

A Data Acquisition Platform for the Development of a Wind Turbine Condition Monitoring System

Grzegorz Swiszc^{1*}, Andrew Cruden¹, Campbell Booth¹ and William Leithead¹

¹ University of Strathclyde, Department of EEE, Glasgow, UK

*E-mail : grzegorz.swiszc@eee.strath.ac.uk

Abstract--Often condition monitoring (CM) system developers use simulation or test rig data due to a lack of actual wind turbine measurements however, this data may not necessarily reflect processes happening in the real turbine application. In order to determine different patterns of machinery deterioration, precise data of existing parameters at higher sampling frequencies up to 20 kHz is needed. Additional parameters, not currently commonly monitored, like turbine vibration or tower movement, are also desired. In this paper the design of a data acquisition (DAQ) platform to be mounted on a 660kW Vestas V47 wind turbine is presented. The requirements and limitations of the data acquisition application are discussed. A detailed explanation of the data acquisition program developed in LabVIEW as well as the possibilities for remotely controlling this process are given. Finally, different methodologies for handling, storing and accessing large volumes of data are presented with the authors' optimal solution highlighted.

I. INTRODUCTION

THE wind turbine market has experienced significant and steady growth over the last few years, as more, and greater capacity, turbines are built [1]. However, there has been little improvement in the way crucial parameters are monitored. It is still common that measurands (usually averaged) are transmitted over the SCADA system to the control room where engineers responsible for operation analyse this data online and determine abnormal behavior. Such a monitoring system can be sufficient for fault detection, but in order to be able to predict faults, automatic real-time data analysis techniques need to be employed. Combining the number of monitored parameters with a system capable of showing gradually developing faults would significantly improve maintenance planning, reducing undesired downtime and keep energy production losses to a minimum. To develop a reliable condition monitoring system detailed analysis of fault origins based on real data has to be carried out. This paper presents details of the DAQ Platform for acquiring raw data suitable for such analysis.

This work has been supported by EPSRC Supergen V – Wind Energy Technologies and ScottishPower.

A. Safety Requirements and Limitations

The design of a DAQ Platform has to meet a number of requirements in order to be accepted by the project industrial partner for installation on an operational 660kW Vestas V47 wind turbine (Fig. 1).



Fig. 1. 660kW Vestas V47 wind turbine, Hare Hill, Scotland

Obviously the DAQ system should not effect the normal operation of the wind turbine and should not pose any danger to maintenance engineers. Preferably, there should not be any electrical connection between devices installed at the top (nacelle) and the bottom of the turbine and all direct electrical connections to the controller must feature an electrical isolation barrier. Control of the data acquisition and collection should be enabled from the bottom of the turbine. Installation engineers should have completed recognised Wind Turbine Climbing [2] and First Aid Courses [3], to be familiar with aspects of safe access, evacuation and rescue procedures. Prior to installation Risk Assessment and Method Statement documents have to be prepared and accepted by the wind farm operator.

B. Measurand Selection

Amongst the range of parameters available in a wind turbine only the most critical for analysis have been selected. Due to the limitations of the DAQ cards, which do not allow sampling rates to be set for individual channels, the chosen parameters were divided into two groups to be sampled at two different sampling rates – low speed and high speed groups. The list of parameters selected for monitoring at low speed i.e. 50Hz is enclosed in Table I. Table II contains parameters to be sampled at high speed i.e. 20 kHz rate.

