RHEONIK

RHE4X
Desktop Reference
Precision Flow Analysis

GET FLOW MEASURED

Rheonik Messtechnik GmbH
Rudolf-Diesel-Straße 5
D-85235 Odelzhausen
Germany

Tel + 49 (0)8134 9341-0
info@rheonik.com

LEARN
MORE

Document No.: 8.2.1.29
Version 1.00
October 2020

RHE4X
Transmitter

Addendum Desktop Reference Precision Flow Analysis

RHEONIK.

Table of Contents

Baseline reference:

Version 0.10 of this document reflects the properties of the RHE4X firmware version 2.04 and later. The corresponding version of the RHEComPro program is 3.10 or later.

Version 0.20 of this document reflects the properties of the RHE4X firmware version 2.05 and later. The corresponding version of the RHEComPro program is 3.11 or later.

1 Introduction

The ordering option FT, Ultra-Fast Signal Tracking, is intended for customers who need very fast response time for their flow measurement application. As described in the RHE4X Desktop Reference Manual, document number 8.2.1.14, the related digital signal filtering schemes may require some tuning tailored to the installation requirement.

Such a tuning is best based on data sampled at the installation site. The Precision Flow Analysis features provides a means to sample mass flow data as a time series of mass increment values down to a resolution of 0.25 ms which corresponds to a recording frequency of 4 kHz when the Prism filtering is active. For the standard Zero-Crossing evaluation methods the resolution corresponds to a period resp. the frequency of the RHM oscillation.

It is expected that the behavior of valves, injection nozzles, and other portioning devices including any mechanical or electrical disturbances can be analyzed with the help of the Precision Flow Analysis. The RHE transmitter will sample flow data in form of mass increments into a wrap-around buffer which then can be read-out and stored in a CSV file e.g. by the RHEComPro program. This CSV file can be then analyzed offline on a standard PC.

This document describes the Precision Flow Analysis Feature on several detail levels from user handling up to the level needed to write own software handling the potential large amount of data. This does not seem necessary since the RHEComPro program is able to perform standard operations with the sample data like starting or stopping a sampling and writing a CSV file from the data sampled.

This document contains three major sections:

- Principles of the Precision Flow Analysis feature
- User Interface to the Precision Flow Analysis feature
- Detailed Description of the Software Interfaces

The first section describes the principles of the Precision Flow Analysis Feature and should be read by users who intend to employ this feature. Not all technical details mentioned in there need to be understood because all related and derived facts are summarized in clear statements.

The second section describes the handling of the Precision Flow Analysis Feature with the help of the RHEComPro program. This is the easiest way to handle the potentially large amount of data generated by the Precision Flow Analysis Feature and therefore, this section should be read by users who want to handle the recorded data on a PC and for whom the functionality of the RHEComPro program is sufficient.

The third section describes the software interfaces of the Precision Flow Analysis in detail and is recommended only to users who need to write their own software to handle the recorded data. This section is complemented by the appendices which contain the data structures needed for the implementation ready to be used for the C# programming language.

2 Principles of the Precision Flow Analysis Feature

The Precision Flow Analysis feature returns the best information when based on the Coriolis phase difference measurement performed by the RHE4X Prism filters. This method is enabled when the Modbus holding register PhsDSPMethod (0x636C) contains the value 2 and bits 2 and 3 are set in the AssurancePresent (0x6090) register.

In this case the Coriolis phase difference is calculated every 0.25ms and converted into a mass increment in kg units and stored as sample in the wrap-around buffer when the Precision Flow Analysis feature is running. The Prism methods, 2 and 3 in PhsDSPMethod, have the advantage of a constant delay of around 20ms for small RHM types. This can be further reduced to 10ms or less for special environments.

For the zero-crossing evaluation methods, 0 or 3 in PhsDSPMethod, the sampling is done once in a RHM oscillation cycle. The sampling rate depends on the type of the RHM and usually lays in the range of 4 to 10ms for smaller RHM types.

The samples are written into a wrap-around buffer of at least 12000 entries in size when the Precision Flow Analysis feature is activated. The samples can be read out in a sequential manner. When samples are read their buffer entries can be overwritten by new samples. When read fast enough, e.g. via Modbus TCP/Ethernet, ordering options EA or EB, there is no time limit for a recording of the samples. For slower data transfers, e.g. via serial Modbus RTU, the recording time span is guaranteed to be at least 3 seconds. This is considered long enough in order to capture anticipated events.

