Class II
Item no.:0004-3134
2010-01-25

Vestas
Serial Test Interface Protocol (V112)

History of this Document

Rev.	Date (dd mmm yyyy)	Changed by;	Description of changes (including affected section numbers)	Reviewed by; Date;	Approved by; Date;
D0.1	22-06-2009	RAVIS	First version		

Template rev. R3

Contents

1. Introduction

1.1 Purpose

This document describes the Data Streaming functionality in Vestas turbines, intended for performance measurements.

1.2 Reading Guidelines

Special words/terms and abbreviations are explained in section 3.2 Word/Abbreviation List.

1.3 Summary

The data streaming is a continues output of data from the turbine. The transmission is a one way transmission from the turbine control system to the ”measurement equipment”. The data is transmitted through a DSUB 9-pin RS-485 connection supplied by Vestas.

2. Protocol description

2.1 Overall description

Data is transmitted from the turbine with the configured rate, in messages with real-time values for each channel in IEEE single precision floating point format and header information as ASCII characters.

A stand alone program is available for simulating the data stream on a computer. The program can stream simulated messages to a virtual serial port, or to actual serial ports on the computer.

2.2 Configuration

The data streaming is configured by Vestas.

2.2.1 Setup

The interface uses the following default transmission settings:
Hardware interface: RS485 2-wire
Hardware handshake: None
Transmission rate: 57,6; 115,2; 230,4; 460,8; 921,6 Kbit/s
Character setup: 1 startbit, 8 databit, 1 stopbit
Parity: No (check on message level: CRC16)

2.3 Message information

Each message includes a timestamp, number of channel info, the data, tail information and a 2 byte CRC. Each telegram has the following structure (The individual sections will be described in the following sub-chapters):

Sync	Version	Time-stamp	Tail in-fo type	No. of. chan-nels	Channel 1 data	Channel 2 data	…	Channel n data	Tail info	CRC16
5 by-tes	1 bytes	23 by-tes	1 byte	2 by-tes	4 bytes	4 bytes		4 bytes	31 by-tes	2 bytes

The transmission order is starting with the version and ending with the CRC.

2.3.1 Synchronization header

To indicate a start of a message the telegram always starts with 5 ‘$’ characters (‘$$$$$’). Since a message can end with a $, the preferred method for message synchronisation is to search for $$$$$ and a character that is not a $ (in this version a B => $$$$$B) .

2.3.2 Version

Each message contains a 1 ASCII character version of the protocol format. This protocol has revision: B.

2.3.3 Timestamp

Each message contains a timestamp. The timestamp is the sample time from the wind turbines in local time. It is a 23 bytes ASCII timestamp ex.: **2008-11-06T11:11:11.111**

2.3.4 Tail Info Type

Each message contains one “1” ASCII character to indicate tail information.

2.3.5 Number of channels

Each message contains “Number of channels” information. These 2 ASCII characters inform on the following number of measurement channels.

2.3.6 Data

All data from the turbines are transformed into 32 bit single precision floating point numbers that conforms to the IEEE 754-1985 standard [1].
The order of the data is the one obtained by using the host-to-network byte order function “htonl()” (Big endian) . When receiving the data one must use the opposite network-to-host byte order function “ntohl()” in order to have the data ordered correctly on that given platform.
If turbine data is invalid, then NaN [1] is streamed as data. NaN is defined as the value 0x7FC00000.

2.3.7 Tail information

The “tail information block” will contain information on only one channel and the turbine controller will subsequently send information on the individual channels contained in the message.

Channel number: First 2 characters are ASCII channel number. The channel numbers start at ‘1’ and are always 2 characters. (e.g. the first channel would be represented with ‘01’)

Short name: Informative unique name for the channel of 20 ASCII characters (filled up with spaces). The short name is supplied in the configuration.

SI Unit: The 5 ASCII character long SI unit (e.g. m/s or kVar) (padded with spaces).

ID: A 32 bit unsigned integer that uniquely identifies the signal being listened to. These bits are for internal Vestas use only.

Version	Timestamp	Tail info type	No. of. Channels	Channel 1 data	Channel 2 data	…	Channel n data	**Tail info**	CRC16
1 bytes	23 bytes	1 byte	2 bytes	4 bytes	4 bytes		4 bytes	**31 bytes**	2 bytes

Channel number	Short name	SI Unit	Phoenix ID
2 bytes	20 bytes	5 bytes	4 bytes

2.3.8 CRC16

On each message the integrity and quality is controlled by a 16 bit Cyclic redundancy check (CRC16). The calculation is performed on all message characters (bytes), except the check bytes itself.

The calculation is performed on all message characters (bytes), except the check bytes itself. The low-order byte (Crc_Lbyte) is appended to the message first, followed by the high-order byte (Crc_Hbyte). The receiver of the message calculates the check bytes again and compares them with the received ones. (example using 'C')

```
void main()
{
        unsigned char data[NUMDATA+2]; // Message buffer
        unsigned char Crc_Hbyte,Lbyte; //
        unsigned int Crc;
        ....
        Crc=0xFFFF;
        for (i=0; i<NUMDATA; i++)
        {
                Crc = CRC16 (Crc, data[i] );
        }
        Crc_Lbyte = (Crc & 0x00FF); // Calculate low-order byte
        Crc_Hbyte = (Crc & 0xFF00) / 256; // Calculate high-order byte
}

// CRC16 calculation
// ----------------

unsigned int CRC16(unsigned int crc, unsigned int data)
{
        const unsigned int Poly16=0xA001;
        unsigned int LSB, i;
        crc = ((crc^data) | 0xFF00) & (crc | 0x00FF);
        for (i=0; i<8; i++)
        {
                LSB=(crc & 0x0001);
                crc=crc/2;
                if (LSB)
                crc=crc^Poly16;
        }
        return(crc);
}
```

2.3.9 Message example

$$$$$B2009-07-23T10:58:09.233123¾Þ┬ „?┼ ô¾?Èz_@$=0@d=0@'-˜@²-˜@Ò-˜@ò-˜A
⌂ LA├⌂ LA)⌂ LA9⌂ LAI⌂ LAY⌂ LAi⌂ LAy⌂ LA„‡¦AŒ‡¦A"‡¦Aœ‡¦A¤‡¦A¬‡¦17 Pitch Angle C rad01852¬

3. Appendices

3.1 References

Reference Number	Description
1	Floating point IEEE standard http://en.wikipedia.org/wiki/IEEE_floating-point_standard

3.2 Word/Abbreviation List

Word/ Abbreviation:	Explanation: