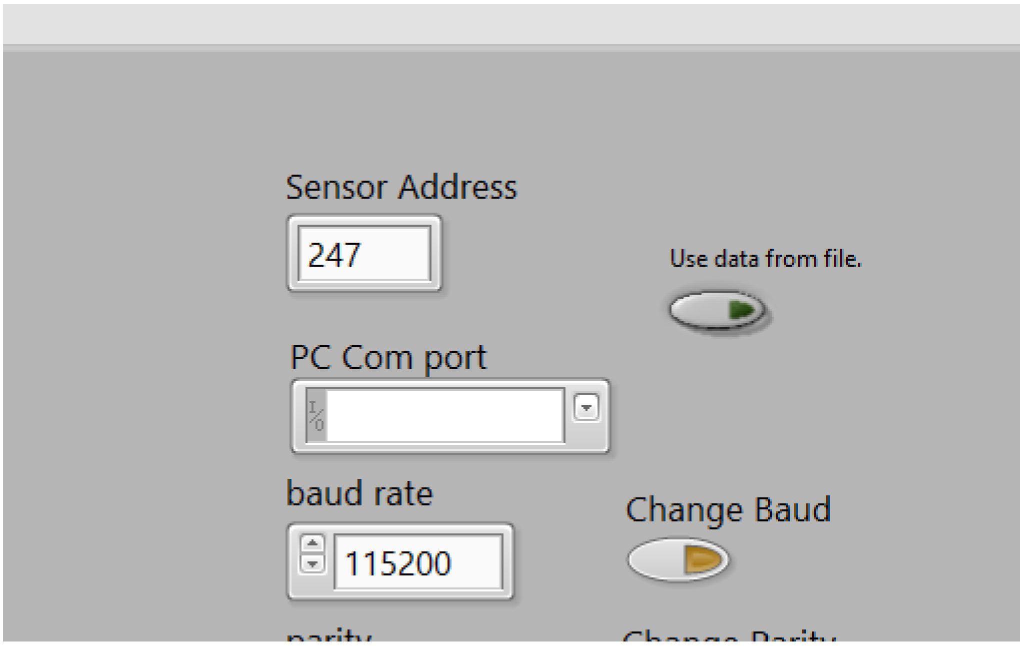


Figure 5 No comport selected.

Select the correct comport. This is the comport where the RS485 converter is connected.

Go back to Live data.

If the sensor is brand new and has the address 247, from factory, then data should be seen on the screen. It is recommended to restart the software pushing the red button just below Operate and afterwards push arrow to the left to restart the software again see Figure 6.

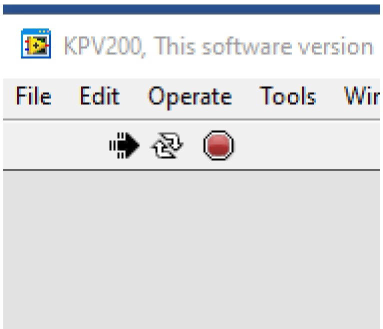


Figure 6 Start and stop buttons.

Users guide KPV200 LabVIEW software.

KPV200 and KPV200 LabVIEW software is a system for measuring vibrations.

The KPV200 is a vibration sensor.

Figure 7 KPV200 vibration sensor.

The sensor communicates on RS485 Modbus RTU. To monitor the sensor a GUI (Graphical User Interface) has been developed.

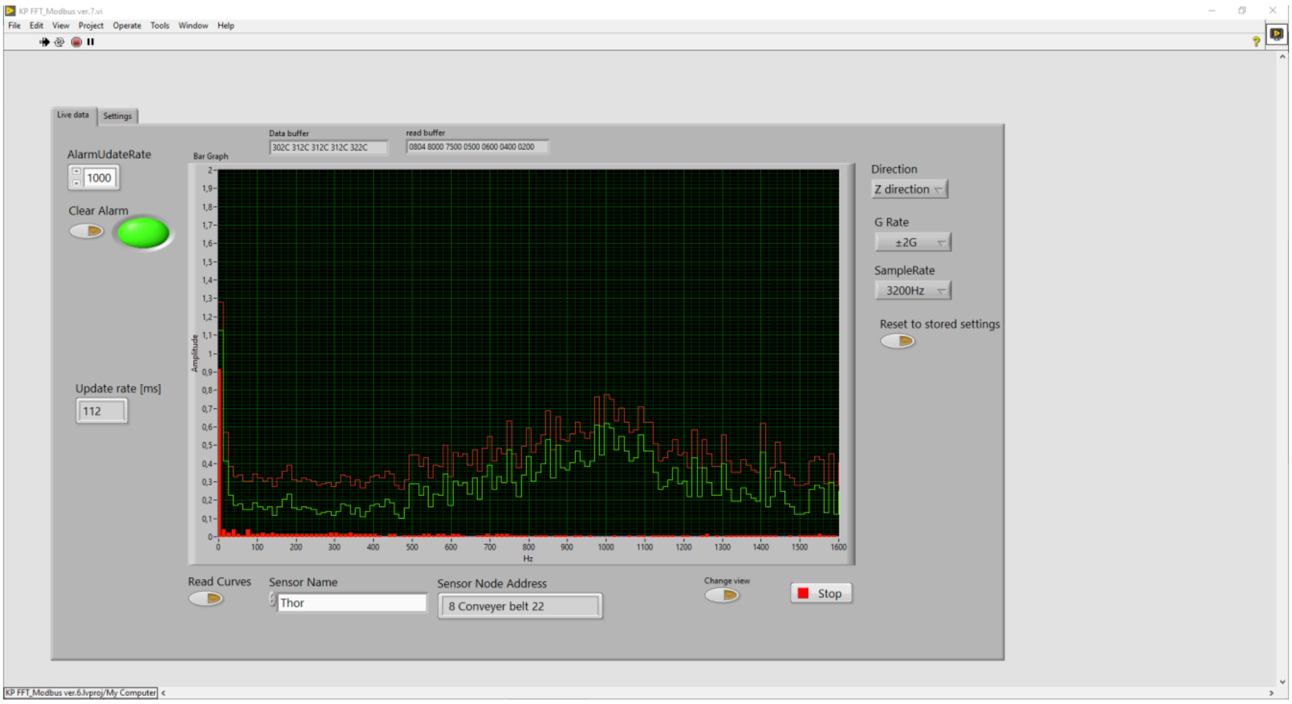


Figure 8 KPV200 GUI.

The sensor is intelligent. All FFT data and all data is calculated in the sensor. LabVIEW is only the interface for the eye. All settings are stored int the sensor, except the settings related to the computer running the software.

The KPV200 GUI has two main screens. One called “Live data” and one called “settings”.

The Live data screen contains all the live curves and stuff that moves. And the settings screen contains most of the settings.

Live data.

On the live screen all the graphs and FFT data are monitored. The system has two graphs and FFT data. The data on the screen is related to the sensor monitored. The sensor name and the sensor address can be seen at the bottom of the screen.

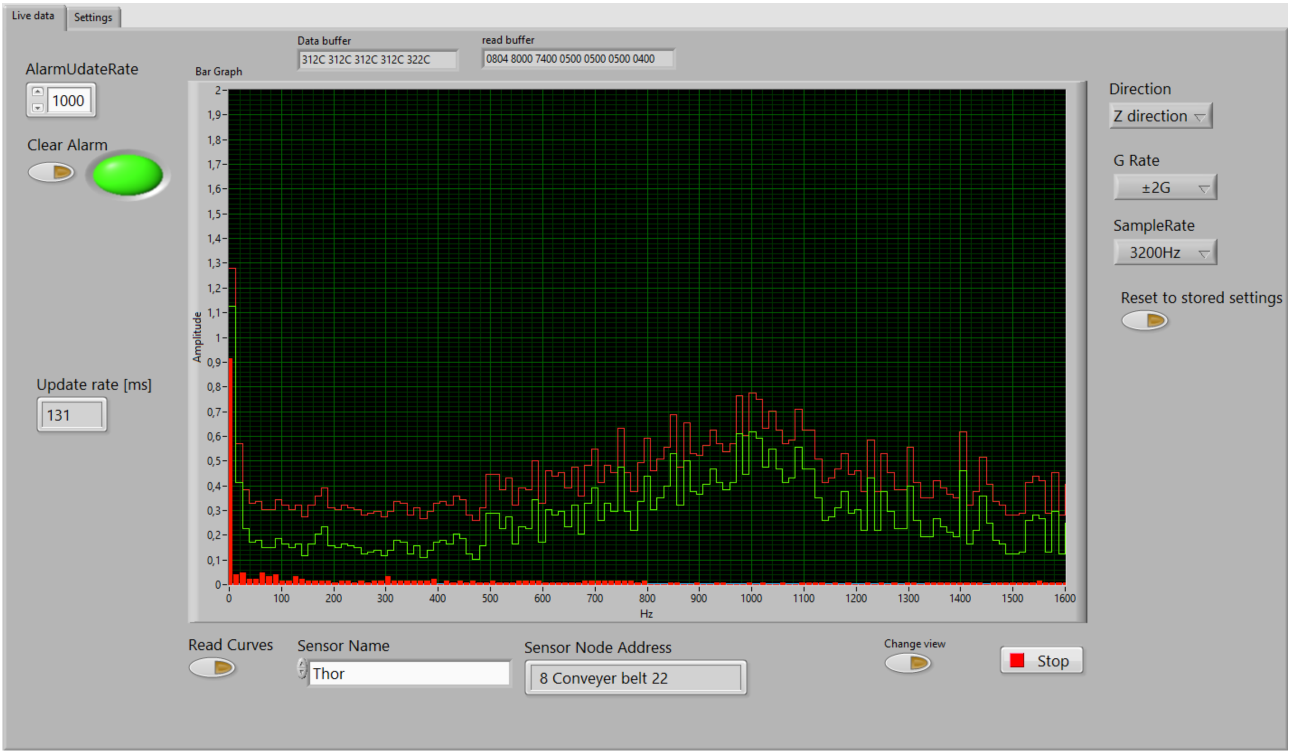


Figure 9 Live data screen.

