International Journal of Computer Applications (0975 – 8887)
Volume 51– No.5, August 2012

LabVIEW FPGA based Software Implementation for an Automated Test System of Shafts used in High Lift System of an Aircraft

Md. Shouqatali A.C.
Test Engineer,
MOOG India
Bangalore

Navin Karanth P.
Asst. Professor,
Mech. Dept., NITK
Surathkal, Mangalore

Karthik S.P.
Senior Test Engineer,
MOOG India
Bangalore

Vijay Desai
Phd, Professor,
Mech. Dept., NITK
Surathkal, Mangalore

ABSTRACT
Shafts used in the high lift system (wings of a plane) of an aircraft undergo extreme load conditions during takeoff and landing. Performance of shaft deteriorates along the life span of it. The failure of shaft can lead to a major catastrophe. Therefore, to ensure the safety of passengers, there is a need to develop a test system which can subject different shafts to various loads to which they are designed for and test them for their life cycle. This paper presents implementation of a test system built using LabVIEW – Field Programmable Gate Array (FPGA) which is able to simulate different load conditions on shaft. The real time data of torque and speed values are recorded using FPGA card. Software design of test system and results obtained for a test shaft are discussed in this paper.

General Terms
Automated Test System, LabVIEW FPGA.

Keywords
LabVIEW Virtual Instrumentation, FPGA, Test System/Equipment, Automated Test System.

1. INTRODUCTION
Shafts of high lift system are connected between hydraulic actuators (drive) and Flap/Slat surfaces (load). During flight cycle, actuators drive shaft in one direction and due to wind load, surfaces drive shaft in opposite direction. These two forces create torque all along the length of the shaft. The geometrical design and material of the shaft should withstand this torque and must be able to transmit power from actuators to the surfaces.

A shaft can undergo thousands of flight cycle before failure. It is not practical to conduct all the tests manually and record the data. Therefore, an automated test system is developed which would subject shafts to different load conditions for which they are designed and record the data continuously for later analysis.

The test systems published such as multi axis motion control system [1] and landing gear testing system [2] used the LabVIEW programming as it provides advantage in programming and debugging [3], [4] and it can simplify and speed up the development of test systems. It also has powerful interface support[5]. A Peripheral Component Interface (PCI), Field Programmable Logic Array (FPGA) is used for the real time (RT) part which can be programmed easily using LabVIEW software.

In this paper, software implementation of the test system is explained in detail. The host part and the real time part are explained separately. An industrial PC is used as host and is used to operate all the non-real time data (example: change of recording parameter, selection of test, setting speed of drive, setting torque of load etc) using LabVIEW software. National Instrument’s PCI FPGA [6] card NI-7833R is used as real time part which is used for applying command to the drive, load and for data acquisition.

Rest of the paper contains the results obtained when drive and load motors are coupled with a test shaft. The trend of the automatic testing system development turns to make use of smart instruments technology and complete a series of testing tasks by replacing all kinds of instruments and measuring devices with a single multifunctional testing system at the core of the processor [7].

2. OVERVIEW OF THE SYSTEM
The test system consists of two motors, one motor is acting as a driver and the other one acting as load and they are placed on a test bed facing each other. These two motors are driven by servo drives. Servo drives are used to control the speed, torque and direction of motors.

The servo drives can be configured in analogue mode, meaning the servo drive is able to recognize 0 V to 10 V on one of its port and accordingly run the motor in speed loop/torque loop as configured. The 0 V corresponds to the zero speed/torque and 10 V corresponds to the maximum speed/torque (The maximum speed/torque should be configured first in the servo drive. For example, if it is configured to max. 2000 rpm then if 5V is applied to servo drive then it drives the motor at 1000 rpm).

