ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 9, September 2014

Controlling of Multivariable Process Station Using – Cascade and a Selector in Labview

N. Nithya Rani[1], Jagatis Kumaar B[2],Vivek Vijayan C[3], Vinoth K [4]

Assistant Professor, Department of ICE, Saranathan College of Engineering, Trichy, Tamilnadu, India[1]

UG Student, Department of ICE, Saranathan College of Engineering, Trichy, Tamilnadu, India[2,3,4]

ABSTRACT: Conventionally, feedback controllers are used to control the multi variable process station. But this method is not efficient. So we propose the use of cascade controller along with a selector for a multi variable process station with air compressor. Since in this process, the pressure and flow is to be measured and controlled. We implemented the controller as cascade controller along with a selector in LabVIEW platform. Using this method, the air flow will be efficiently controlled with minimum error.

KEYWORDS: LabVIEW, cascade controller, selector, multi variable process station.

I.INTRODUCTION

The aim of this paper, is to presents robust control of air flow in a system. Control of flow is an important and common task in all process industries. Traditional PID controllers are easy to understand and implement. But by using conventional PID controller for controlling the flow, it cannot achieve ideal control. It suffers deficiencies in the face of uncertainly and plant non-linearities. This conventional techniques were unable to give satisfactory result. Cascade controller is a practical control design strategy that is employed in many advanced control design package. Although the cascade controller provides superior performance in comparison to simple traditional controllers such as PID. This paper generalized the results for the PID controller design. The proposed method incorporates this method in cascade controller and implementing the cascade controller along with a selector. Among the other method, cascade and a selector controller has gained a better performances, its easy and simple in understand, robustness and fast in practical applications. For the optimal operation and real time control an accurate description of the time domain behaviour of the system is required.

In industries lot of process analysis software available like a MATLAB, LabVIEW etc….. LabVIEW offers unrivaled integration with thousands of hardware devices and provides hundreds of built in libraries for advanced analysis and data visualization all for creating virtual instrumentation. In the process analysis methods only we can simulate the process then the exact parameters are feed in to the specific controller, by this method the process will get some interrupt during the parameter change. If we are using LabVIEW, directly we can analysis the process and change the accurate parameter to process.

II.MULTIVARIABLE PROCESS SETUP

Here the important components are compressor, motor, low value selector, flow controller, pressure controller, speed controller. The motor drives the compressor. Compressor produces the air flow with high pressure. This pressure is an important variable for flow process. The pressure and flow of the multi variable process are sensed separately and controlled separately by using different controllers. Then the output of these two controllers are given to a selector. This selector which select low value. This is a one loop of the cascade controller. The second loop is speed control of the motor. The speed control is given by separate controllers. The feed back value for the controller is output value of the selector. The reference value is taken from motor. Then that controller controls the speed of the motor. So the flow will get controlled.