On the screen the Alarms are monitored. If an alarm is set light green “lamp” will turn red. The Alarm can be cleared by pushing clear alarm button. AlarmUpdateRate is set to 1000, which means that the alarm is updated every 1000ms. This can be changed but it will return to 1000 on software restart.

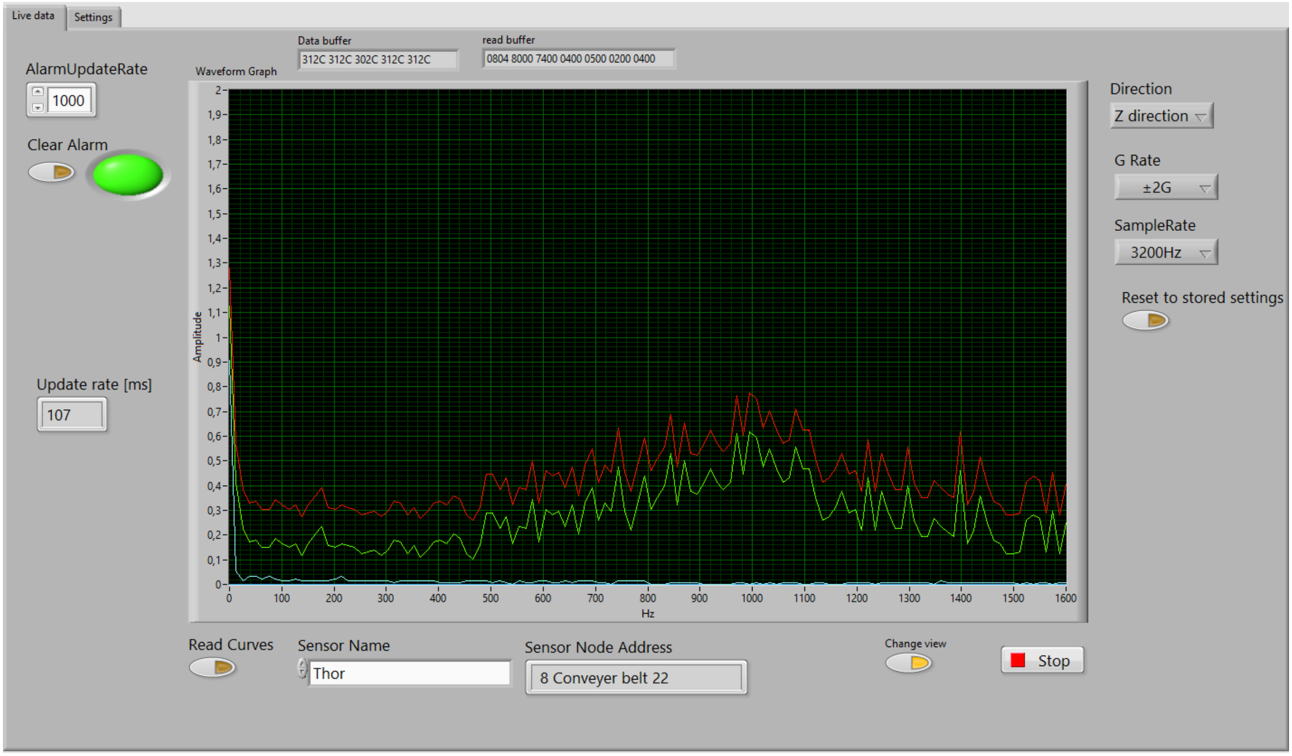
The Direction button changes the direction of the measurement, see “Direction, EEPROM. The only different is that that a change here is not stored in the memory. It is possible to make a lot of changes here. It is always possible to return to stored settings.

The G_Rate button changes the amplitude of the measurement, see “G Rate EEPROM.” for details. The only different is that that a change here is not stored in the memory. It is possible to make a lot of changes here. It is always possible to return to stored settings.

The SampleRate button changes the sample rate of the measurement, see “SampleRate, EEPROM.G Rate EEPROM.” for details. The only different is that that a change here is not stored in the memory. It is possible to make a lot of changes here. It is always possible to return to stored settings.

When you make these changes an alarm could get set. You can always return to stored settings by pushing the button “Reset to stored settings”.

If you don’t like bars you can push “Change View”. Then the appearance changes.



The curves on the GUI.

Two curves and FFT data are shown on the screen. The curves are Known curve, which is the characteristic of the sensor placed in its specific place. If the sensor is moved this characteristic is changed. The characteristic is measured by starting a “Learning Curve”. See “Start Learning curve. for details. The other curve is the Alarm Curve. The alarm curve is generated by adding an offset to the known curve.

See Figure 10.

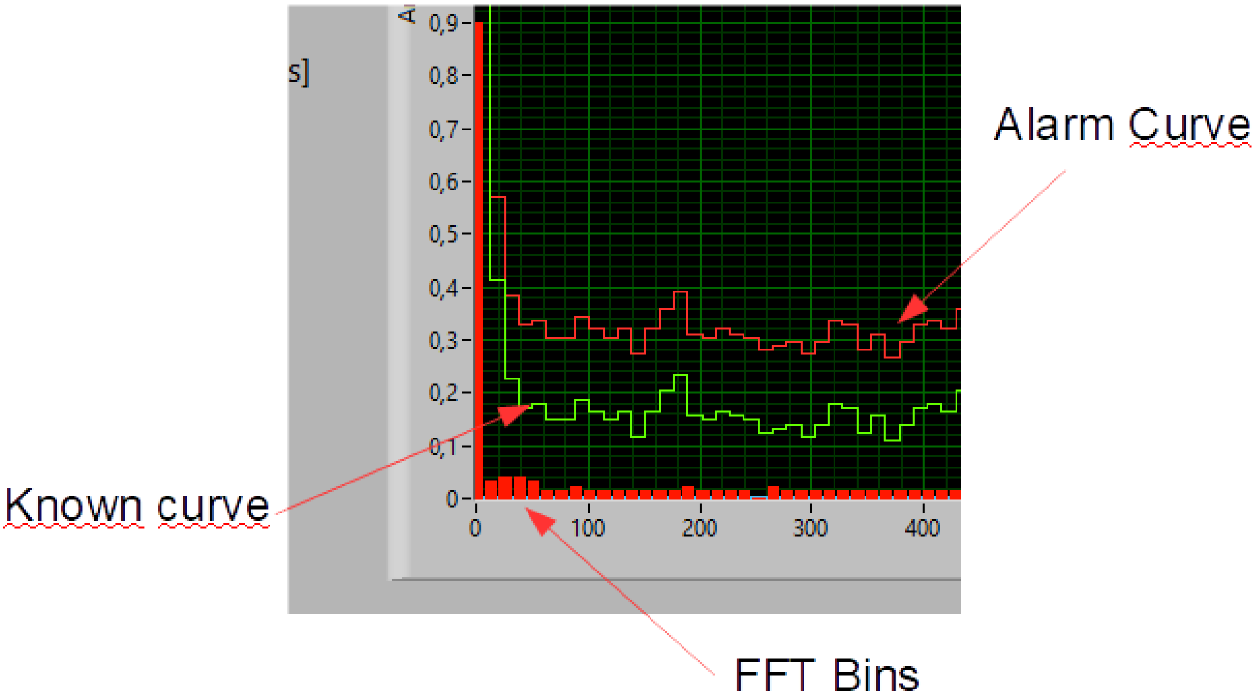


Figure 10 The FFT data and curves.

The FFT Bins is the frequency response from the FFT algorithm in the sensor. A bin represents a frequency span. If the system sample rate is 3200 Hz. Then the frequency response is 0 – 1600Hz. The system has 128 bins. Then each bin represents 1600/128 = 12,5 Hz.

The alarmcurve is the base for generating an alarm. If one of the 128 FFT bins is higher than the alarmcurve and alarm can be generated. Generating this alarm is also depended of the alarm delay. See the section regarding “Alarm Delay. For a detailed description.

In Figure 11 an alarm is generated due to this permanent high FFT Bin.

Figure 11 Alarm is generated.