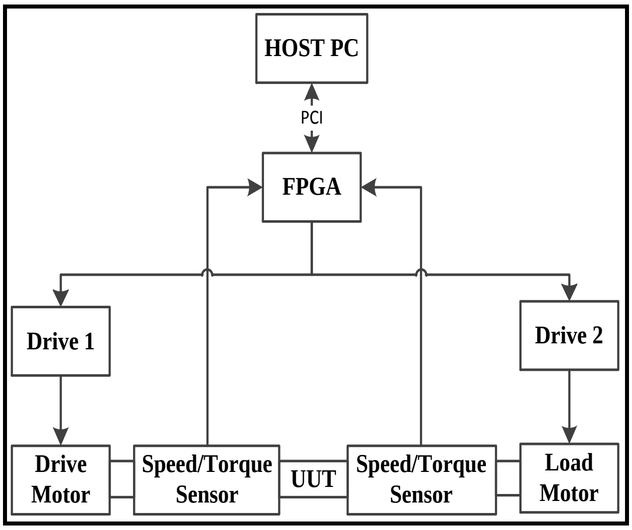

Figure 1 : Setup Block diagram

Test shaft is coupled between the two motors with torque-meter (which measures both speed and torque) sensors on each side of the shaft, which correspondingly give the drive speed/torque and load speed/torque. The real time data like speed/torque of drive and load, status of emergency stops etc, are recorded using FPGA. Figure 1 shows the block diagram of system and it depicts an in-expensive PC based Automated Test System (ATS) [8] design.

3. IMPLEMENTATION

Front panel Graphical User Interface (GUI) of the software is made 'technician centred' [4] so that user can understand the on-going process in the test system and be able to identify the errors in the system. Implementation steps of the host part and FPGA part are explained in this section.

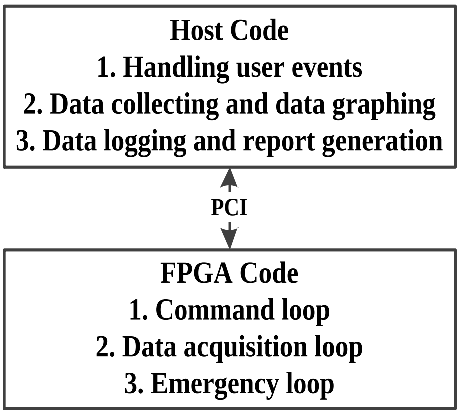

Figure 2 : Software Component Execution

Different software components which are executing on Host part and FPGA part are shown in Figure 2.

3.1 Host Part Implementation

The host PC is polling for the user events. As user events are generated they are sent to the appropriate loop.

3.1.1 Handling user events

The design of handling user events is explained with help of Figure 3. This includes setting of safeguarding parameters, saving the shaft details, creating a sequence such as, run N number of times to check the endurance of the shaft, changing recording parameters – this will give convenience to user to record only interested parameters.

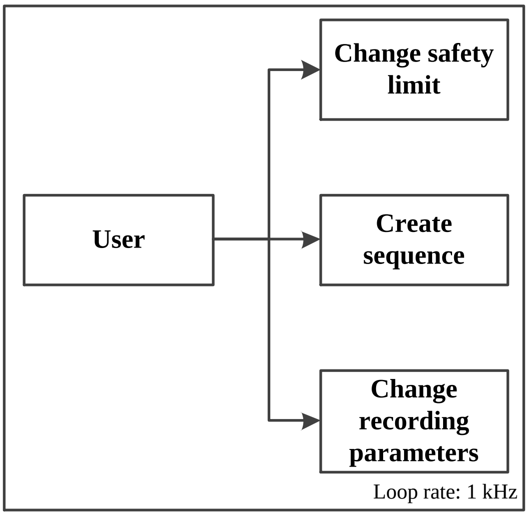

Figure 3 : Handling User Events

3.1.2 Data collecting data graphing

The real time data is being collected by FPGA (First In First Out – Buffer) FIFO. The FIFO is of finite length and hence there is a need to flush/over-write the FIFO as soon as host reads it. The rate of flush/over-writing depends on the FIFO length and the sampling rate of FPGA.

In the present case the FIFO length is 1k and FIFO length is also 1k, therefore rate of over-writing is 1 second (data acquisition rate of FPGA is 1 ms and hence 1k samples are collected in 1 second). Data collecting and graphing module from FPGA is shown in Figure 4.