TABLE I
LIST OF PARAMETERS SELECTED FOR 50 HZ SAMPLING RATE
ACQUISITION [4]

No	Parameter	Output	Connection / Interface
1.	Temperature, nacelle	PT100	Direct to Controller
2.	Temperature, ambient	PT100	Direct to Controller
3.	Temperature, generator	PT100	Direct to Controller
4.	Temperature, gear oil	PT100	Direct to Controller
5.	Temperature, gear bearing	PT100	Direct to Controller
6.	Pitch position	0-10V	Direct to Controller
7.	Rotor speed	Hi-pulse	Controller through F/V Converter
8.	Generator speed	Hi-pulse	Controller through F/V Converter
9.	Wind speed	Hi-pulse	Controller through F/V Converter
10.	Yawing direction - CW	Hi-start	Direct to Controller
11.	Yawing direction - CCW	Hi-start	Direct to Controller
12.	Wind vane - 0 degree	0/24VDC	Controller through Voltage Divider
13.	Wind vane - 90 degree	0/24VDC	Controller through Voltage Divider
14.	Low speed shaft vertical displacement	0-10V	Laser Distance Meter
15.	Low speed shaft horizontal displacement	0-10V	Laser Distance Meter
16.	Tower vibration - XY	0-10V	Structural Vibration Sensor

TABLE II
LIST OF PARAMETERS SELECTED FOR 20 KHZ SAMPLING RATE
ACQUISITION

No	Parameter	Connection / Interface
1.	Voltage L1	1kV Differential Voltage Probe
2.	Voltage L2	1kV Differential Voltage Probe
3.	Voltage L3	1kV Differential Voltage Probe
4.	Current L1	1000A Current Clamp
5.	Current L2	1000A Current Clamp
6.	Current L3	1000A Current Clamp
7.	Vibration 1	Accelerometer
8.	Vibration 2	Accelerometer
9.	Vibration 3	Accelerometer
10.	Vibration 4	Accelerometer
11.	Vibration 5	Accelerometer
12.	Vibration 6	Accelerometer
13.	Vibration 7	Accelerometer

The layout of the vibration and displacement sensors is shown in Fig. 2. There are a total of seven vibration sensors attached to the generator, gearbox and main bearing in both axial and radial directions. Sensors have been chosen to minimise interference with existing equipment by using adhesive fixings instead of screwed fixings. They are able to detect frequencies over a range from 0.8 Hz to 15 kHz ± 3 dB with sensitivity of 100mV/g. There are also two laser distance meters mounted in a vertical and horizontal position to measure the low speed shaft displacement.

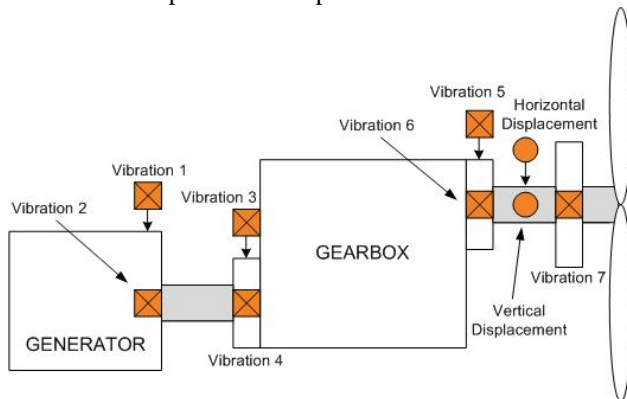


Fig. 2. Vibration and displacement sensors layout.

C. Large Volume Data Storage Techniques

The selection of appropriate data storage techniques and determination of the possible volume of data to be gathered is a very important issue of the DAQ project design. Four possible methods of storing data have been considered as shown in Table III. Usually the file format of choice for data acquisition is ASCII text, because text files are easy to save and read and they can be easily opened by other software e.g. Matlab, spreadsheet or word processor. However difficulties arise when there is a need for storing large volumes of data. ASCII is a seven-bit code, meaning it uses patterns of seven binary digits (a range of 0 to 127 decimal) [5] to represent each character, making it space consuming.