Settings

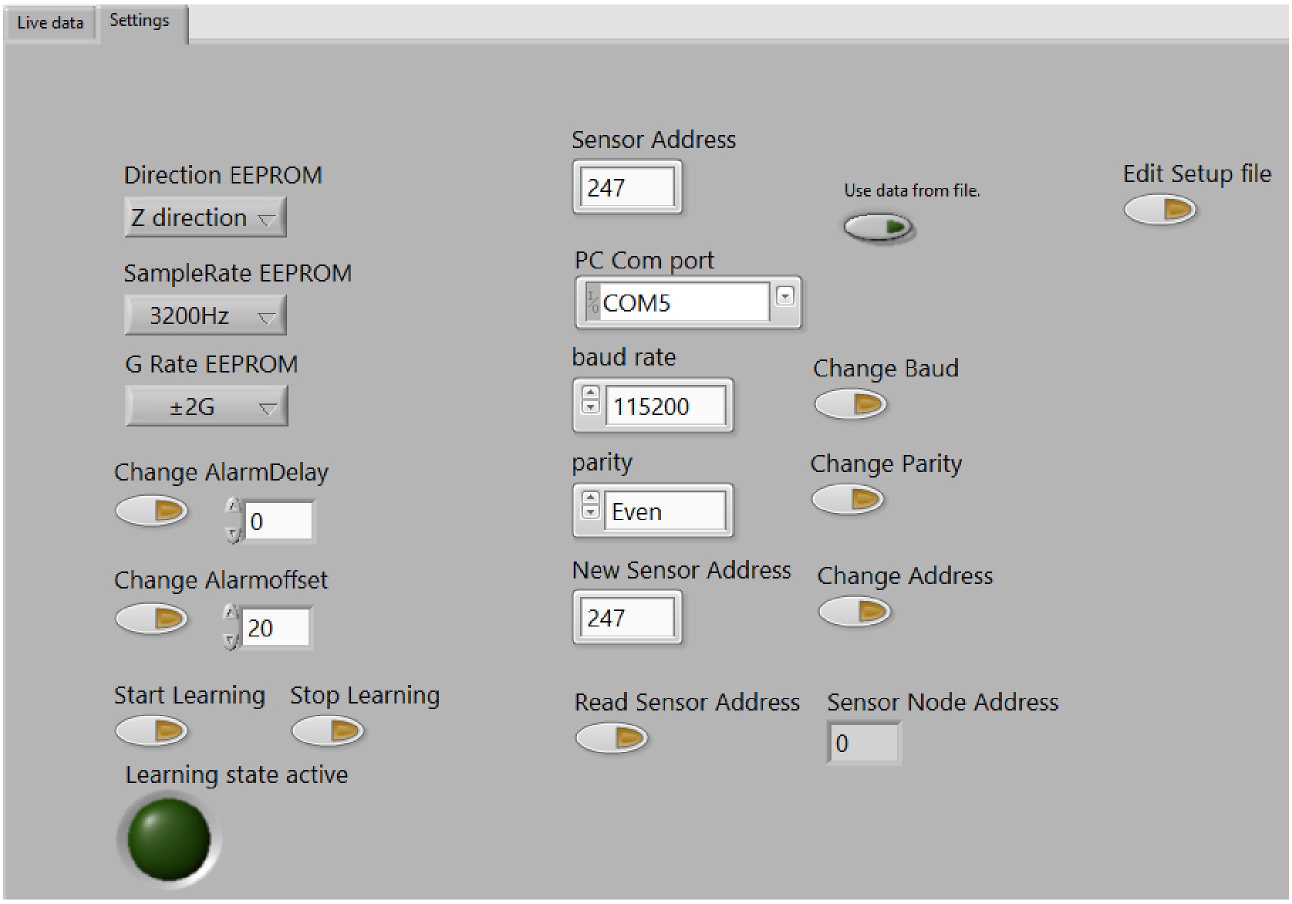


Figure 12 Screen cut from the setting page.

On Figure 12 a screen clip from the setting screen is shown. Some of the settings on this page will be stored in the sensor memory. These settings will be the standard setup for the sensor when the sensor is power booted.

In the next couple of pages, a description of the settings will be shown.

Direction, EEPROM.

The direction button is for setting the direction of the data used for the FFT calculation.

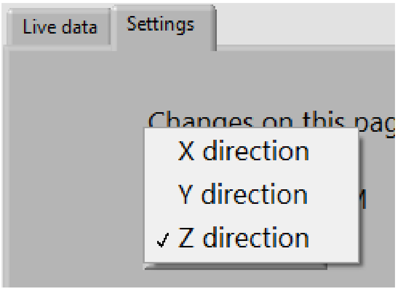


Figure 13 Direction

The sensor can measure in three directions. X, Y and X. There can be huge different in vibration amplitude depended of the measurement direction.

Figure 14 Directions on KPV200 vibration sensor.

On Figure 15 a vibration sensor is mounted on top of an electrical engine. One sees a belt from a pulley mounted on the engine shaft. The forces in the Z directions will be higher than the forces in the X direction.

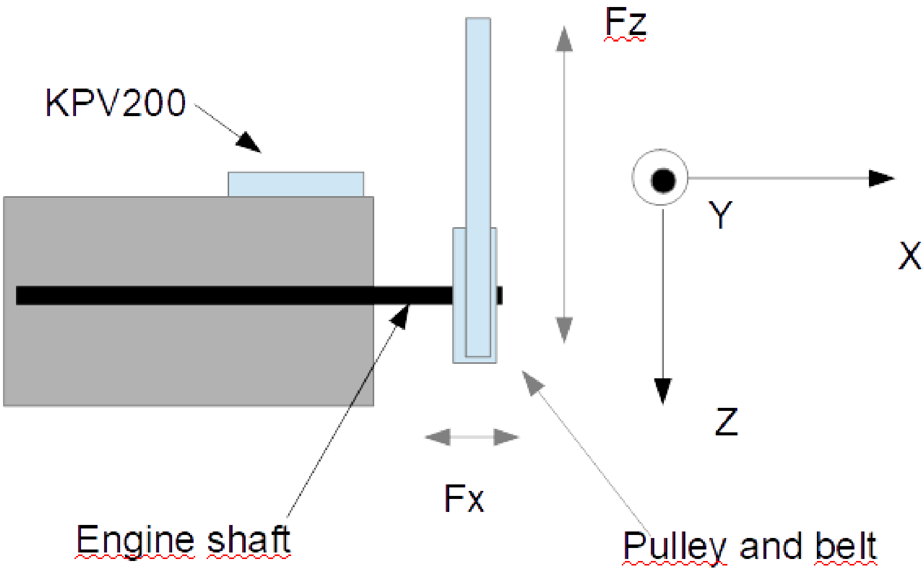


Figure 15 Model of engine with KPV200 mounted on the top.

One must be aware of the it does matter which direction is used for the measurement. Vibrations measured will vibrations from the system and vibration from the engine. Vibrations from the engine will typically be frequency’s that somehow corresponds to the speed of the engine. Vibrations from the system can be very random.

SampleRate, EEPROM.

The sample rate button is used for adjusting the speed of the samplings.

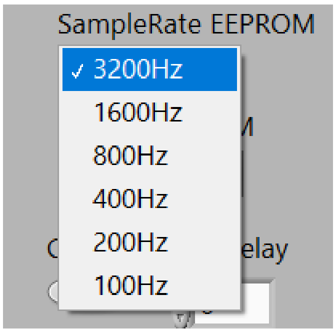


Figure 16 Sample rates

As it can bee seen on Figure 16 KPV200 has six different sampling rates. Going from 3200Hz down to 100Hz.

Measuring with 3200Hz gives a spectrum of 1600Hz due to the Nyquist criteria. In KPV200 via have 128 bins. A bin is “frequency container”.

If a signal the one in Figure 17 it is easy to see that this signal contains many frequencies. It is not a clean sinus. The signal can be treated in an FFT algorithms. This algorithm gives a response of the frequency content of the signal.

On Figure 18 the processed signal can be seen. Each red bar represents as frequency bin. At 3200Hz sampling rate a measurement range of 1600Hz can be made. Each bin represents 1600/128 12,5Hz.

Sample rate [Hz]	Range [Hz]	Hz / bin [Hz]
3200	1600	12,5
1600	800	6,25
800	400	3,125
400	200	1,5625
200	100	0,78125
100	50	0,390625

Table 1 Sample rates vs Hz/bin.

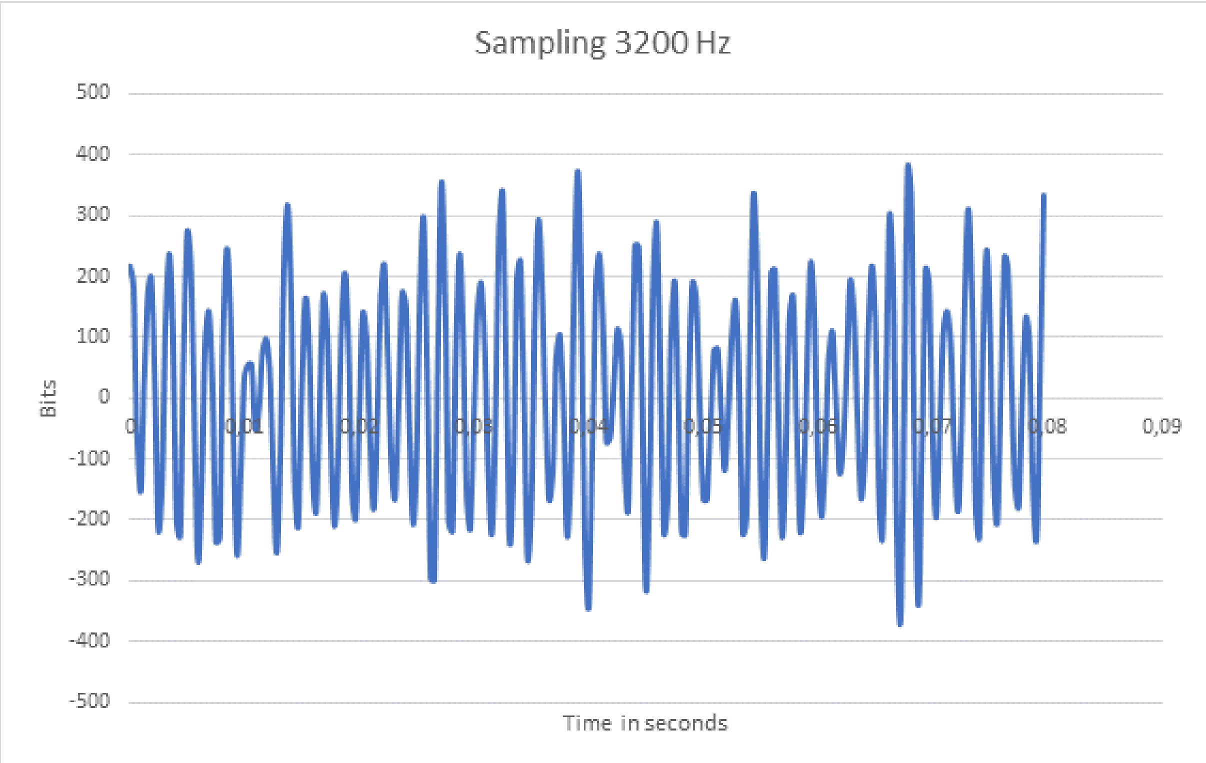


Figure 17 Signal measured at 3200 Hz.