The data graphing is for online analysis in which real time data is shown on data display and analysis module. The data collecting is for offline analysis in which real time data, collected from FPGA is stored into the local host computer[9].

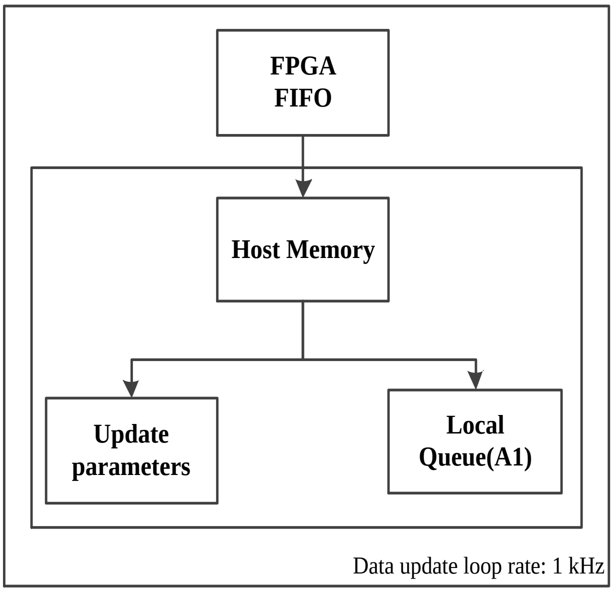

Figure 4 : Data collecting and data graphing

3.1.3 Data logging and report generation.

The design of data logging and report generation is explicated in Figure 5.

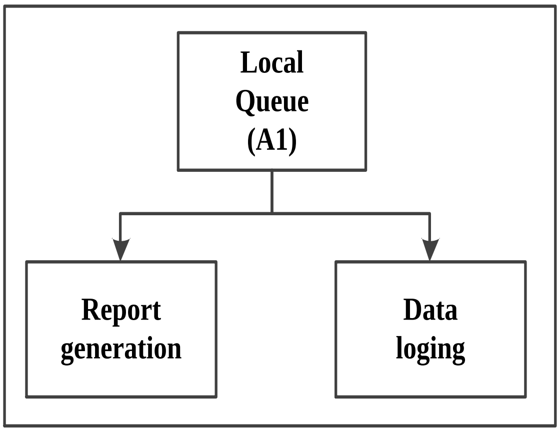

Figure 5 : Data logging and report generation

The local queue A1 referred in Figure 5 has been referred from Figure 4.

The data logging can be done in binary, ASCII or text format as required and report generation can be as per user's requirement of presenting the data.In this setup data logging is done in text format and it is logged in simple .txt file.

The test software can be used to write test data directly into the product database through local area network. At the same time the results can also be published in WEB server to make the test result for global enterprise view[7].

3.2 FPGA Part Implementation

All the three loops (as sown in Figure 2) are running independently.

In this setup, command loop is running at the rate of 10mS. Data acquisition and the Emergency loop are running at the rate of 1 mS.

3.2.1 Command loop

The logic to apply speed and torque command is explicated in Figure 6.

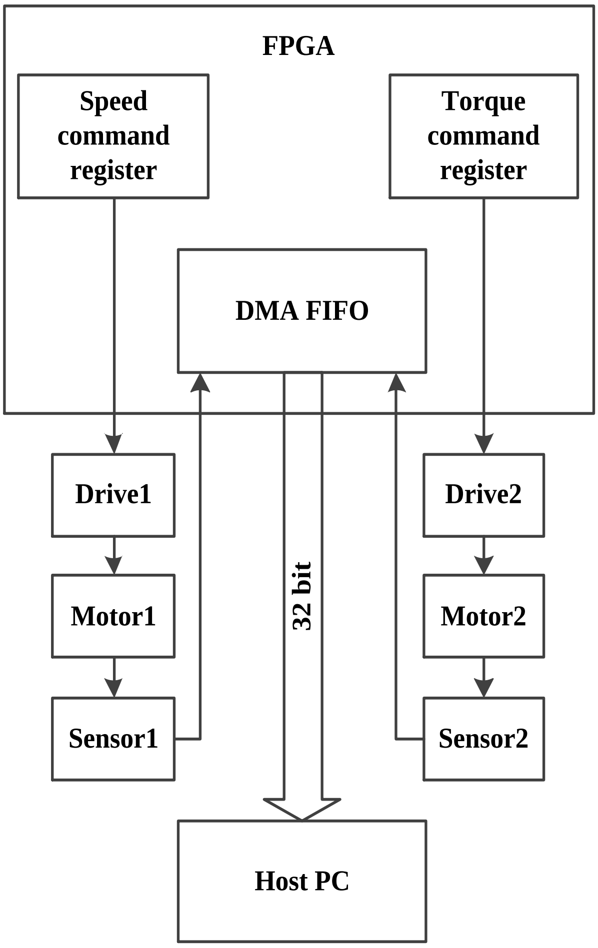

Figure 6 : Command loop

Both commands are applied with proper ramping (change of speed/torque to achieve set value). While using in manual mode speed and torque command are directly applied with given ramp rate whereas while giving a particular sequence – list of speed and torque commands are given using FIFO with calculated ramp rate from time parameter(change of speed or torque/ time difference = ramp rate).

3.2.2 Data acquisition loop

Data acquisition system as explained in by (Beitao Guo et al. 2009) [10] is adopted. The data from the torque/speed sensors, status signals are continuously acquired at the rate of 1 mS and sent to the FIFO. After every second the FIFO is overwritten. Synchronization is made between Host and FPGA so that any collected data should not be over-written without transferring to the host. Figure 7 explicates the design.

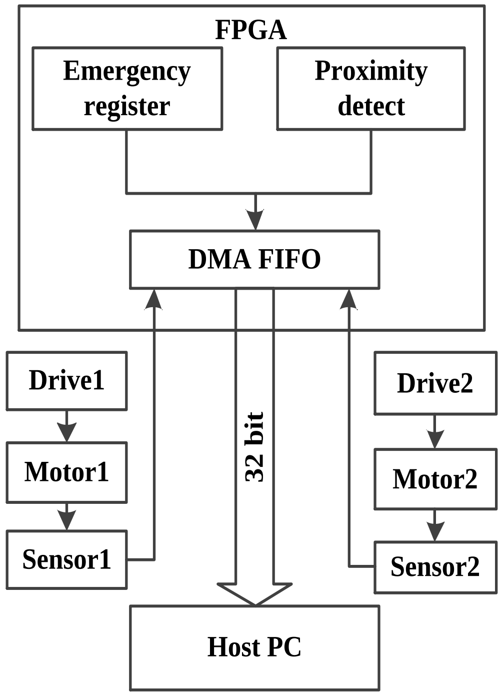

Figure 7 : Data acquisition loop

3.2.3 Emergency loop

Emergency loop is very important from safety point of view. FPGA is continuously monitoring for the emergency signals, for example – Emergency stops, Motor over-temperature indicator, higher speed/tolerance tolerance, three phase AC mains failure etc. All the emergency signals are being monitored and as soon as any one of the signal goes high, command loop is pushed into the safe speed/torque values. Figure 8 shows the block diagram of the emergency loop design.

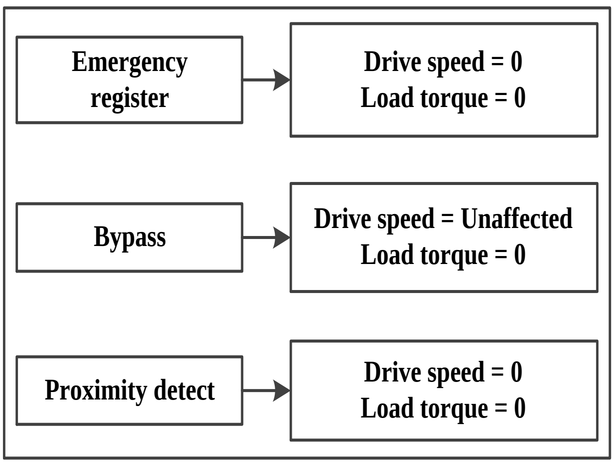

Figure 8 : Emergency loop

4. SIMULATION AND RESULTS

Results for acceptance test and endurance test are shown for a test shaft. Sample flight cycle is shown in Table 1.

Table 1 : Sample flight cycle

Sl. No	Time in second	Torque in Nm	Speed in rpm
1	0.00	0.00	0.00
2	0.450	51.561	1100
3	1.095	52.157	1100
4	1.130	51.199	1100
5	1.295	50.996	1100
6	1.700	5.099	110
7	2.275	5.099	110
8	3.395	0	0
9	12.950	42.753	1100
10	13.660	43.628	1100
11	13.840	42.176	1100
12	13.865	45.910	1100
13	14.840	46.017	1100
14	14.915	45.756	1100
15	15.275	4.150	110
16	15.875	0	0

The results of above test are shown in Figure 9 and Figure 10.

Optimized flight cycle shown in Table 1 indicates the range of torque and speed values that a shaft undergoes during a flight cycle. A flight cycle cannot be completed in 16 seconds. It is optimized for quick conduction of test.

4.1 Acceptance Test

The selection of acceptance test depends on the customer, who is accepting shafts from a manufacturer. It is a test to assure whether manufacturer has manufactured the shafts according to design.

This acceptance test can also be used as maintenance test for shafts.

An example of sample acceptance test is:

Drive speed of 500 rpm with the ramp rate of 300 rpm/s applied. 30 Nm of torque is applied at the load side at the ramp rate of 50 Nm/s. This is maintained for about 45 seconds with reversal of direction of drive during 30th second.

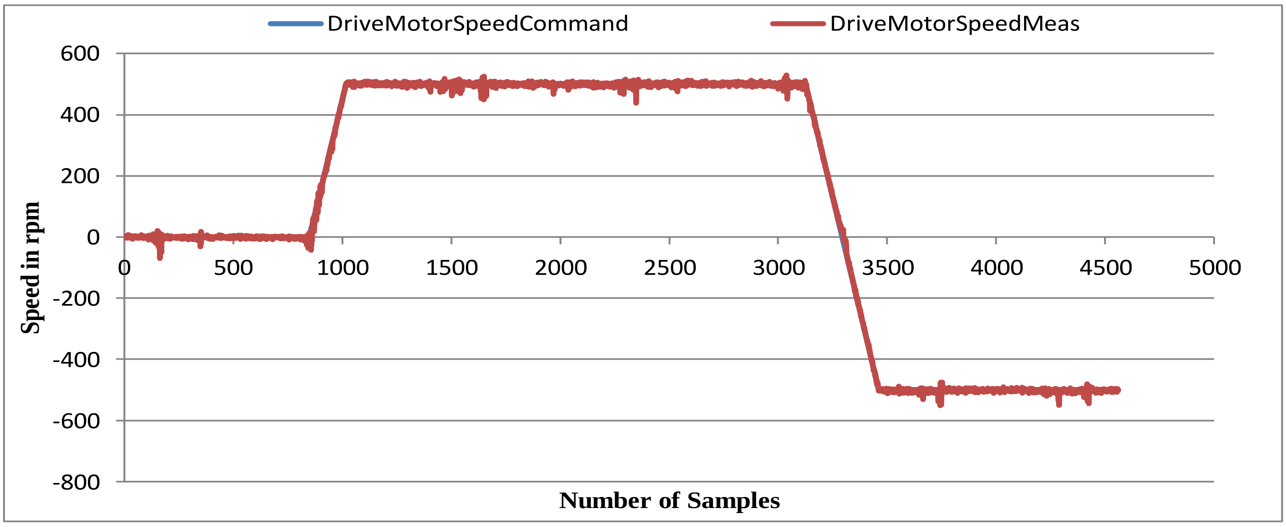

Figure 9 : Graph of drive motor speed v/s drive motor speed measurement

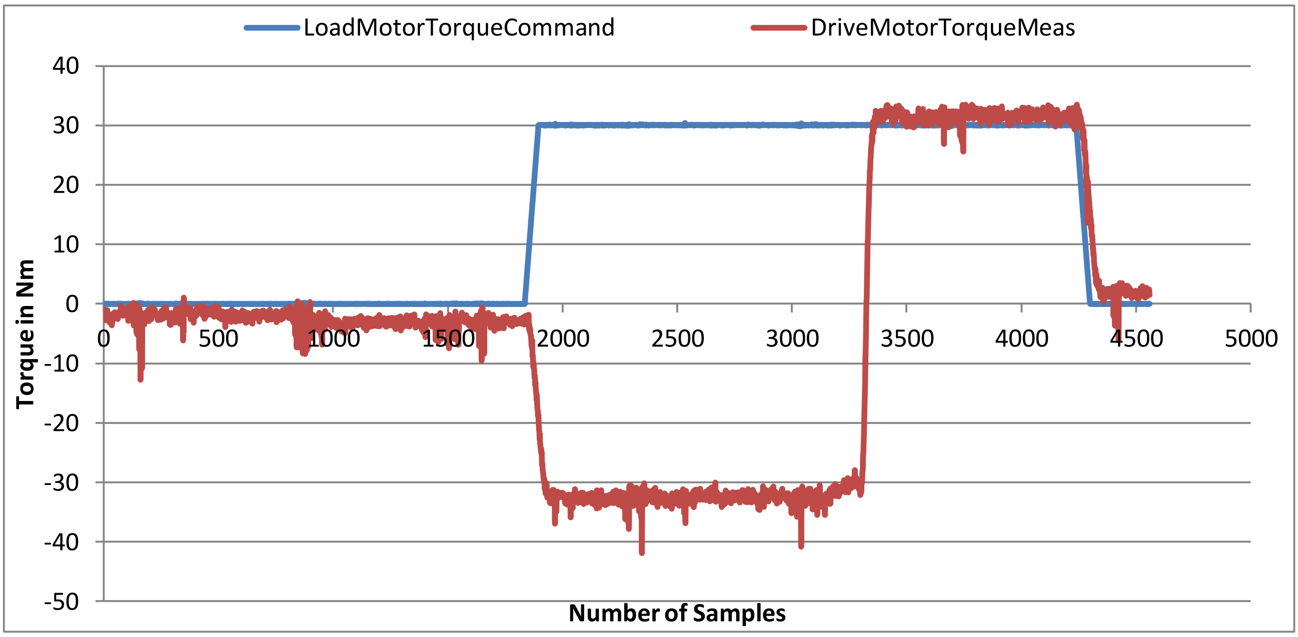

Figure 10 : Drive motor speed command v/s drive motor speed measurement

Figure 9 shows shift in the torque direction which is because of change of direction of rotation at drive motor as shown in Figure 10.

4.2 Endurance Test

An optimized profile of one flight cycle as shown in Table 1 is loaded in to the system. The result of flight cycle (Table 1) is as shown in Figure 11 and Figure 12.

Figure 11 indicates the measured speed with respect to commanded speed and Figure 12 indicates the measured torque with respect to applied torque at load motor.

If the predicted life of a shaft is one million cycles then this ATS can be used to conduct one million cycles and check whether designed shaft meets its criterion or fails before itself.

