Available online at http://www.journalijdr.com

IJDR

International Journal of
DEVELOPMENT RESEARCH

ISSN: 2230-9926

International Journal of Development Research
Vol. 07, Issue, 01, pp.10957-10961, January, 2017

Full Length Research Article

REMOTE TEST MONITORING AND CONTROLLING OF MULTIPLE SENSORS USING ANDROID APP FOR SMARTPHONES

***Pratiksha Milind Kulkarni and Mrunal Avinash Kulkarni**

Electronics and Telecommunication Department, M.K.S.S.S's Cummins College of Engineering for Women, Karvenagar, Pune-52

ARTICLE INFO

Article History:
Received 27th October, 2016
Received in revised form
14th November, 2016
Accepted 28th December, 2016
Published online 30th January, 2017

Key Words:

Android app, Smart phone, Sensors, Wi-Fi module.

ABSTRACT

This paper describes an advancement which could be implemented in the Automation Industry with the development of Android application for smart phones. The various sensors used in industries such as temperature sensors used in blast furnaces, fractional distilling columns for fuel extraction, food processing, pharmaceutical product manufacturing, water level sensors used in Dams, storage tanks for hydroelectric projects and societies, motion sensors used for security in industries and home automation systems can all be monitored and controlled remotely. The Android application receives all sensor values through an ARM LPC2138 based embedded system and Wi-Fi module. Android app is username and password protected and can be easily used by authorized personnel. We have developed a working prototype of the proposed system as our project.

INTRODUCTION

The Programmable Logic Controllers (PLCs) are used for performing various tasks in Mechanical, Automobile, Pharmaceuticals, Food, Fuel extraction Industries. Multiple Sensors such as temperature, water-level, Pressure and humidity are installed with these PLCs. These sensors help to perform the various production and packaging processes successfully. Temperature sensors used at the blast furnaces and various other industries form a vital part of the system design by keeping a track of temperature values and thereby avoiding any fire hazards. Similarly water-level and motion sensors aid the employees in different industrial environments. The communication between these sensors and the person in-charge is either through Bluetooth or the person has to visit the shop floor all the time for monitoring various processes. We have come up with advancement with respect to this problem by using Wi-Fi module and Android app for communication between sensors and the authorized personnel. All sensor values are communicated to the Wi-Fi module using serial communication cable. Wi-Fi modules increases remote range and these processes can be handled easily, errors corrected through an interactive Android app. A database is generated for maintaining all sensor values.

**Corresponding author: Pratiksha Milind Kulkarni,*
Electronics and Telecommunication Department, M.K.S.S.S's Cummins College of Engineering for Women, Karvenagar, Pune-52

The rest of the paper is structured as follows. Section II describes the literature survey and system architecture and Section III describes the implementation. Section IV presents the result.

Literature survey and system architecture literature survey

A similar kind of application is currently available in the market from some Controller manufacturing companies for the desktop PC installed in the Lab where various tests are being performed. The different parameter values coming from several controllers performing various tests are available through this application designed by the Controller's manufacturer. The communication between different controllers and the desktop PC is Ethernet based. This sets a limit on the distance from where all such tests can be controlled. They have to be monitored and controlled by a person who has to constantly remain present on the field. Following are some images of the application developed for desktop PC which is monitoring and controlling various tests.
In this paper we have proposed a method of overcoming such limits in the industries and companies. By using Android application we can monitor and control various sensor values effectively and immediately.