TABLE III
CHARACTERISTIC OF DIFFERENT STORAGE TECHNIQUES FOR
LARGE DATA VOLUME

	Access to any part of the file (table)	The example volume of data	File (table) size limit	Comments
ASCII Text file	Poor: whole file has to be loaded to memory first (slow)	15.6MB	2GB	Easy to handle, but poor accessibility and bulky
Binary file		9.4MB	2GB (practical)	Compact but poor accessibility and difficult to handle
MySQL InnoDB table		16.3MB	2TB (Win32 NTFS)	Fairly easy to handle, good accessibility, but bulky
MySQL MyISAM table	Good: access through SQL query to any part of the data	7.8MB	2TB (Win32 NTFS)	Compact, good accessibility and fairly easy to handle

Another important consideration is that the maximum size for a text file in Windows is limited to 2GB [6], hence larger volumes of data have to be stored in a sequence of files. This requires good file organisation and is more difficult to analyse. The whole text file has to be loaded into memory, even if only a selected part of it is to be analysed. Storing each channel of the data in separate text files would be more appropriate, but this would further significantly increase the total number of files saved.

An alternative option to text files are binary files, mainly as they are less space consuming. The drawback is that they cannot be easily opened by programs, nor can they be read by any program without a detailed knowledge of file's format. Similar to text files, whole binary files have to be loaded to the memory prior to reading. As most desktop PCs run out of memory with files of 2GB or more in size, this makes them difficult to handle.

These problems are efficiently solved when using the open source MySQL database [7]. This database offers two types of tables: MyISAM and InnoDB. In either configuration up to 2TB of data can be saved in one table. Another very important feature is that the data can be read without loading the whole table into memory. In this project MyISAM tables were used as they are faster and more compact. In order to communicate with the database from other software, an Open Database Connectivity (ODBC) [8] driver has to be installed.

LabVIEW is not inherently equipped with blocks supporting connection to the MySQL database and such a module is expensive to purchase. In the project, an open source LabSQL [9] by Jeffrey Travis was successfully used instead.

D. Data Acquisition Platform Overview

The heart of the DAQ Platform is a computer PC1 equipped with two DAQ cards placed at the top (nacelle) of the wind turbine as shown in Fig. 3. Each DAQ card has a different maximum sampling rate: the slower 250ks/s USB card has been designated for Table I parameter data acquisition, and the faster 1.25MS/s will gather the Table II voltage, current and vibration data. As the acquisition of existing parameters of the wind turbine is to be done by direct connection to the controller, the DAQ card was chosen with an integral isolation barrier. The maximum analog input voltage this device can accept is $\pm 10V$ hence to decrease the input voltage, for some of the measurements like the wind vane, a voltage divider interface had to be used.

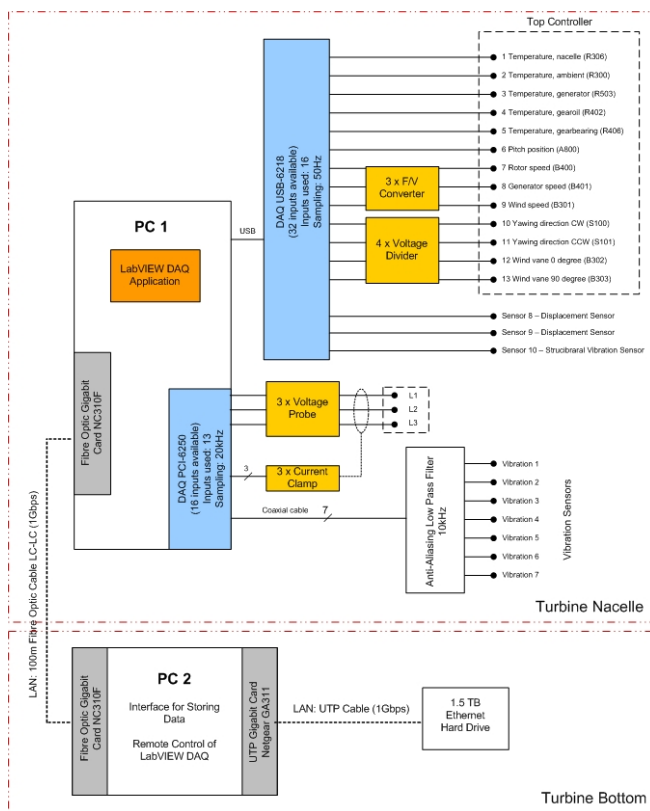


Fig. 3. Data Acquisition Platform layout.