In order to save transfer bandwidth only the mass increment (or – optionally – the phase difference or pickup data) samples are transmitted in larger Modbus telegrams (see “Read Precision” below). The respective time stamps must be reconstructed out of additional information passed in the telegrams related to the Precision Flow Analysis feature.

At the start of the data recording the “Start Precision” telegram sets the time stamp to be used for the first sample of the recording. This time span is a 64-bit value based on a 100ns unit. This data format is used by the .NET programming environment and is called “Tick” there. Usually the current date and time encoded in this format is sent to the RHE transmitter. This allows a synchronization of the Precision Flow Analysis recording with other recordings. A precise synchronization has to take the RHE-internal filter delay into account.

The RHE transmitter returns a related time stamp with each “Read Precision” response telegram and a time increment in the same 100ns unit as the time stamp. The time stamp specifies the time of the first sample returned and the time increment has to be added to it for each following samples in order to obtain a valid time stamp for these samples.

Due the nature of the time increments time stamps can be calculated up to the point where the internal wrap-around buffer overflows and a discontinuity in the time intervals would occur. Therefore, the recording is stopped by the RHE transmitter when a buffer overflow occurs. When this occurs the Modbus input register PrecisionStatus (0x0x4048) returns 2. The recording also can be stopped by a “Precision Stop” telegram. In this case the value in the PrecisionStatus register is set to 0. In both cases values may still be readable from the wrap-around buffer until a non-full sample packet “Read Precision” is received.

The table below list the Modbus registers relevant to the Precise Flow Analysis feature.

Modbus Address	Register Name	Description
0x636C	PhsDSPMethod	(RHE4X) **DSP Method for Phase Difference Calculation:** DSP Method employed: 0: Zero crossing. 1: Prism precision. 2: Prism for fast filling (4kHz). 3: Prism for fast filling (ZC). Values 1 to 3 are only available if Bit 2 in AssurancePresent (0x6090) is set.
0x6090	AssurancePresent	**Assurance Status Present:** (RHE2X) Bitset determining the functions available in the firmware. Bit 0: Assurance Status supervision present in the firmware when set. Bit 1: (RHE4X) Recording Feature present in the firmware when set. Bit 2: (RHE4X) Fast Filling/Prism Algorithm present in firmware. Bit 3: (RHE4X) Precision Flow Analysis feature present in firmware.
0x60D6	PrecisionMode	(RHE4X) **Recording Mode for the Precision Analysis** Bit 0: 0=Use unfiltered phase data (default), 1=employ the primary phase filter. Bits 1-7: 0=Return Mass Data 1= Return Phase Data 2= Return pickup signal from Left Coil 3= Return pickup signal from Right Coil 4= Return filtered signal from Left Coil 5= Return filtered signal from Right Coil 6= Return a triple of left and Right Coil filtered signals and Mass Data
0x4048	PrecisionStatus	(RHE4X) **Status of the Precision Flow Analysis Feature:** 0: Stopped 1: Running 2: Stopped due to buffer overrun

Aside from samples of mass increments in [kg] units it is possible to obtain raw phase difference samples in 8ns units from the Precision Flow Analysis feature. This is controlled by Bit 1 in PrecisionMode, register 0x60D6 (value 2 or 3). The phase differences can be roughly converted to kg mass units by multiplying it with the K-Factor in register 0x6922, MassFlowKFactor, the value 8.0E-9, and and the time increment supplied by the RHE transmitter in the “Read Precision” telegram converted to second unit (divided by 10000000). This, however, will not take the temperature and other correction factors into account.

It also is possible to engage the primary phase difference filter before the samples are taken when Bit 0 of the PrecisionMode is set. The Precision Flow Analysis feature thus offers the possibility to observe the effects of the primary phase filter as a basis for its optimization. Bits 1 to 7 in PrecisionMode specify the data type to be returned. The data types 2 to 6 are intended for diagnosis purposes. Customers mainly will be interested in the mass data, data type 0.