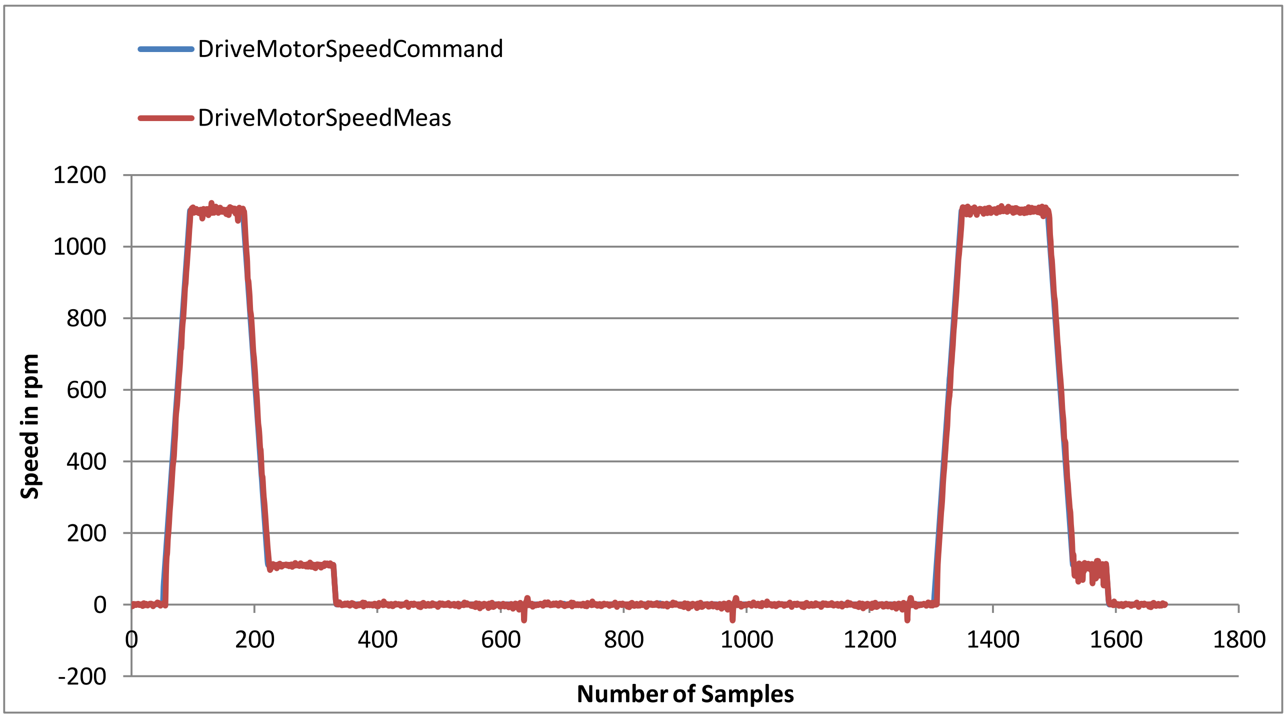

Figure 11 : Graph of drive motor speed command v/s drive motor speed measurement

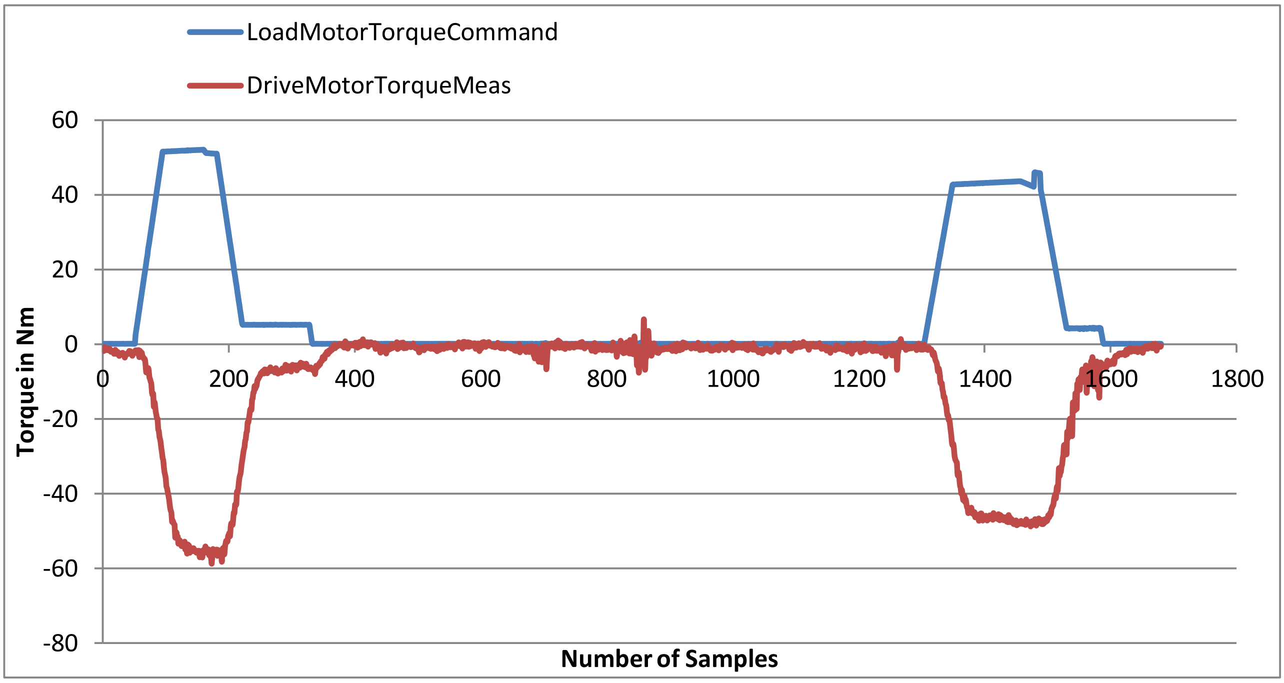

Figure 12 : Load torque command v/s Drive motor torque measurement

5. CONCLUSION

The process of testing of shafts is automated. The results of optimized flight cycle are shown in Figure 11 and Figure 12. This flight cycle for the test shaft can be run n-number (where n is predicted life cycle of shaft) of times to check its endurance.

The system can acquire, extract, process and automatically create the testing reports. Some of the software modules follow the architecture defined by R.McDonell [11].

The test software development using Labview provided mainly two advantages.

1. The enhanced GUI of the software makes readability of different test parameters very easy. It greatly reduced the system development time.
2. The data collected from FPGA is of very good real time and highly accurate. Similarly it made data handling at the host side very easy.

6. REFERENCES

[1] Chaturi Singh and K. Poddar, "Implementation of a VI-Based Multi-Axis Motion Control System for Automated Test and Measurement Applications" National Wind Tunnel Facility, IIT Kanpur.