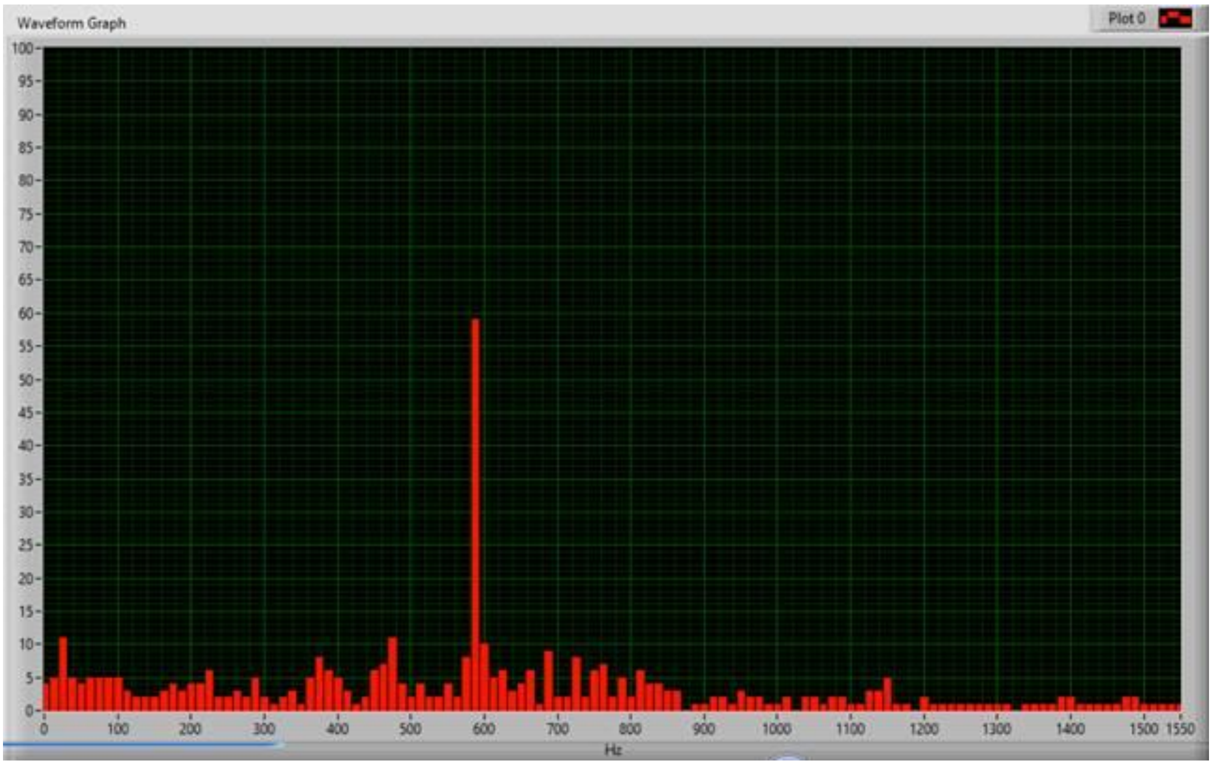


Figure 18 The processed signal.

G Rate EEPROM.

The G Rate button is used for adjusting the strength in the signal. 16G is a very strong signal, 2G is less strong.

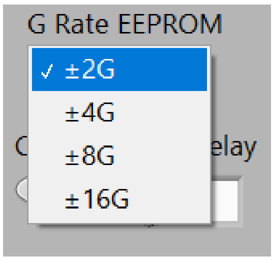


Figure 19 Adjustments for G Rate.

The sensor has four levels of G Rates. Normally 2 or 4 G is used. But again, it depends of the system.

Alarm Delay.

The default setting of the Alarm Delay is 0. This means if you have a signal higher than the Alarm Curve an alarm will be set immediately. This can sometimes be annoying. To avoid an alarm just because of a bump the alarm delay setting can be used. With that setting you can force the sensor to see several "too high" signals before an alarm is set. Mostly a signal that indicates an error is a permanent high bin.

Here is an example from real life.

A sensor is mounted on a motor driving a conveyer belt. When the system was started Normal and alarm curves was generated. The green and red curves. Everything was fine for a couple of months. The alarms began to appear. The alarm delay was used to make the alarm to be harder to set in the beginning. But in the end it was obvious that the sensor was measuring something that not was right. It shoved up that a ball bearing in a roll on the conveyer belt was damaged. That gave a permanent high signal on the sensor. After repair the high signal was gone.

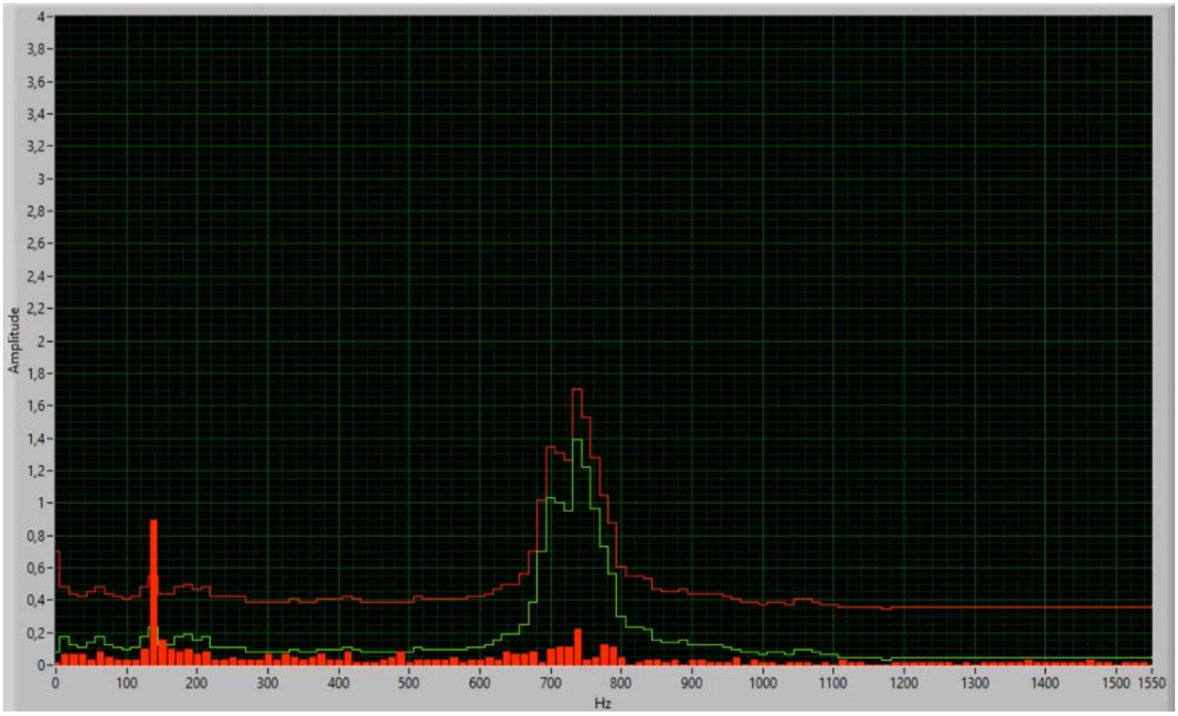


Figure 20 Failure measured by sensor.

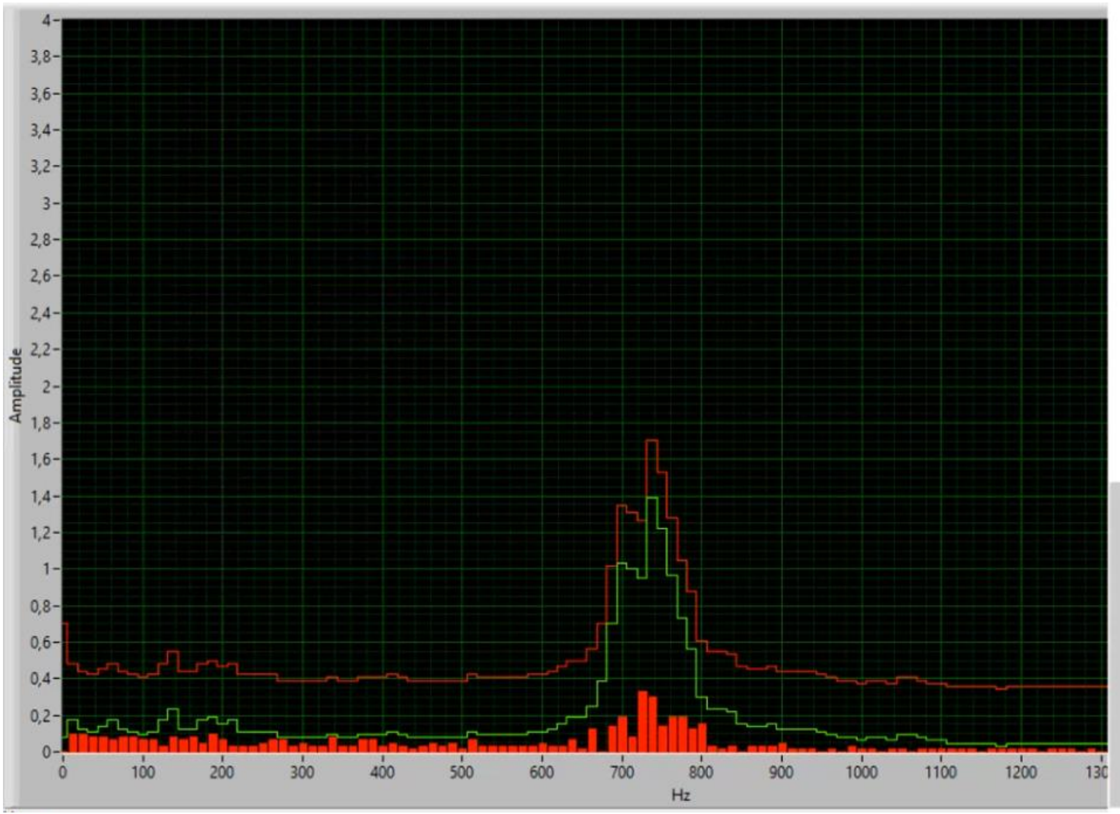


Figure 21 The failure was gone after repair.