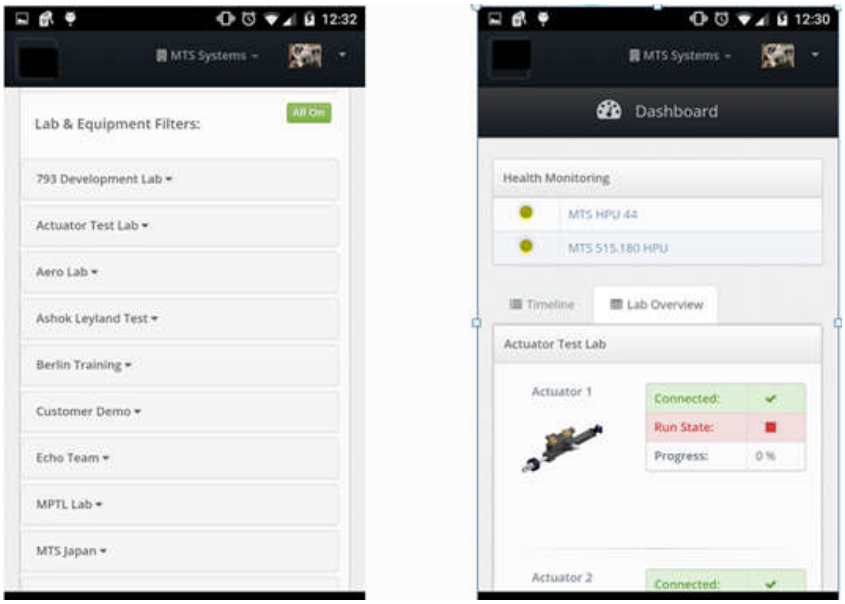


Fig. 1. Literature survey findings(1)

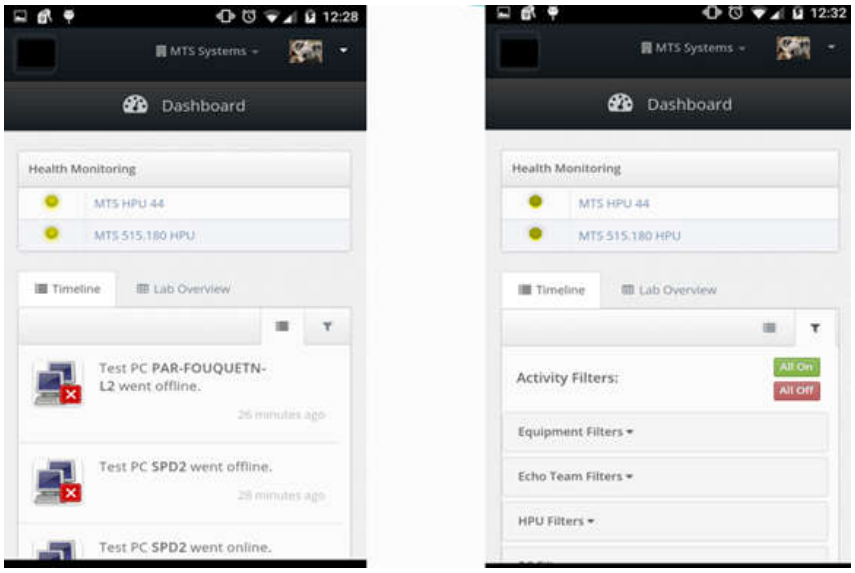


Fig. 2. Literature survey findings(2)

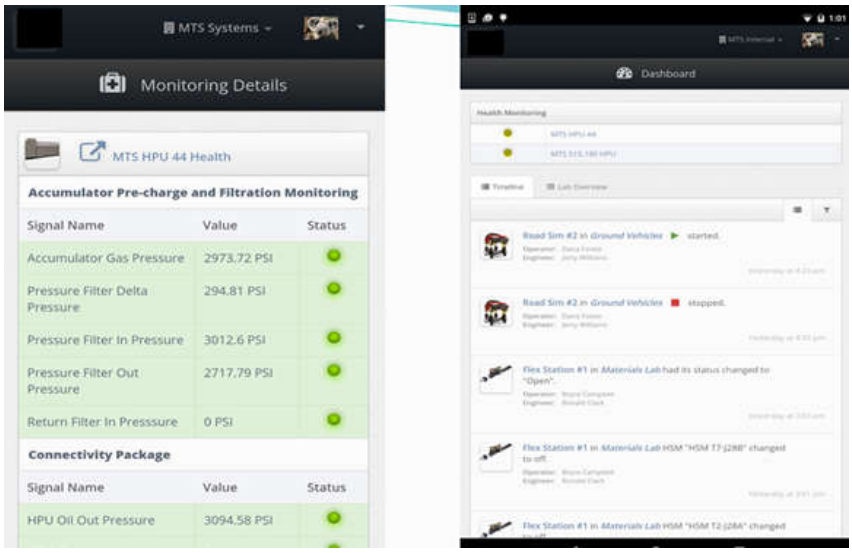


Fig. 3. Literature survey findings(3)

System architecture

The android application has been developed with Android SDK and Eclipse ide. The application consists of four different activities such as the Configure activity, History activity, Monitor and Control activity working together with the Microcontroller based hardware. The Microcontroller actually measures the value of each sensor parameter and the android app is used to have knowledge of it and control it. Following is the algorithm for developing Android application.