10.15662/ijareeie.2014.0309067

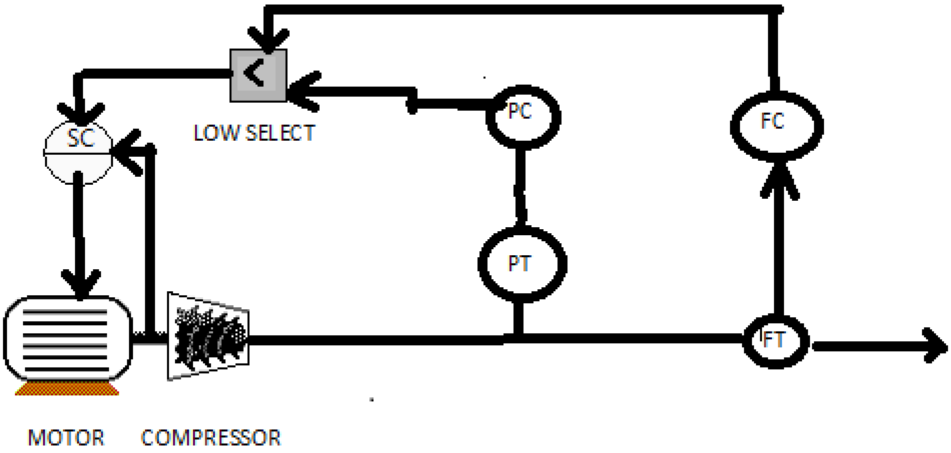


Fig. 1 Process Setup

III.CONTROLLER DESIGNING

ON-OFF CONTROLLER

An on-off controller is the simplest form of flow control device. The output from the device is either on or off, with no middle state. An on-off controller will switch the output only when the flow crosses the setpoint. For flow control, the output is on when the flow is below the setpoint, and off above setpoint. Since the flow crosses the setpoint to change the output state, the process flow will be cycling continually, going from below setpoint to above, and back below. In cases where this cycling occurs rapidly, and to prevent damage to contactors and valves, an on-off differential, or “hysteresis,” is added to the controller operations. This differential requires that the flow exceed setpoint by a certain amount before the output will turn off or on again. On-off differential prevents the output from “chattering” or making fast, continual switches if the cycling above and below the setpoint occurs very rapidly. On-off control is usually used where a precise control is not necessary, in systems which cannot handle having the energy turned on and off frequently, where the mass of the system is so great that flow change extremely slowly, or for a flow alarm. One special type of on-off control used for alarm is a limit controller. This controller uses a latching relay, which must be manually reset, and is used to shut down a process when a certain temperature is reached.

CASCADE CONTROLLER

Cascade control can improve control system performance over single-loop control whenever either: (1) Disturbances affect a measurable intermediate or secondary process output that directly affects the primary process output that we wish to control; or (2) the gain of the secondary process, including the actuator, is nonlinear. In the first case, a cascade control system can limit the effect of the disturbances entering the secondary variable on the primary output. In the second case, a cascade control system can limit the effect of actuator or secondary process gain variations on the control system performance. Such gain variations usually arise from changes in operating point due to setpoint changes or sustained disturbances.Cascade control has two objectives. The first is to suppress the effect ofdisturbances on the primary process output via the action of a secondary, or inner control loop around a secondary process measurement. The second is to reduce the sensitivity of theprimary process variable to gain variations of the part of the process in the inner control loop.As we shall demonstrate, cascade control can be usefully applied to any process where a measurable secondary variable directly influences the some dynamics. We will also demonstrate that despite frequent literature statements to the contrary, inner loop dynamics do not have to be faster than the outer loop dynamics. However, the traditional cascade structure and tuning methods must be modified in order for cascade control to achieve its objectives when the inner loop process has dynamics that are on the order of, or slower than, the primary process dynamics.

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 9, September 2014

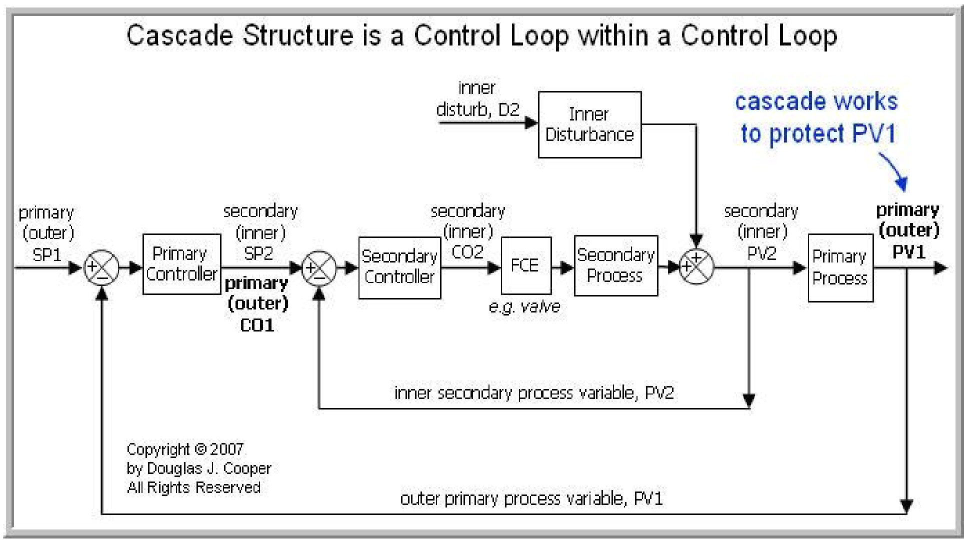


Fig. 2 General Block Diagram of Cascade Control Loop

IV. DESIGN OF SYSTEM SOFTWARE

LabVIEW

LabVIEW is a highly productive development environment for creating custom applications that interact with real world data or signals in fields such as science and engineering. LabVIEW is unique because it makes this wide variety of tools available in single environment. LabVIEW is a development environment for problem solving accelerated productivity and continual innovation. LabVIEW itself is a software develop the environment that contains numerous components.