Change Alarm Offset

The default setting of the Alarm offset is 10. This can be calculated to G value. The resolution on the G axis is G_Value/256. If your G value is 2 and your Alarm offset is 10, the offset can be calculated to 0,078G or 78 mG. This setting is used to define how high a signal can go over the normal curve before an alarm is set. The alarm is also depended on the alarm delay.

Start Learning curve.

When a sensor is brand new and never used before a default known curve and a default alarm curve is shown on the screen. The known curve is placed at 1G and the alarm curve is 78mG higher, due to the offset of 10 in the default alarm offset.

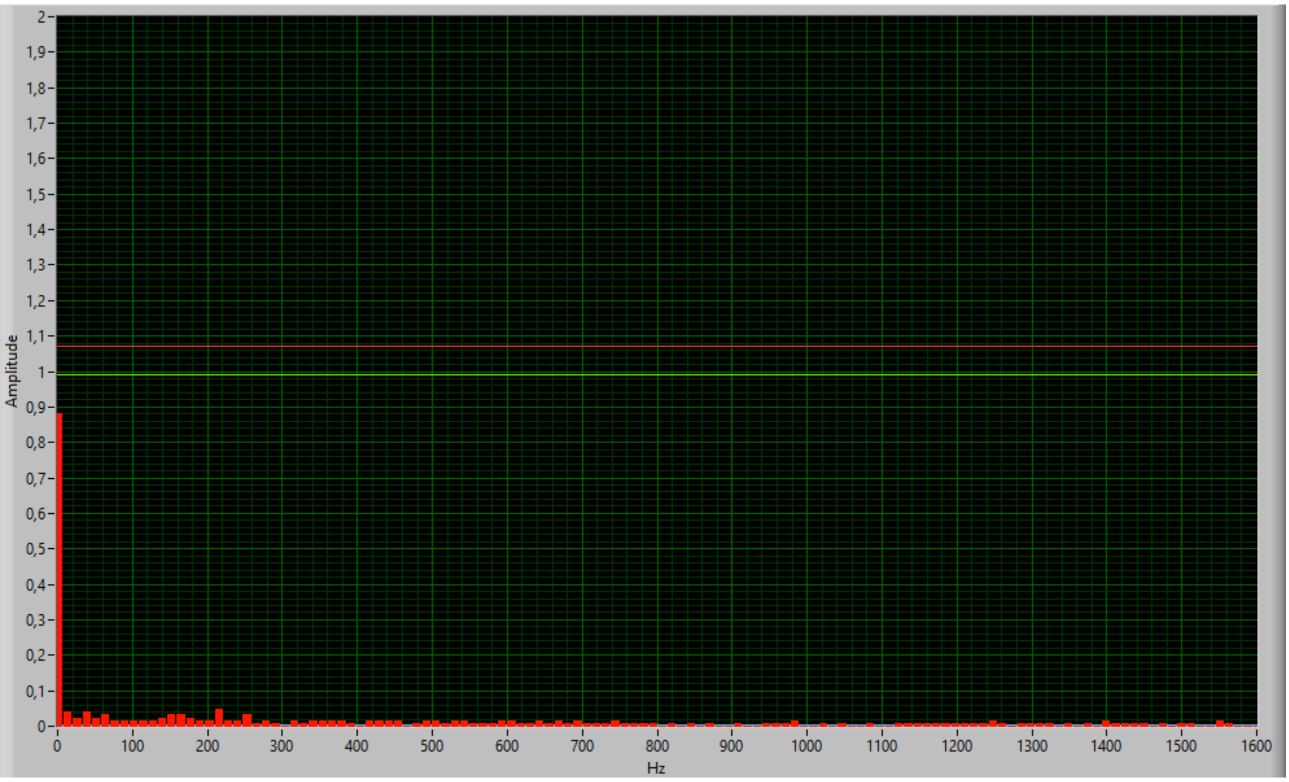


Figure 22 Characteristic of a new sensor.

When the Start Learning is pushed, the Learning state active turns light green to indicate the sensor is in learning mode.

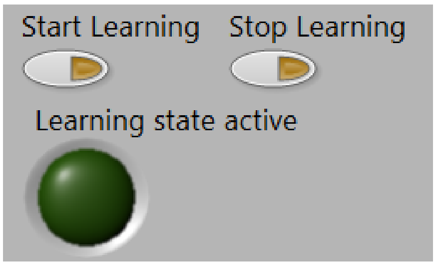


Figure 23 Control learning curves.

The button is pushed.

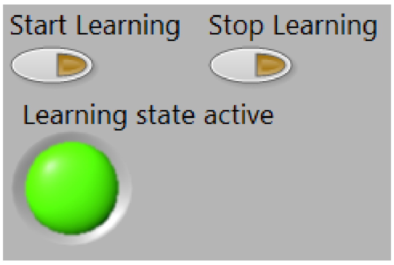


Figure 24 Active learning state.

Now after some time, maybe a week turn of the learning mode again. This is done by pushing Stop Learning. The characteristic of the sensor appears on the screen.

This is the characteristic of the sensor in that specific place.

It could turn out like this:

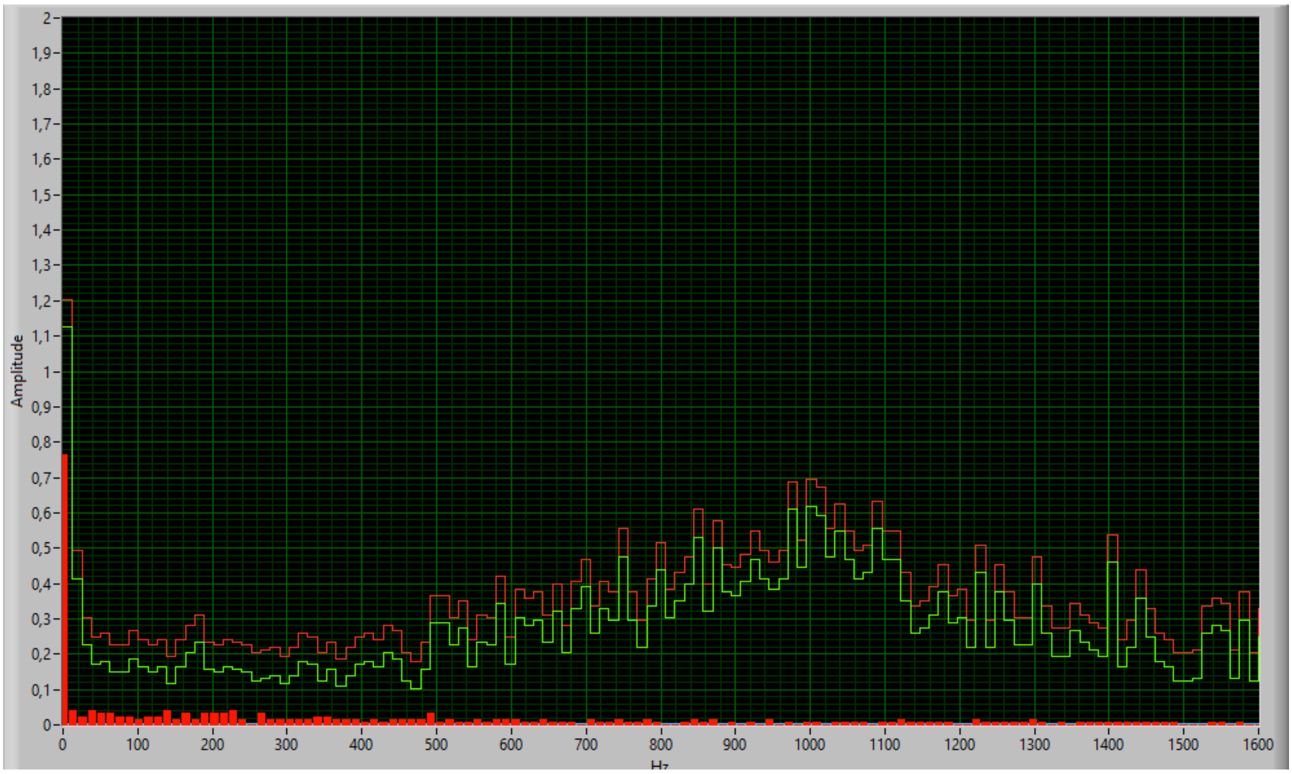


Figure 25 The new characteristic of the sensor.