[2] YangZhou, ChenYunxia and KangRui (2011), "A Study of Aircraft Landing Gear Testing System on PHM", IEEE Conference on – Prognostics & System Health Management Conference, MU3162.

[3] Russ Turley, Matthew Wright, "Developing Engine Test Software in LabVIEW" 1997 IEEE.

[4] Stanley P. Case, Kenneth C. Craig Jr. and Michael H. Nichols(1994), "A Path to Superior Testing in the 21st Century", IEEE, 219 -226.

[5] Chien-Lung Cheng, S.C.Chern, Yi-hung Lan "Automatic Testing System Based on LabVIEW for DC Motor of Portable Washing Machine", 2008 IEEE.

[6] Y. Han, R. Tzoneva and S.Behardien, "MATLAB, LabVIEW and FPGA linear control of an Inverted Pendulum", 2007 IEEE.

[7] LIANG Zhiyuan, Zhang Jianqing, "A Dynamic Balancing Testing Framework for Gyroscope Based on Embedded System", 2009 IEEE.

[8] Joseph Czapski(2000), "Useful Features of Automated Test Systems in the R&D Laboratory", IEEE, 601-613.

[9] Shen Wei-guo. Lian Yu, Zhen Hong-kang, "A Test System of Roving Flyer Based on LabVIEW", 2011 IEEE.

[10] Beitao Guo and Jianwei Zhang, Xin Niew, "Application of LabVIEW for Hydraulic Automatic Test System", 2009 IEEE.

[11] Richard McDonell and Robert bracket, "Designing an open test software architecture featuring Lockheed Martin LM-Star Case Study", 2004 IEEE.