Additionally a frequency to voltage converter was used for the rotor, generator and wind speed inputs to simplify data analysis. The vibration sensors are connected through an anti-aliasing 5th order Butterworth low pass filter with 10 kHz cut-off frequency to match the Nyquist limit for the 20ks/s sampling rate [10]. The nacelle computer is connected to the bottom computer via a 1Gbps fiber optic cable giving both essential electrical isolation between the two systems and the desired fast data transfer rate. As the wind turbine is not

equipped with an Internet link, the bottom computer acts primarily as the data storage medium. A high capacity 1.5TB Ethernet hard drive, featuring a high speed 1Gbps UTP link is connected to the bottom computer. This configuration enables easy hard drive swapping when collecting the captured data. The top to bottom computer and the bottom computer to hard drive, act as two separate Ethernet networks. In order to send data from the top computer to the hard drive, a bridge between these two networks is required. The appropriate configuration of network card IP addresses is presented in Table IV.

TABLE IV
NETWORK CARD IP ADDRESS SETTINGS FOR BRIDGING TWO NETWORKS

	Top Computer	Bottom Computer (bridge)	Hard Drive
IP Address	169.354.1.2	169.254.1.1	169.254.4.254
Mask	255.255.0.0	255.255.0.0	255.255.0.0

E. Data Acquisition Program User Interface

The data acquisition program user interface is shown in Fig. 4. The number of parameters required for setting up this interface is limited to the minimum essential. No knowledge of SQL is required by the user as all MySQL commands are hidden and execute automatically. To configure the program, the number of channels, type of input channels and sampling rates are needed. For the MySQL Table setup the data precision is required, however the ODBC Connection and Table Name are pre-described. In the Current Status area the user can observe current scans being written and the table name. The parameters and current status can be controlled and observed for both DAQ cards via the window tab control. Data acquisition is time controlled in a 24-hour cycle, which means the program starts and stops every day at the same time. For each DAQ card a new table is created every day with the name format beginning with the prefix ‘a’ or ‘b’ for the slow and fast DAQ cards respectively followed by the current date. Each MyISAM table is represented by three files, e.g. a07Dec07.myd, a07Dec07.myi, a07Dec07.frm one containing the data and the other two the table format. The state of the data acquisition is indicated by flashing ‘LEDs’ in top right corner of the window.

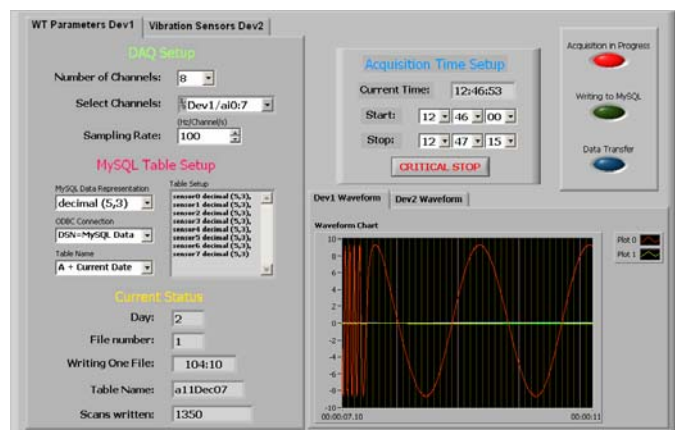


Fig. 4. Data Acquisition Platform user interface.

There is also the capability of remote control of the data acquisition process from the bottom computer by connecting to the running program from LabVIEW “Operate >> Connect to remote panel” or via a web page. LabVIEW server has to be configured on the nacelle computer to enable this connection to be made.