One can add a little more alarm offset. 20 instead of 10 is set in the alarm offset.

Now the distance between the known curve and the alarm curve is 156 mG.

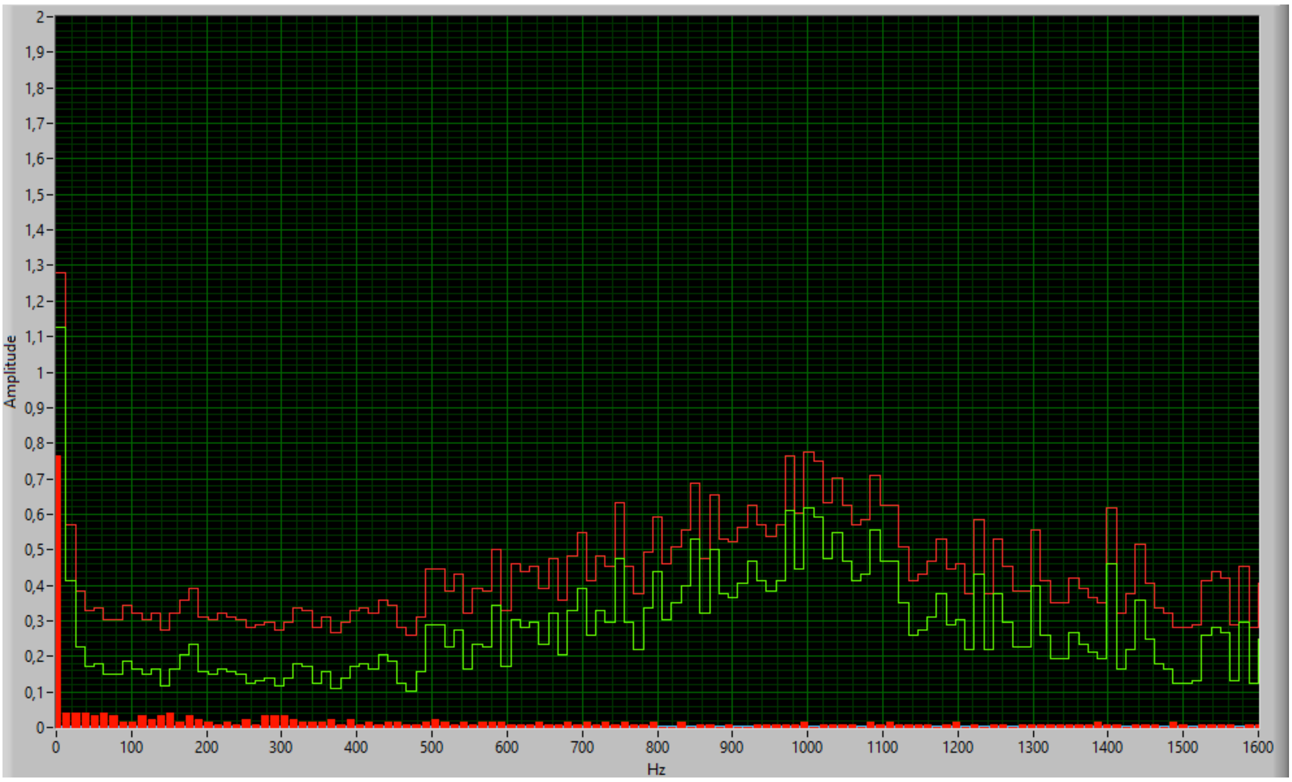


Figure 26 Offset added to alarm curve

The setup file.

The KPV200 software has possibility to store all communication settings and all sensor name and address information in a file. The software will read the file when it is started, if the software is to do it.

It is possible to edit the file from the software or one can edit the file from a txt based editor. Find the folder C:\KPV200Support files\

Edit sensorlist.txt. This file contains all information for the software system.

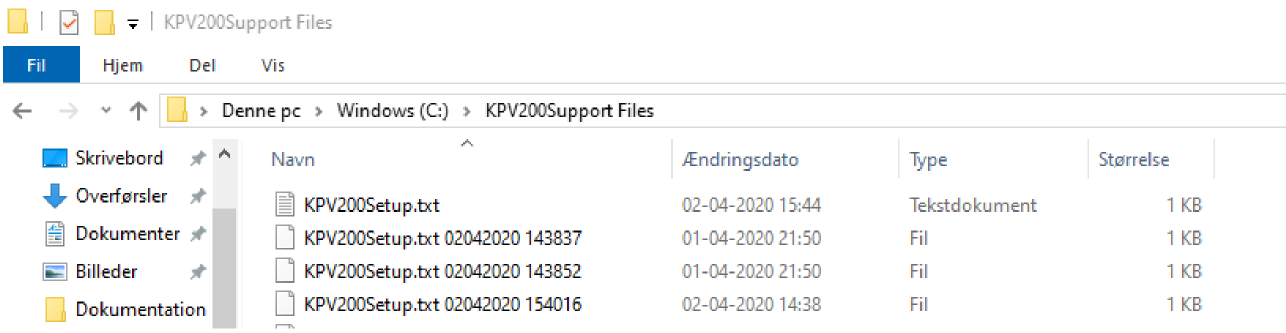


Figure 27 The KPV200Support files folder.

It is also possible to click the button “Edit Setup file”.

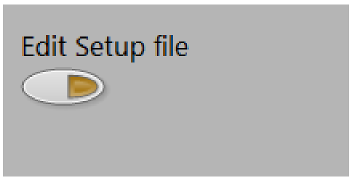


Figure 28 Editor for the support file

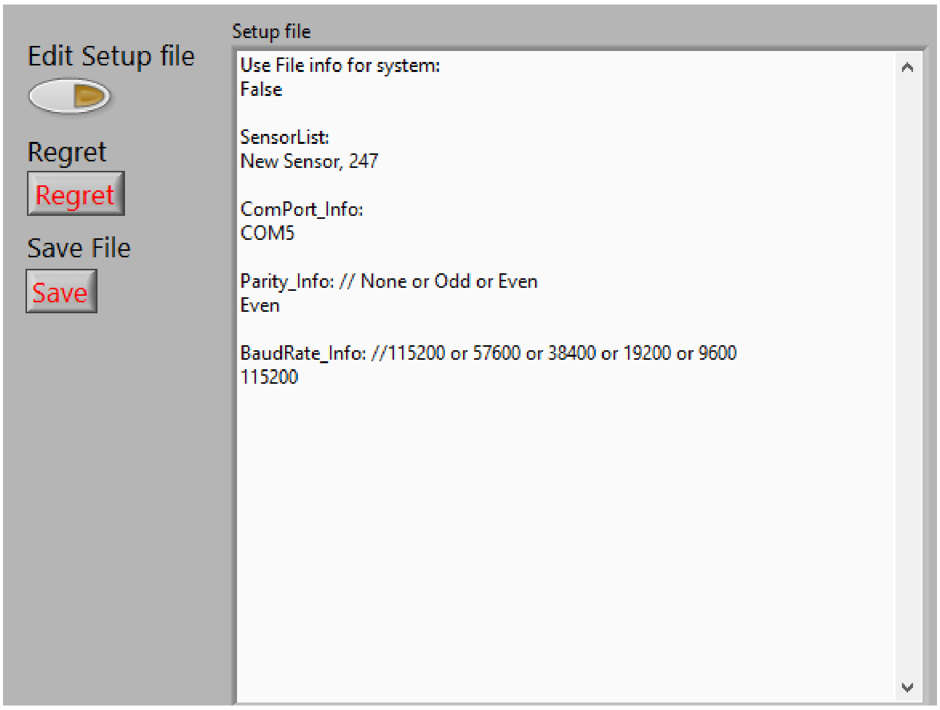


Figure 29 Open support file editor.

In the following each setting will be explained.

Use File info for system:

This setting tells the software if it should use setup file or the system should be manually controlled.

When the system is installed this setting defaults to False, which means that the system is manually controlled. By manually setup one must remember the communication setup and all sensor addresses.

All this information can be stored in the file.

If the setup file should not be used, then write False in the next line after:

It must look like this:

Figure 30 Don't use setup file

If the setup file should be used, then write True in the next line after :.

Figure 31 Use setup file.

The system is case sensitive; therefore it is important to use Capital letter first and then small letters afterwards.

SensorList:

The settings bellow “SensorList:” is a list of sensor names and sensor addresses.

It can be very difficult to remember all addresses in a system. And it can also be hard to remember placement. This is solved by adding a name followed by an address and placement.

When the system is new the only sensor in the sensor list is “New Sensor” with address 247. All new sensors from factory comes with the address 247.

Figure 32 Default sensorlist.

It a good idea to insert a new sensor with another address in the sensor list.

In the line after New Sensor, 247 I add the new sensor.

I am naming it Thor it must have address 8 and it is placed a conveyer belt 22.

Figure 33 Sensorlist with new sensor

Save the file by pushing the button “Save File”. See Figure 34.