The contents of the register PhsDSPMethod, 0x636C, determines the employed digital signal filtering methods and thus also the sample rate of the Precision Flow Analysis feature. A value 0 in this registers specifies the standard IIR filtering and zero-crossing evaluation which will return one sample per RHM oscillation cycle. A value 2 in this register engages the fast-response Prism filtering combined with a Prism evaluation stage which offers a 4 kHz (0.25 ms) sampling rate.

3 User Interface to the Precision Flow Analysis Feature

This section is intended for users who want to use the RHEComPro program to handle the Precision Flow Analysis feature and generate a CSV file out of the data samples.

The user interface of the RHEComPro program is documented in RHEComPro Suite User Manual, document number 8.2.1.18. Here only the details relevant for the Precision Flow Analysis are described.

In the following subsections the preparation as well as the start and the stop of the data sampling is described followed by a description of the stored CVS files and some hints for the handling of the data.

3.1 Preparing the Recording of the Samples

Before you start the recording of the samples acquired by the Precision Flow Analysis feature you should check the location where the respective CSV file will be written to. Since this is the same location used by the RHEComPro data recording, the respective configuration should be checked using the Configure Data Logging dialog box. This dialog box is activated via the “Data” / “Data Logging Configuration” menu item. The activated dialog box is shown in Figure 1.

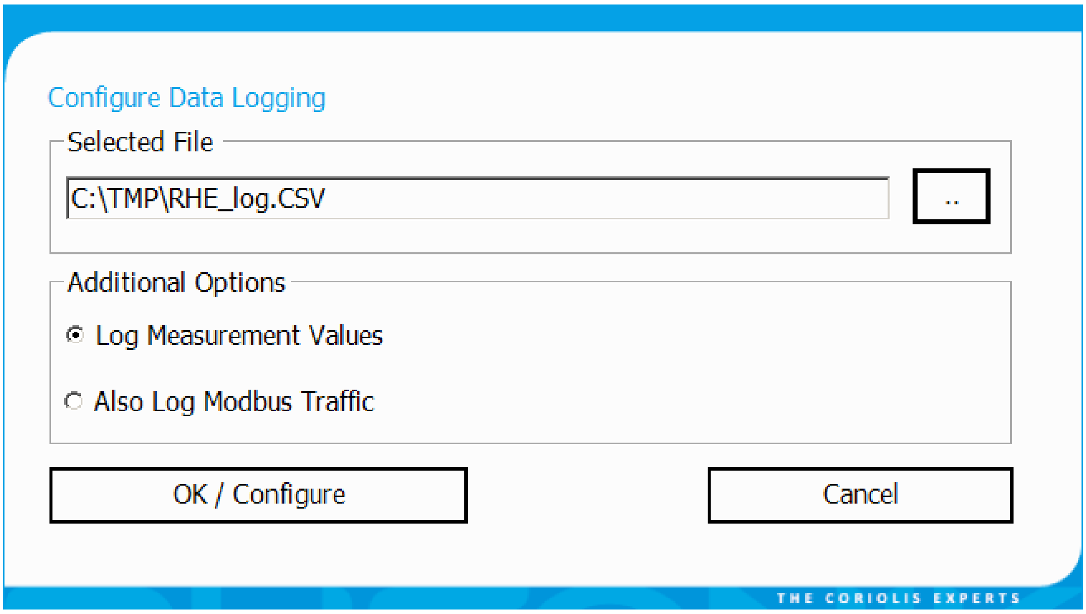

Figure 1: RHEComPro Configure Data Logging

Each time a recording is started, the current date and time is appended to the specified file name. The other options are not used by the Precision Flow Analysis feature.

The resulting file is written in CSV-format and is suitable for import and viewing in a PC program such as Microsoft Excel. Since the German CSV format is used the delimiting “;” characters and the decimal commas must be replaced with the help of a text editor by characters conforming to local conventions when the result is to be processed by a local Excel version.

3.2 Starting the Recording

When an RHE4X series transmitter with enabled Precision Flow Analysis feature is attached to the RHEComPro program it makes the Precision Flow Analysis dialog box available via the "Data" / "RHE4X Data Logging and Precision Flow Analysis" / "Precision Flow Analysis" submenu. Selecting this submenu item starts the dialog box shown below.