G PROGRAMING LANGUAGE

G Programing language is a central tool LabVIEW using it you can quickly tight it together data acquisition, analysis, and logic operations and understand how data is being modified. It is a high level data flow graphical programming language designed to develop application that are

a. Interactive
b. Executing parallel
c. Multicore

LabVIEW contains powerful optimizing compiler that examines your block diagram and directly generates efficient machine codes.

1. HARDWARE SUPPORT

With LabVIEW you can use all of your hardware with single development environment connectivity is made possible with driver software which serves as the communication layer between LabVIEW and your hardware. LabVIEW supports thousands of hardware device including scientific instruments, data acquisition device, sensors, cameras, motors and actuators, machine vision and image processing hardware, wireless sensors and field-programmable gate arrays (FPGAs). In the rare event a LabVIEW driver doesn’t already exist you can also import driver from other programming languages.

2. UI COMPONENTS AND REPORTING TOOLS

UI tools are a set of reusable VIs that can help you create a better user interface experience. Interactive control such as graph, LED, and tables to view your acquire data. There are many tools to save a data to files are database, or automatically generate reports. Every LabVIEW block diagram also has an associated front panel which is user

10.15662/ijareeie.2014.0309067

interface of your application. On the front panel you can place generic control and indicator such as string, buttons, number and indicators such as thermometer, and dials and scales.

a. MODELS OF COMPUTATION: simulation syntax, state chart, component-level IP codes, interrupt driven programming, ODE modeling, C code. When LabVIEW was first released, G was the only way to define the functionality needed.

b. TECHNOLOGY ABSTRACTION: Harness emerging technologies such as FPGA's multicore CPUs and virtualization without painful relearning and additional development effort.

TOOLS USED IN LAB VIEW

.

WHILE LOOP:

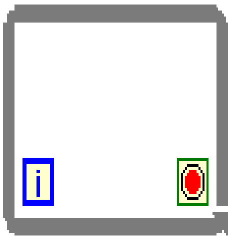


Fig. 3 While Loop in LabVIEW

It repeats the code within its subdiagram until a specific condition occurs. A while loop always executes at least one time.

Conditional Terminal—Evaluates a Boolean input value to determine whether to continue executing the While Loop. To specify whether the loop stops for a TRUE or FALSE Boolean value, configure the continuation behavior of the loop. You also can determine when the loop stops by wiring an error cluster to the conditional terminal.

Iteration Terminal (i)—Provides the current loop iteration count. The loop count always starts at zero for the first iteration. If the iteration count exceeds 2,147,483,647, or 2^{31}-1, the iteration terminal remains at 2,147,483,647 for all further iterations. If you need to keep count of more than 2,147,483,647 iterations, you can use shift registers with a greater integer range.

PID LEAD LAG:
Implements a PID controller with a lead/lag function, which is generally used as a dynamic compensator in feedforward control schemes. This VI uses a positional algorithm and is an approximation of a true exponential lead/lag. Use the DBL instance of this VI to implement a single control loop. Use the DBL Array instance to implement parallel multi-loop control.

PID:
Implements a PID controller using a PID algorithm for simple PID applications or high speed control applications that require an efficient algorithm. The PID algorithm features control output range limiting with integrator anti-windup and bumpless controller output for PID gain changes. Use the DBL instance of this VI to implement a single control loop. Use the DBL Array instance to implement parallel multi-loop control.

NOISE:
It generates random numbers with a little bit of smoothing and selectable amplitude.