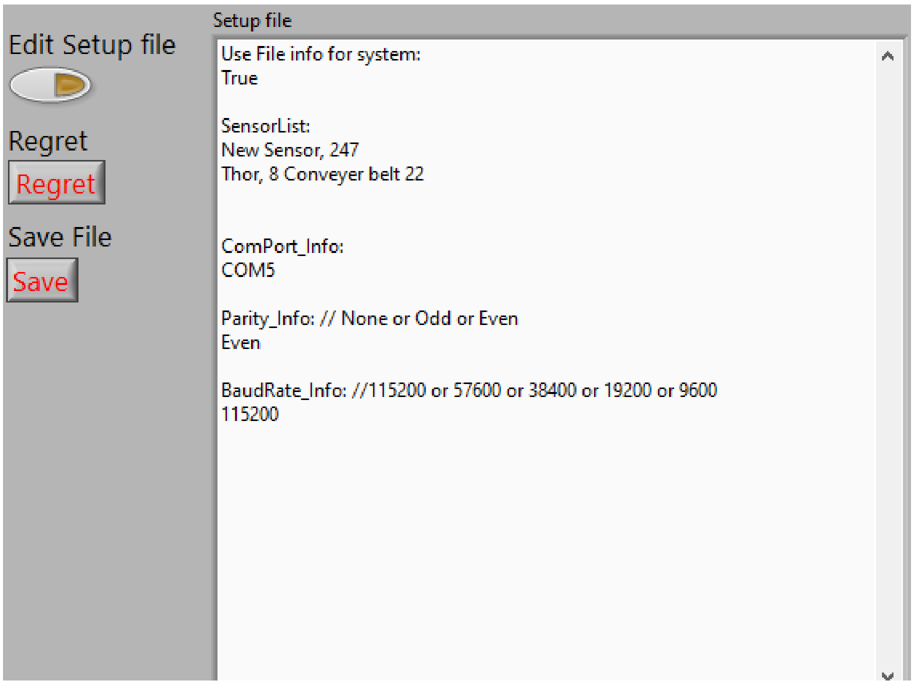


Figure 34 Sensor Added to sensor list push save when finished.

The sensor Thor now appears in the sensor list on the “Live Data” screen.

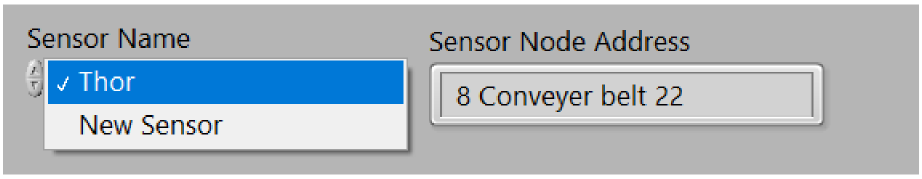


Figure 35 Thor appears in list.

Data will not flow on screen choosing sensor Thor. Address needs to be changed in the sensor. Go to the section “**New Sensor Address**.

As soon as Edit setup file is pushed a copy of the setup file will be stored in the same directory. The file name could look like this:

KPV200Setup.txt 02042020 154016

Where:

KPV200Setup.txt is Org. filename.

02042020 is date <Day_Moth_Year>

154016 is timestamp <Hour_Minute_Second>

It is always possible to go back in history.

ComPort_Info:
The comport info setting tells the system which comport to use. When using the software a RS485 module to USB or COM port converter is needed. A RS485 to USB converter typically pop up as a COM port on the computer. For test in the LAB an converter from ADAM is used. It has an isolated USB connection to protect the computer against surges.

It is very important to verify that the COM port works before starting the software. Else there will be some issues regarding communication. See section “Error handling.

On Figure 34 it can be seen that “COM5” is used for this software.

Parity_Info:
This setting is for the parity in sensor and computer. Default setting is Even. There is no need to change this if only vibration sensors are on the Modbus line. But if needed for some reason this can be changed to None or Odd. See Figure 34.

BaudRate_Info:
Default setting is 115200. Normally there is no need to change this. But sometimes it is necessary.

See Figure 34.

PC COM Port
It only makes sense to use this possibility if setup from file isn’t used.

It is possible to change the communication port from this setting. Available COM port will appear here.

By pushing the little arrow, it can bee seen that COM5 and COM26 is available.

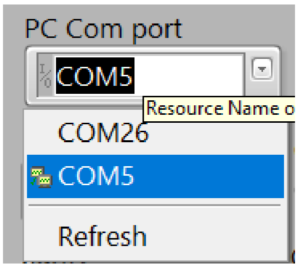


Figure 36 Available COM ports.

Baud Rate.
The BaudRate can be changed in the system. If an BaudRate of 57600 is need click on 57600 and afterwards click on the button “Change Baud”. The Baud Rate in the sensor will change immediately. Remember to change the Setup File afterwards if you are using this. See “The setup file.BaudRate_Info:”.

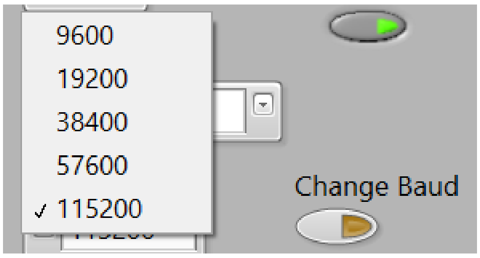


Figure 37 Change Baud Rate.

Parity.

The parity can be changed in the system. If Odd parity is need click on Odd and afterwards click on the button “Change Parity”. The Parity in the sensor will change immediately. Remember to change the Setup File afterwards if you are using this. See “The setup file.Parity_Info:”.

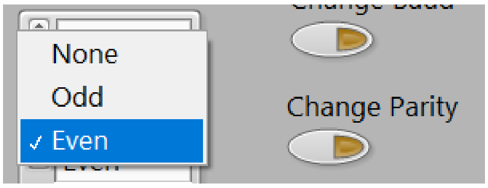


Figure 38 Change Parity.

New Sensor Address.

It is possible to change the address in the sensor. If several sensors are connected on the Modbus network each must have an unique address. Earlier in this document we added a new sensor to the sensor list. See “SensorList:”. It was Thor with the address 8. Now we need to change the address in the sensor.

I have set the system to communicate with the new sensor at address 247, simply by choosing it from the front sensor menu.

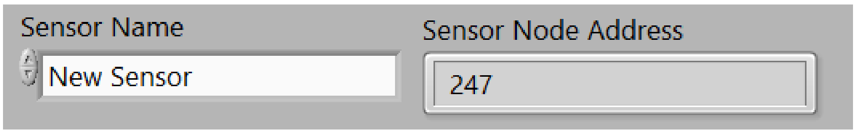


Figure 39 New Sensor is running.

Or if file setup isn’t used, simply type the address in the “Sensor Address” text field. In this case 247.

Now type the new address in the text field “New Sensor Address” and push the button “Change address”.

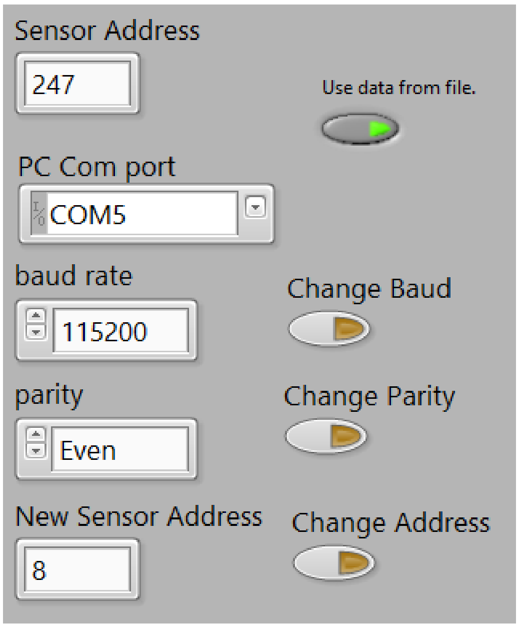


Figure 40 Change Address

Before the change no data was coming for Thor. See Figure 41.

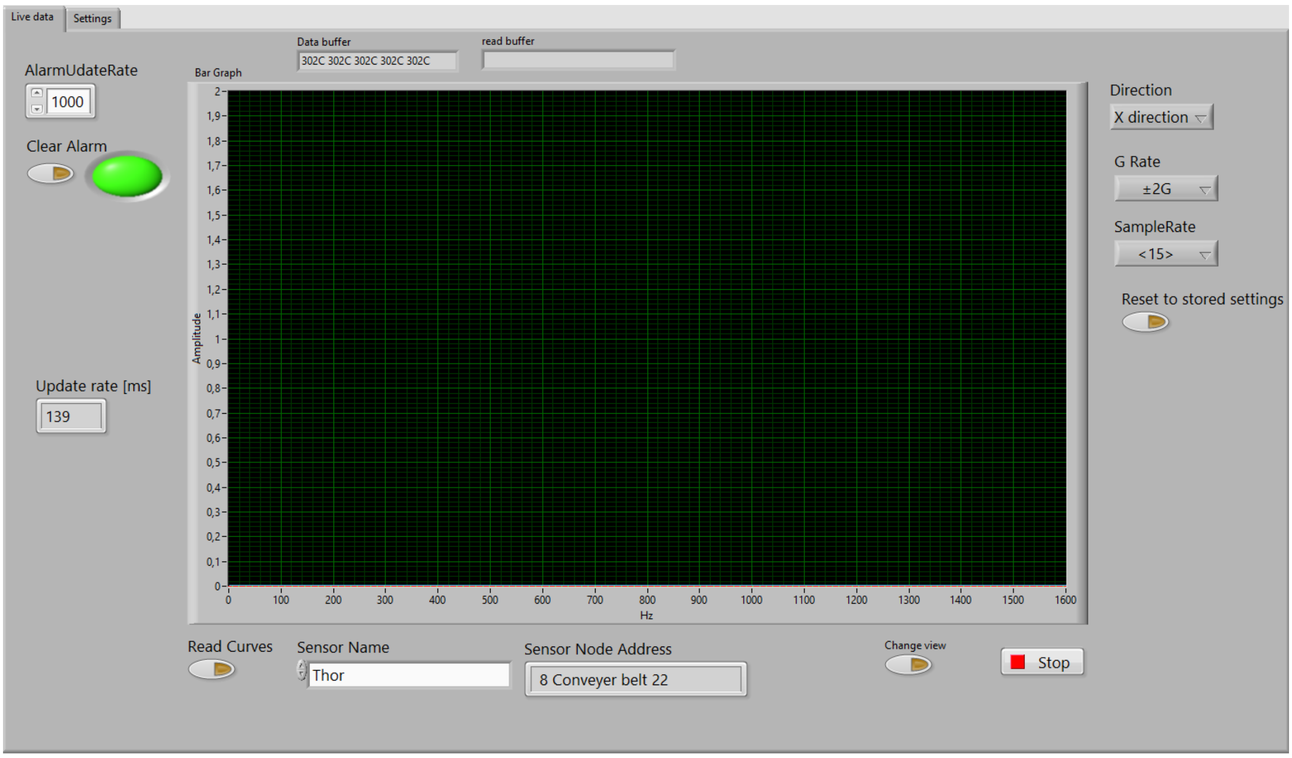


Figure 41 Thor sensor before address change.