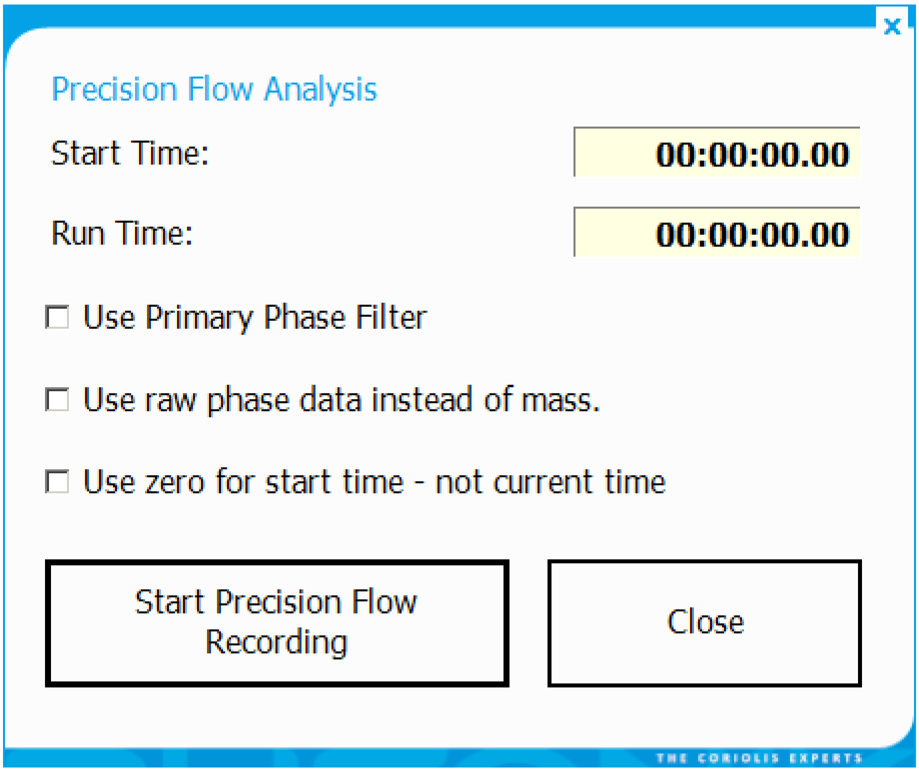

Figure 2: Precision Flow Analysis Dialog Box before the Start of Recording

The dialog box offers a small set of options. The first option "Use Primary Phase Filter" causes the samples to be taken after the primary phase filter in order to observe the effects of this filter. By default the samples are taken directly out of the evaluation stage of the digital signal processing pipeline.

The option "Use raw phase data instead of mass." should be chosen by experience users only.

Normally the time stamps in the CSV file will be Excel date numbers with a precision down to 100ns. These stamps usually must be converted to seconds unit before the can be used in diagrams. The date number has the advantage that events recorded elsewhere can be synchronized to the recording data. For the immediate use in diagrams the option "Use zero start time …" will cause the time stamps written in seconds units with a 100ns resolution.

Pressing the "Start Precision Flow Recording" button will start the recording and the dialog box changes as shown below.