10.15662/ijareeie.2014.0309067

WAVEFORM CHART:

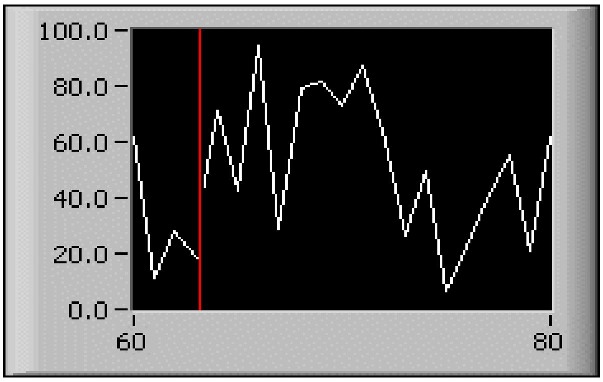


Fig. 4 Waveform Chart in LabVIEW

The waveform chart is a special type of numeric indicator that displays one or more plots of data typically acquired at a constant rate. The following front panel shows an example of a waveform chart.The waveform chart maintains a history of data, or buffer, from previous updates. Right-click the chart and select Chart History Length from the shortcut menu to configure the buffer. The default chart history length for a waveform chart is 1,024 data points. The frequency at which you send data to the chart determines how often the chart redraws.

SELECT FUNCTION:

Return the value wired to the T or F input depending on the value of Sif S is true, this function returns the value wired to T. If S is false this function returns the value wired to F. For building the select function select the select function under comparison palatte.

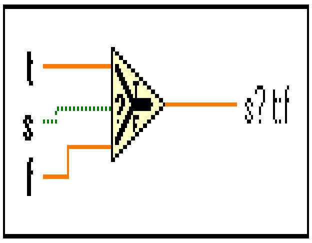


Fig. 5 Select Function in LabVIEW

BUNDLE:

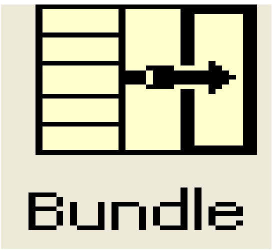
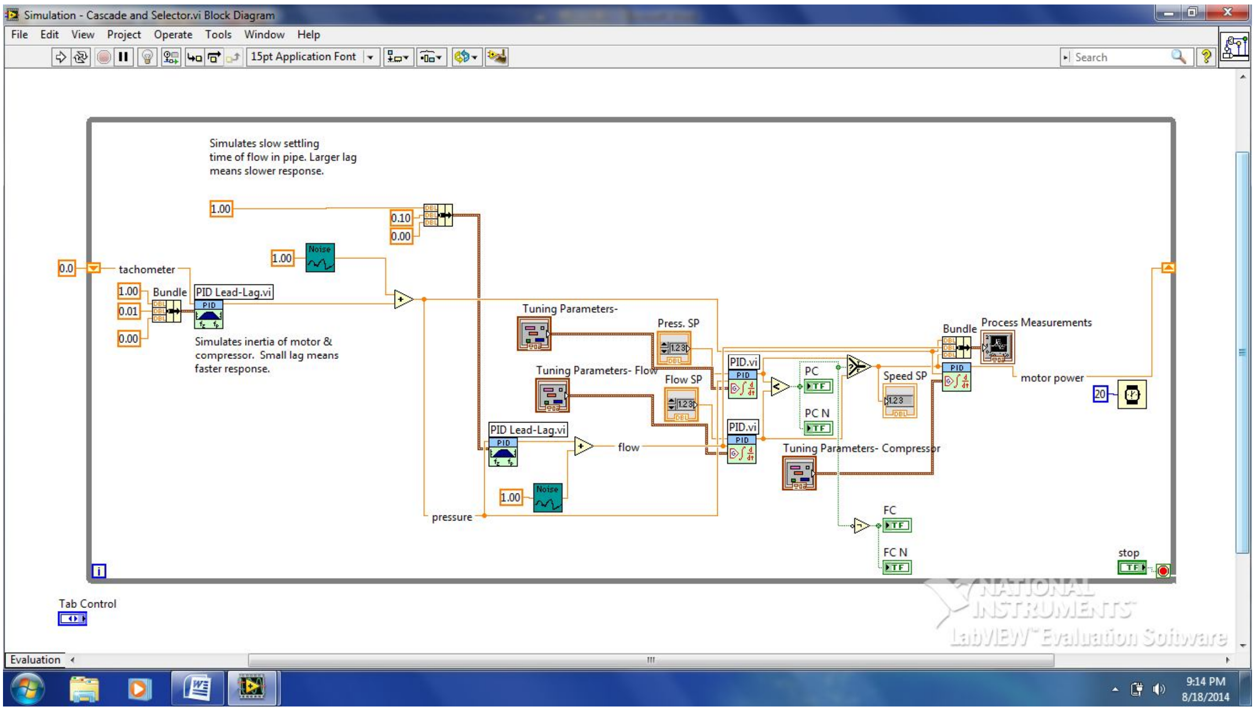