Afterwards data is streaming.

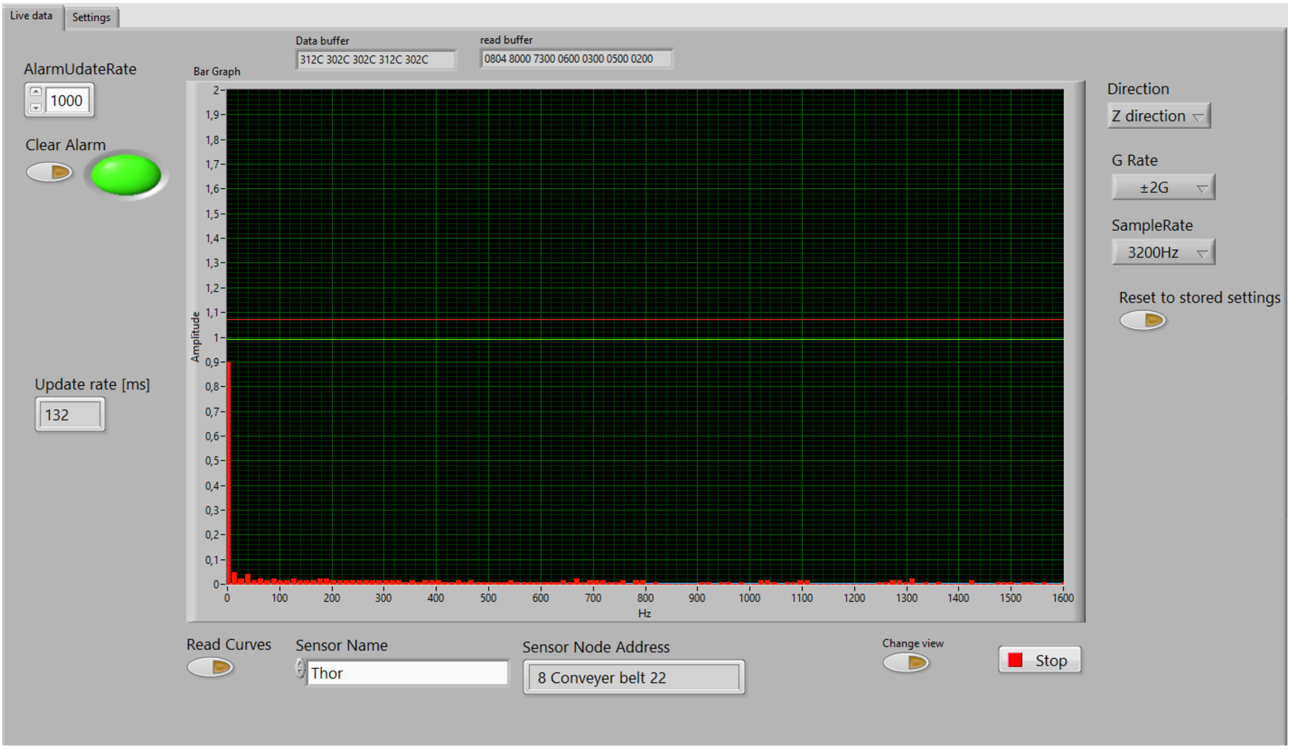


Figure 42 Data is streaming from Thor.

Error handling.

If the system says No COM port found, see Figure 43, no serial communication port is found.

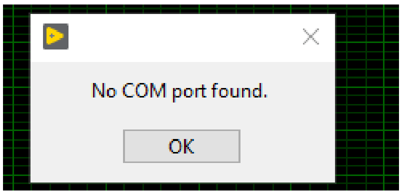


Figure 43 No COM port

This is probably because no RS485 to serial converter is attached to the system.

More to come.

Modbus Protocol Description

In the following the modbus interface for KPV200 is described.

The physical interface between PC, PLC or another controller and KPV200

Default settings.

Physical media	RS485
Protocol	Modbus RTU
Modbus address	247
Baudrate	115200
Databits	8
Stopbits	1
Parity	Even
Timeout	100ms

Modbus messages supported

The following Modbus messages shall be supported.

3 Read holding

4 Read input

16 Write multiple registers

Register 30501 to 30515 must only be used for setup. The data are stored in the internal Eeprom.

If dynamic change is needed where the controller is shifting, example given, directions the dynamic registers must be used. The registers are named 30701 to 30707.

The reason for this is that an eeprom has limited write and erase cycles.

The protocol.

FUNCTION CODE 03 - READ Holding REGISTERS								
Register	Data Address	Description	Data Type	Value range	System Value	Factory settings	Destination	Read write mode.
30001	0	Known curve First FFT BIN	Unsigned int	0-255		127	Eeprom	Read
…	…							
30128	127	Known curve Last FFT BIN	Unsigned int	0-255		127	Eeprom	Read
30129	128	Alarm curve First FFT BIN	Unsigned int	0-255		137	Eeprom	Read
…	…							
30256	255	Alarm curve Last FFT BIN	Unsigned int	0-255		1.0	Eprom	

FUNCTION CODE 03 - READ HOLDING REGISTERS (Holding registers, Settings) Eeprom registers FUNCTION CODE 16 – WRITE MULTIPLE REGISTERS (Holding registers, Settings)								
Register	Data Address	Description	Data Type	Value range	System Value	Factory settings	Destination	Read write mode.
30501	500	Spare						
30502	501	Spare						
30503	502	Spare						
30504	503	Spare						
30505	504	Firmware version				1.0		
30506	505	Baudrate	Unsigned int	Value= baudrate 96 = 9600 192 = 19200 384 = 38400 576 = 57600 1152 = 115200		115200	Eeprom	Read / write

30507	506	Modbus Address	Unsigned int	1 to 247		247	Eeprom	Read / write
30508	507	Parity	Unsigned int	0 = No parity 1 = Odd parity 2 = Even parity		Even	Eeprom	Read / write
30509	508	Spare				-		
30510	509	Spare				-		
30511	510	Direction. Choose X, Y or Z direction.	Unsigned int	0 = X 1 = Y 2 = Z		Z direction.	Eeprom	Read / write
30512	511	Samplerate	Unsigned int	10 = 100 Hz 11 = 200 Hz 12 = 400 Hz 13 = 800 Hz 14 = 1600 Hz 15 = 3200 Hz		3200Hz	Eeprom	Read / write
30513	512	G-Rate	Unsigned int	0 = ±2G 1 = ±4G 2 = ±8G 3 = ±16G		±2G	Eeprom	Read / write
30514	513	AlarmOffset	Unsigned int	0 - 255		10	Eeprom	Read / write
30515	514	AlarmDelay	Unsigned int	0 - 65535		0	Eeprom	Read / write

FUNCTION CODE 03 – READ HOLDING REGISTERS (Holding registers, Settings) Dynamic registers FUNCTION CODE 16 – WRITE MULTIPLE REGISTERS (Holding registers, Settings)								
Register	Data Address	Description	Data Type	Value range	System Value	Factory settings	Destination	Read write mode.
30701	700	Clear Alarm	Unsigned int	1	0	0	Code	Write
30702	701	Reset to stored settings	Unsigned int	1	0	0	Code	Write
30703	702	Start Learning	Unsigned int	1 = Start Learning		0	Code	write
30704	703	Stop Learning	Unsigned int	0 = Stop Learning		0	Code	write
30705	704	Change Direction	Unsigned int	0 = X 1 = Y 2 = Z		Z direction.	Code	Read / write
30706	705	Change Samplerate	Unsigned int	10 = 100 Hz 11 = 200 Hz 12 = 400 Hz 13 = 800 Hz 14 = 1600 Hz 15 = 3200 Hz		3200Hz	Code	Read / write
30707	706	Change G-rate.	Unsigned int	0 = ±2G 1 = ±4G 2 = ±8G 3 = ±16G		±2G	Code	Read / write

FUNCTION CODE 04 – READ INPUT REGISTERS								
Register	Data Address	Description	Data Type	Value range	System Value	Factory settings	Destination	Read write mode.
40001	0	Vibration Data First Bin	Unsigned int	0-255	0	0	Code	read
40128	127	Vibration Data Last Bin	Unsigned int	0-255	0	0	Code	read

40129	128	Alarm Flag	Unsigned int	0 to 1	0	0	Code	read

Versions

Version	Date	Description
KPV200 user guide	31/03-20	Initial version.
KPV200 user guide 1.1	03/04-20	Hardware setup and more descriptions added
KPV200 user guide 1.2	07/08-20	Modbus protocol added.
KPV200 user guide 1.3	27/08-20	Frontpage updated.