F. Detailed Data Acquisition Program Description

The data acquisition program flow chart is shown in Fig. 5. When the DAQ program is executed and all the parameters are set up it then waits until the user set start time to begin the acquisition cycle. Before the data acquisition starts the program automatically creates two MySQL tables and two temporary folders on the local hard drive. Data acquisition can then be performed simultaneously from both DAQ cards writing data to a sequence of 2GB text files, each card to its own folder.

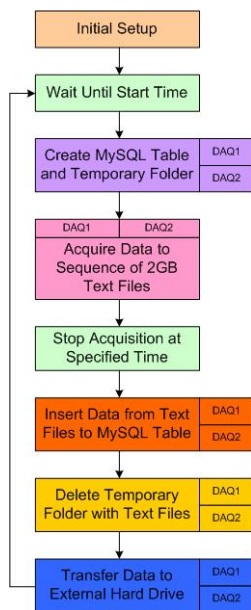


Fig. 5. Data acquisition program flow chart.

As stated earlier the optimal format technique for large volumes of data is as a MySQL table. Direct insertion of data from LabVIEW to the MySQL database using LabSQL works well only for low sampling rates where the data is sent sample-by-sample. For higher sampling rates i.e. 20 kHz the stream of data needs to be buffered, which means samples are first sent to temporary memory and after a specified time, e.g. one second, the whole packet is moved to the storage destination. Unfortunately, buffered data cannot be direct input to MySQL using LabSQL therefore this is solved by temporarily saving data to the sequence of 2GB text files first. After acquisition is stopped the data is then sequentially copied to MySQL tables using SQL commands and all text files are then deleted. The data now in the form of MySQL files is ready for transmitting to the Ethernet hard drive. After this operation is completed the nacelle computer’s hard drive has no data and the program awaits idle for the next acquisition start trigger.

The final configuration of the data acquisition system is: 16 channels sampled at 50Hz and 13 channels sampled at 20 kHz generating 50MB of data per second in MySQL format. The data acquisition time strategy is shown in Fig. 6. With the data being acquired for 12 hours a day, 7GB needs to be transferred to MySQL and then 3.6GB to the hard drive every day which takes 2 hours in total. For this case the amount of data generated monthly is 1.08TB, giving approximately 40 days data collection capability before hard drive recovery and replacement.

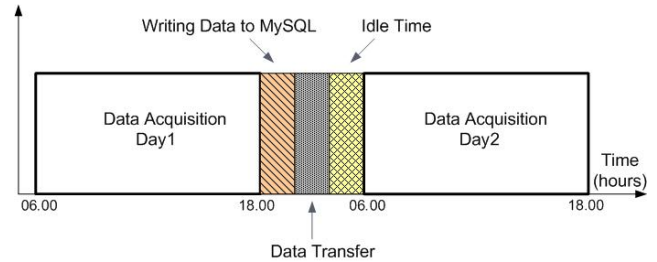


Fig. 6. 24-hour strategy for data acquisition.

G. Conclusions

The authors’ data acquisition platform fulfils all the performance and safety requirements to be accepted by the industrial partner. Results obtained from initial tests show that the DAQ system is an easy to operate source of reliable data. Adding extra sensors does not require program reconfiguration and can be done from the user interface. The program is also designed to easily accommodate a real-time data processing module in the future.

II. REFERENCES

- [1] <http://www.bwea.com/ukwed/index.asp>
- [2] <http://www.narc.co.uk>
- [3] <http://www.redcrossfirstaidtraining.co.uk>
- [4] Vestas V47 660kW Wind Turbine - Electrical Manual
- [5] <http://en.wikipedia.org/wiki/ASCII>
- [6] <http://msdn2.microsoft.com/en-us/library/aa366778.aspx>
- [7] <http://www.mysql.com>
- [8] <http://www.mysql.com/products/connector/odbc>
- [9] <http://jjeffreystavis.com/lost/labsql.html>
- [10] Gary W. Johnson, Richard Jennings “LabVIEW Graphical Programming”, Ed. New York: McGraw-Hill, 2006, p. 306