Fig. 6 Bundle Array Function

You also can use this function to change the values of individual elements in an existing cluster without having to specify new values for all elements. To do so, wire the cluster you want to change to the middle cluster terminal of this function. When you wire a cluster to this function, the function resizes automatically to display inputs for each element in the cluster. The connector pane displays the default data types for this polymorphic function.

10.15662/ijareeie.2014.0309067

V. CONTROLLING PROCESS



Simulates slow settling time of flow in pipe. Larger lag means slower response.

Simulates inertia of motor & compressor. Small lag means faster response.

Fig. 7 Block Diagram Window in LabVIEW for Multivariable Process

Overview: This VI demonstrates a cascade and selector control. A compressor, driven by a motor, has a tachometer on it which requires a PID loop to control the speed (the downstream loop). Pressure and flow out of the compressor are fed to individual PID controllers. We want to control the flow, but if the pressure exceeds a specified setpoint, it then becomes the controlled variable. This calls for a "low select" function to combine the two upstream controller outputs. The lower of the two outputs then becomes the setpoint for the compressor speed.

Run the VI and note that the PC (pressure controller) and FC (flow controller) turn black when they are selected. Normally, you would leave the pressure setpoint constant. Try making an upward step in flow and notice how the PC takes over control as pressure overshoots. In the real world, a plugged pipe would also cause the pressure loop to take over.For simplicity, all variables in this demo are in percent. In a real application, you could normalize all the input and setpoint values to percent before passing them to the PID controllers.Note that the downstream loop, also known as the inner loop, which is the compressor's speed control, must be faster than the outer loops. A factor of 10 is usually recommended. Otherwise, oscillation is assured. The response speed is simulated through the use of lags.