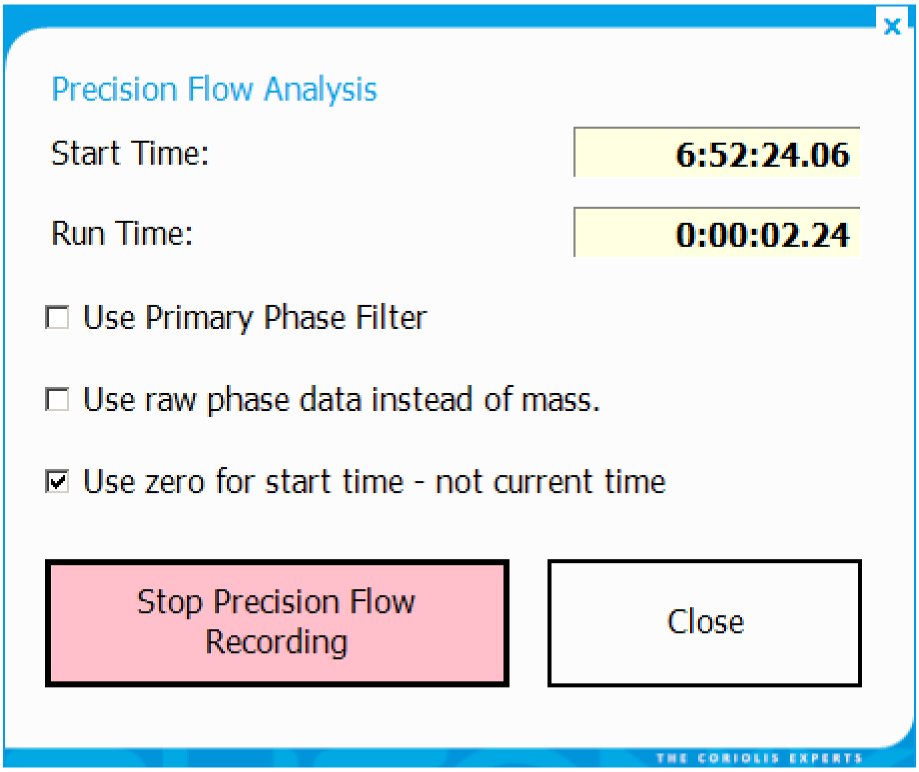

Figure 3: Precision Flow Analysis Dialog Box after the Start of Recording

During the recording the start time and the time progress of the data written into the file is displayed. When the recording is stopped via the button "Stop Precision Flow Recording" remaining data still may will read from the RHE transmitter for a small while. Should the data transport lag too much behind the sampling the RHE-internal wrap-around buffer overflows and the recording is stopped automatically. The size of the wrap-around buffer guarantees a recording time of at least 3 seconds.

For an unlimited recording time the use of a Modbus TCP/Ethernet connection is recommended. This connection should not be influenced negatively by external data traffic.

3.3 Description of the Generated CSV File

The CSV starts with a line containing a readable start time in ISO-Format followed by a two-column header line. The exact contents of the header depends on the selected option. The example below shows the first three lines of the CSV-file displayed by a text editor when the "Use zero start time …" option has been activated only:

```
Date: 2019-06-20 16:24:48
time [s];Unfiltered Massincrement [kg]
0,00000000; 3,044534E-08
```

The third and all following lines contain pairs of time stamp and sample values. As mentioned above the CSV file is written in a format fit for German Excel installations. Before it can be opened by Excel following the conventions of other countries it must be converted by a text editor. Usually the German decimal comma has to be replaced by a decimal point first before the semicolon is replaced by a comma or similar character.

When the date time stamp format is chosen, here the first three lines of a respective CSV file,

```
Date: 2019-06-24 15:12:55
date/time [d];Unfiltered Massincrement [kg]
43640,6339768479; -1,359446E-09
```

the date and the time information may be made visible by formatting the first column with the custom format string “JJJJ-MM-TT hh:mm:ss,000” for German Excel versions or “YYYY-MM-DD hh:mm:ss.000” for English Excel versions. The example number `43640,6339768479` above will be converted to

```
2019-06-24 15:12:55,600
```

Assuming that the start time stamp of an interesting time series is located in the Excel cell A3 a different column can be filled with time offsets in seconds unit by setting the first offset to

```
=(A3-$A$3)*60*24*24
```

and then pulling this field down the respective column as far as it needed.

Out of these time series diagrams may be generated which show periodic an aperiodic events clearly. The effect of disturbances can be studied and to be used to optimize processes. Periodic signals can be determined by Fourier or similar analysis. Filtering the information will show low-frequency disturbances.

4 Detailed Description of the Software Interfaces

This section is intended for persons who want to write proprietary software in order to access the Precision Flow Analysis feature of the RHE transmitters. It also represents a documentation to Rheonik software developers who are not yet familiar with this features.

Since section 2 already provides an extensive description of the operation of the Precision Flow Analysis feature this section concentrates on the details relevant to programmers.

4.1 Modbus Registers

See section 2 for a detailed description of the Modbus registers related to the Precision Flow Analysis feature.

4.2 RHE4X Precision Commands

The RHE4X Precision Commands are implemented as proprietary Rheonik Modbus Commands with the command code 0x72. The byte below the Modbus command code contains subcommands which constitute a whole set of commands reserved for the use in Rheonik transmitters.

All data items within the Modbus Commands described here are stored in "little-endian" format and thus do not be swapped when read or written from a PC.

4.2.1 "Precision Start" Subcommand

The "Precision Start" command starts the data sampling. The command has following format:

Byte	Meaning	Value
0	Modbus ID	1
1	Rheonik Command Code	0x72
2	Rheonik Subcommand "Precision Start"	40
3 .. 10	Time Stamp 100ns units (64-bit integer)	
11	CRC low (not for Modbus TCP)	
12	CRC high (not for Modbus TCP)	

The time stamp in this record is expected as 8-Byte .NET tick in 100ns resolution (e.g. taken from C# DateTime.Now.Ticks). This time stamp will be associated with the first sample returned by the Precision Read Command following the Precision Start Command.

Following response is expected:

Byte	Meaning	Value
0	Modbus ID	1
1	Rheonik Command Code	0x72
2	Rheonik Subcommand "Precision Start"	40
3	Precision Mode	
4	CRC low (not for Modbus TCP)	
5	CRC high (not for Modbus TCP)	

The value returned in the "Precision Mode" field is the value found in the Modbus Register PrecisionMode (0x60D6). The operating mode defined there is valid for the data sampling until a stop command occurs or the internal Precision Flow Analysis buffer overruns which also cause a stop of the data sampling.

4.2.2 "Precision Stop" Subcommand

The "Precision Stop" command stops the data sampling. The command has following format:

Byte	Meaning	Value
0	Modbus ID	1
1	Rheonik Command Code	0x72
2	Rheonik Subcommand "Precision Stop"	41
3	CRC low (not for Modbus TCP)	
4	CRC high (not for Modbus TCP)	

After the Precision Stop command has been issued remaining samples still can be read-out of the wrap-around Precision Flow Analysis buffer until the Precision Read command returns less than 50 samples.

Following response is expected:

Byte	Meaning	Value
0	Modbus ID	1
1	Rheonik Command Code	0x72
2	Rheonik Subcommand "Precision Stop"	41
3	CRC low (not for Modbus TCP)	
4	CRC high (not for Modbus TCP)	

4.2.3 "Precision Read" Subcommand

The "Precision Read" command reads a portion of the data samples stored on the Precision Flow buffer. Data samples can be read in a sequential order only. After a Precision Start the data are read from the start of the Precision Flow Analysis buffer. The next read returns the samples following the last read sample and so forth. The Precision Read command has following format:

Byte	Meaning	Value
0	Modbus ID	1
1	Rheonik Command Code	0x72
2	Rheonik Subcommand "Precision Read"	42
3	CRC low (not for Modbus TCP)	
4	CRC high (not for Modbus TCP)	

Following response is expected:

Byte	Meaning	Value
0	Modbus ID	1
1	Rheonik Command Code	0x72
2	Rheonik Subcommand “Precision Read”	42
3	Status	
4 .. 11	Timestamp in 100ns units (64-bit integer)	
12 .. 15	Time Increment per sample (32-bit IEEE float)	
16 .. 17	Sample Count returned, may be 0 (16-bit integer)	N
18 .. 217	Up to 50 Samples returned in 32-bit IEEE floating point format. Sample with indices equal or beyond the value in the Sample Count field are to be ignored. The unit of a sample is [kg].	
18 + 200	CRC low (not for Modbus TCP)	
19 + 200	CRC high (not for Modbus TCP)	

The Status field contains a copy of the current value in PrecisionStatus (0x4048). This value should be checked only when Sample Count is smaller than 48. In this case a value of 0 means that the data sampling has been stopped and no more data can be read. A value of 2 means that a buffer overrun has occurred and no more data can be read. A value of 1 indicates that the Precision Flow data sampling still is active and that the Precision Read command should be repeated to retrieve further samples.

The Timestamp specifies the time for the first sample returned in the buffer. The first sample after a Precision Start command gets the Timestamp specified in the Precision Start command. The Time Increment field specifies the time to be added for each additional sample. Both items are in 100ns units. It is recommended to keep a separate time increment sum in a floating point format that is added to the time stamp in order to calculate the reference time because the time increments may contain timing portions smaller than the 100ns units when derived from the RHM oscillation.