Algorithm for android app

1. Start
2. Declare a package for managing multiple activity windows of the app and also a separate package for each activity.
3. Import a bundle for managing various tasks.
4. Import a view object for rectangular screen and declare every activity with public class.
5. Use the onCreate() event handler to create the activity screen.
6. Write the XML file for every activity.
7. Generate a database to store record of every connection.

Configure activity

8. Import a widget-'button'. Use onClick () event handler to be able to click on configure button.
9. Use the text fields to accept ip-address, port number and mobile number for configuration. Create 'ok' button for completing configuration. Generate toasts to prompt the user to fill every text field.

Control activity

10. Import the java.io. DataInput Stream method to read the inputs and allocate them in the buffer. Import java.io. DataOutputStream to display outputs collected in the buffer.
11. Import the java.net package for socket programming. It is use to establish a connection between server and client and also for closing the connections.
12. Import view object for rectangular screen. Use the widgets like compound and toggle buttons.
13. Develop a pre-defined protocol for identification of communication between different sensor devices and android app.
14. Label the toggle buttons as device1, device2, device3. Use the on CheckedChanged() event handler to turn the devices 'on' and 'off'.

History activity

15. Import the java.util.Arraylist method for creating an empty list of 10 elements.
16. Import the java.util.log method to create logs of device connections.
17. Import the Arrayadapter method to display a layout file and the objects of this method are displayed in list view.
18. Use Log.d() method to display logs of device connections on the screen.

Login activity

19. The login screen uses text fields to accept username and password to keep a check on authorization breach.

Menu activity

20. Import view object for rectangular screen consisting of Configure, History, Monitor and Control buttons. Use the onCreate() event handler to create menu activity screen.

Monitor activity

21. Import the java.io.DataInputStream method to read the inputs and allocate them in the buffer. Import java.io.DataOutputStream to display outputs collected in the buffer.
22. Import the java.net package for socket programming. It is use to establish a connection between server and client and also for closing the connections.
23. Import the java.util.log method to create logs of device connections.
24. Import the SMSManager class to send a text message containing device's current values to the user.
25. Import JavaBeans class for using the various getter and setter properties.
26. Develop a pre-defined protocol for identification of communication between different sensor devices and android app.
27. Stop.

Implementation

The proposed design is implemented using an ARM series LPC2138 microcontroller, Wi-Fi module, 16x2 LCD, Android smart phone and temperature, water level, motion sensor.

Following is the prototype of the proposed design:

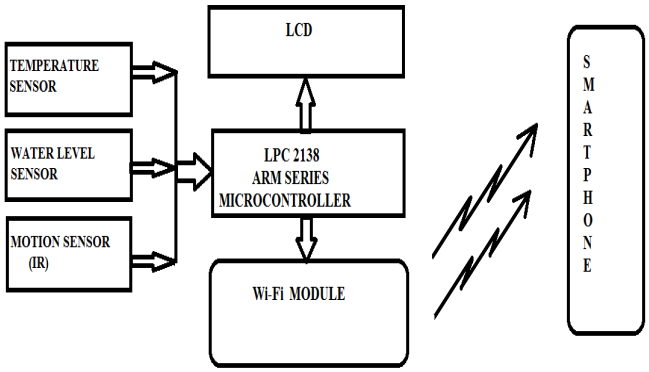


Fig. 4. Detail hardware design

We have implemented the proposed methodology on a small scale by developing hardware made-up of components namely temperature sensor (PT 100), water level sensor, motion sensor (IR sensor), 16x2 LCD, ARM series LPC2138 microcontroller, Wi-Fi module (HLK-RM04), android smart-phone (Samsung and Lenovo), RS 232 and MAX 232 for serial communication, step down transformer for power supply to microcontroller and Wi-Fi module, capacitors and resistors.

The system architecture can be explained as follows:

1. All the sensor values are communicated to the microcontroller LPC2138 via its various I/O (input-output) ports.
2. Each sensor uses a predefined protocol for communication with the microcontroller.
3. The microcontroller measures these values and uses standard values for comparison which are fed to the microcontroller by programming it in Keil software.
4. The 16x2 LCD is used to display sensor values at prime locations within the industry.
5. The sensor values are then communicated to the Wi-Fi module for transmission through RS 232 and Max 232.
6. Android SDK and Eclipse software has been used for developing the Android application.
7. By keeping the Wi-Fi mode on of the smart-phone and by establishing connection with the Wi-Fi module we can receive the transmitted values and take actions accordingly.