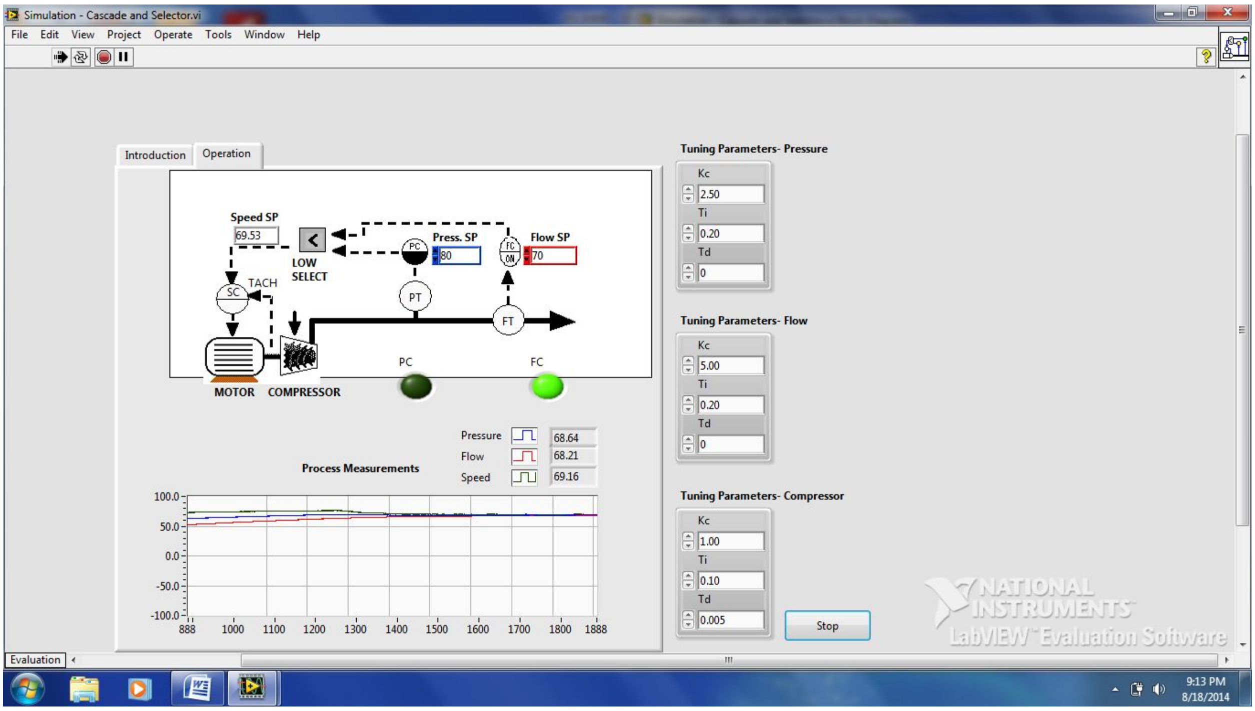
VI. RESULT AND DISCUSSION

Here the control of the multi variable process is done by cascade and a selector controller. Because it is better method than other controlling methods. Let us take on-off controller as an example. If we use on-off controller in this process, flow will go high level at controller at on state and then flow will go suddenly zero level at off state of controller. This is most important disadvantage of using other conventional controllers. So here the cascade and a selector controller is most suitable controlling method for this multi variable process. In this cascade controller one loop is low value selection loop. Then the another loop is speed controlling loop.

Here pressure is an important factor that affects the process. In this air flow control process pressure exceeds above to the air flow. Here we want to control the flow only. But pressure cause some effect at control process. So that low

10.15662/ijareeie.2014.0309067

value selector select low value and produce the control action at motor. So motor control the compressor and air flow is controlled very efficiently.



Speed SP 69.53

Press. SP 80

Flow SP 70

Pressure 68.64

Flow 68.21

Speed 69.16

Tuning Parameters- Pressure
Kc 2.50
Ti 0.20
Td 0

Tuning Parameters- Flow
Kc 5.00
Ti 0.20
Td 0

Tuning Parameters- Compressor
Kc 1.00
Ti 0.10
Td 0.005

Fig. 8 Front Panel Diagram in LabVIEW for Multivariable Process

VII.CONCLUSION

Here we concluded that the cascade controller along with a selector gives a improved performance for the above system. Disturbances in the flow was quickly corrected by the proposal controller (cascade controller along with a selector). It gave the faster response for the multivariable process station.

REFERENCES

1. N.Nithiya Rani, Jagatis Kumaar B, Vasantha P N, Vivek Vijayan C, A.ValithGas Leakage Monitoring and Control using LabVIEW in International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering (IJIREEICE)Volume 2, Issue 8, August 2014.
2. Modelling and simulation of complex control systems using LabVIEW, P.srinivas, P. Durga Prasada Rao, K .Vijaya Lakshmi
3. Hanwei Electronics Co.Ltd, MQ 2 sensor Datasheet.
4. Introduction to Real-time Control using LabVIEW, Ch.Salzmann, D. Gillert, P.Huguenin
5. Design and Implementation of Gas Leakage Security System, Ashish Shrivastava, Raghul veerma, Rajesh kumar, Ratnesh Prabhaker.
6. Current Teaching and Research Projects using Labview Control Design and Simulation, Kim Lankford, National Instruments, VIP 2005.
7. N.NithyaRani “Implementation of OPC-Based Communication Between Temperature Process and DCS on LabVIEW Platform” in the “BEST: International Journal of Management, Information Technology and Engineering (BEST: IJMITE)” Vol. 1, Issue 1, Oct 2013.
8. N.NithyaRani and S.Ranganathan “Advances in Control Techniques and Process Analysis with LabVIEW and DCS” in the “International Journal of Electronics, Communication & Instrumentation Engineering Research and Development (IJECIERD)” June 2013; Volume : 3; Issue : 2; June 2013.
9. N.NithyaRani “Advanced Process Analysis on LabVIEW” in the “International Journal of Advanced Research in Electrical and Electronics Engineering (IJAREEE)” Vol.1, No.1 November 2013.

10.15662/ijareeie.2014.0309067