When Bits 1 to 7 in PrecisionMode specify the data type 6 a sample triple is returned for each timestamp in the sequence Mass Increment, Filtered Left Coil, and Filtered Right Coil Samples. The total sample count returned in a “Precision Read” response will be divisible by 3 with a maximum of 48 samples. Due to the larger amount of data to be transferred it is unlikely that the data can be read in real time via Modbus TCP/Ethernet. However, a recording time span of at least one second is guaranteed by the RHE4X-internal buffer.

Appendix

Appendix A C# Structure used for the Precision Flow Analysis Feature

This section contains the structure of the response of “Precision Read” command. The layout if this structure is defined with the help of the C# *[FieldOffset(x)]* feature which defines the alignment of the data. Since all data is transferred in little-endian byte format no byte swapping is necessary when processed on a standard PC.

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Runtime.InteropServices;

…

    [StructLayout(LayoutKind.Explicit, Size = 218  )]
    public struct precision_record_layout
    {
        /*
         * byte_array is intended to be used in an "unsafe"
         * copy loop to fill the record, e.g.:
         *      unsafe
         *      {
         *          fixed (byte * dest = &record.byte_array)
         *          for (int i = 0; i < 218; i++)
         *          {
         *              dest [i] = buf [i];
         *           }
         *       }
         */
        [FieldOffset(0)]
        public byte byte_array;

        [FieldOffset(0)]
        public Byte Modbus_Id;

        [FieldOffset(1)]
        public Byte Rheonik_Command_Code;

        [FieldOffset(2)]
        public Byte Rheonik_Subcommand_Code;

        [FieldOffset(3)]
        public Byte Precision_Status;

        [FieldOffset(4)]
        public UInt64 Timestamp;       /* In 100ns units */

        [FieldOffset(12)]
        public float TimeIncrement;    /* In 100ns units */

        [FieldOffset(16)]
        public UInt16 Sample_Count;

        /*
         * Samples is intended to be used in an "unsafe"
         * access loop, e.g.:
         *      unsafe
```

```
         *      {
         *          fixed (float * source = &record.Samples)
         *          for (int i = 0; i < record.Sample_Count; i++)
         *          {
         *               myStreamWriter.WriteLine(timestamp.ToString("F8") + "; " +
         *                                         sample[i].ToString("G7"));
         *               timestamp += (double)record.TimeIncrement / 10000000.0;
         *          }
         *       }
         */

        [FieldOffset(18)]
        public float Samples;           /* Array of 50 Samples */

        /* CRC not present for Modbus TCP */
//      [FieldOffset(18+200)]
//      public Byte CRC_low;

//      [FieldOffset(19+200)]
//      public Byte CRC_high;

    }
```

About Rheonik

Rheonik has but one single purpose: to design and manufacture the very best Coriolis meters available.

Our research and engineering resources are dedicated to finding new and better ways to provide cost effective accurate mass flow solutions that provide value to our customers. Our manufacturing group care for each and every meter we produce from raw materials all the way to shipping, and our service and support group are available to help you specify, integrate, start-up and maintain every Rheonik meter you have in service. Whether you own just one meter or have hundreds, you will never be just another customer to us. You are our valued business partner.

Need a specific configuration for your plant? Don't compromise with a "standard" product from elsewhere that will add extra cost to your installation. If we can't configure it from our extensive and versatile product range, our exclusive **AnyPipeFit Commitment** can have your flow sensor customized with any size/type of process connection and face to face dimension you need.

No matter what control system you use as the backbone in your enterprise, with our **AnyInterface Commitment**, you can be sure that connection and communication will not be a problem. Alongside a wide variety of discrete analog and digital signal connections, we can also provide just about any network/bus interface available (for example: HART, ProfibusDP, ProfiNet, EtherCAT, PowerLink, EtherNet/IP, CAN, ...) with our RHE 40 Series family of transmitters. Rheonik RHE 40 Series transmitters can connect to your system – no headache and no conversion needed.

Rheonik Messtechnik GmbH
Rudolf-Diesel-Straße 5
D-85235 Odelzhausen
Germany

Tel + 49 (0)8134 9341-0
info@rheonik.com