RESULTS

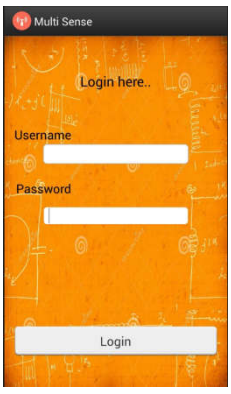
Results from the hardware part

The system developed and described in the paper is a working prototype of how it could be implemented on a larger scale in industries. The results of the monitoring and controlling processes are available at the prime locations within the industry in the form of LCD display. Following are the results obtained after testing the developed system:

Fig.5. Results obtained on hardware

Where,
Output of temperature sensor "T": temperature in degree Celsius.
Output of water level sensor "L": water level in liters.
Output of Motion sensor: Obstacle detected or no obstacle.

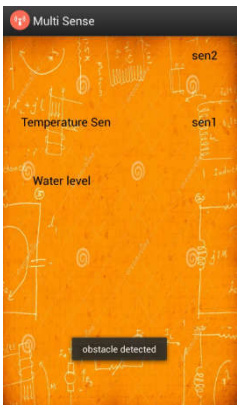
LOGIN WINDOW



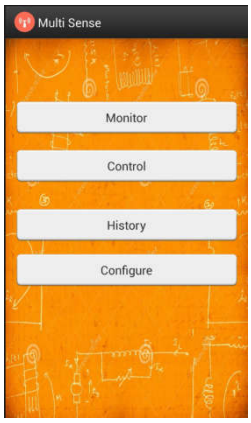
HISTORY WINDOW

CONFIGURE WINDOW

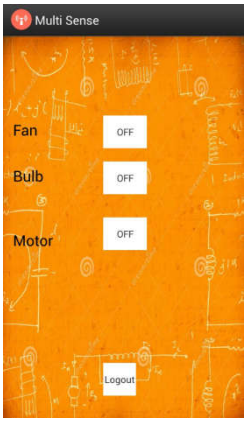
MONITOR WINDOW



MENU WINDOW



CONTROL WINDOW



Results from the smart-phone application implementation

Here are the results obtained after developing the android application and after using it to monitor and control the system designed. Following are the different screen's images representing the different activities performed in an android application for monitoring and controlling the sensors.

Conclusion

The proposed advancement in the form of implementation of a new system and the use of android application eliminates the need for a person to be constantly present on the field for monitoring and controlling various industrial processing. The android application is configured in such a way that only authorized personnel possessing the username and password can have access to the data. A build-up of all the sensor parameters on a single application instance facilitates the analytical study of these values. This analytical study can further be used to build a statistics that would help in learning the patterns in output of the system in which we will deploy these sensors and accordingly the corrective measures could be taken in case of errors.

Future scope

Web server can be developed so that the application can be implemented using any electronic device such as smart-phones, tab's, computers etc. The application can also be made compatible with iPhones, Windows phone and Blackberry phones. Further this application can be made to be platform

independent. The system can be automated even more by applying the control measures programmatically.

Acknowledgments

We would like to thank Prof. Mr. Sachin.A. Paranjape from the Department of Electronics and Telecommunication Engineering, M.K.S.S.S's Cummins College of Engineering for Women, karvenagar, Pune, India for his valuable guidance in developing the required hardware and android application for the proposed system.

REFERENCES

http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=4428690&newsearch=true&queryText=wireless%20sensor%20networks

http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=4625679&queryText=wireless%20sensor%20networks&pageNumber=2&newsearch=true

http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7217802&queryText=android%20%20based%20IR%20motion%20sensor&newsearchs=true

http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7238467&queryText=android%20based%20water%20level%20sensor&newsearch=true

http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7322863&queryText=android%20%20based%20IR%20motion%20sensor&newsearch=true

http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7410335&queryText=android%20app%20based%20motion%20sensor&newsearch=true

http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=4343996&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxpls%2Fabs_all.jsp%3Farnumber%3D4343996

http://www.ibm.com/developerworks/opensource/tutorials/os-eclipse-androidwidget/

http://www.tutorialspoint.com/android/index.htm

https://developer.android.com/training/index.html

https://www.thenewboston.com/videos.php?cat=6http://www.tutorialspoint.com/android/android_hello_world_example.htm

****