# An Itinerant GPRS-GPS and Sensors Integration for Atmospheric Effluence Screening

Rakesh Kumar Giri ,

*Department of Computer Science & Engineering,*
*Bharath University, Chennai,Tamilnadu,*
*rakesh274@gmail.com*

*Abstract*-**This paper presents an online mobile unit for air pollution monitoring. The proposed system consists of a transmitter and receiver part. The transmitter part is integrated single-chip microcontroller, air pollution sensors array, a General Packet Radio Service Modem (GPRS-Modem), and a Global Positioning System Module (GPS-Module) for transmitting the information. The receiver part is a pollution server with internet connectivity and a database unit and connected to various wired and wireless clients for further retrieval of information. The transmitter part consists of a Mobile Data-Acquisition Unit (Mobile-DAQ) and receiver a fixed Internet-Enabled Pollution Monitoring Server (Pollution-Server). The Mobile-DAQ unit gathers air pollutants levels (CO, NO2, and SO2), and packs them in a frame with the GPS physical location, time, and date. The frame is subsequently uploaded to the GPRS-Modem and transmitted to the Pollution-Server via the public mobile network. A database server is attached to the Pollution-Server for storing the pollutants level for further usage by various clients such as environment protection agencies, vehicles registration authorities, and tourist and insurance companies. The Pollution-Server is interfaced to Google Maps to display real-time pollutants levels and locations in large metropolitan areas. The system reports real-time pollutants level and their location on a 24-h/7-day basis.**

*Key words*- **Air pollution, General Packet Radio Service (GPRS), Global positioning systems (GPSs), microcontrollers, embedded systems, wireless mobile networks. Data acquisition, analog-to-digital conversion.**

## 1. INTRODUCTION

Air pollution monitoring is considered as a very complex task but nevertheless it is very important. The environmental concerns have been growing along with the development of human race and their population distribution. To an extent, the last years have experienced a rapid shift in the demographic distribution towards heavily populated urban cities than it was ever before. Environmental monitoring is required to protect the public and the environment from toxic contaminants and pathogens that can be released into a variety of media including air, soil, and water. Air pollutants include sulfur dioxide, carbon monoxide, nitrogen dioxide, and volatile organic compounds, which originate from sources such as vehicle emissions, power plants, refineries, and industrial and laboratory processes. Many air pollution systems in urban and rural areas that utilize smart sensor networks and wireless systems were reported in recent literature. An environmental air pollution monitoring system that measures CO, NO2, and SO2 was reported[1]. The system is based on a smart sensor micro converter equipped with a network capable application processor that downloads the pollutants level to a personal computer for further processing. The system provides alarm message depending on the detected pollution types in the field. A high-resolution surveillance Web-camera was used to monitor air quality via the Internet [4]. The Web-camera can be connected to network via a wired modem or wireless WiFi connection. The system monitors and transmit parameters atmospheric environment to a command center. Another wireless sensor network system was developed to monitor indoor air quality in[6] .

In this paper, I propose a system that integrates a single-chip microcontroller, several air pollution sensors (CO, NO2, SO2), GPRS-Modem, and a general positioning systems (GPSs) module. The integrated unit is a mobile and a wireless data acquisition unit that utilizes the wireless mobile public networks. The unit can be placed on the top of any moving device such as a public transportation vehicle. While the vehicle is on the move, the microcontroller generates a frame consisting of the acquired air pollutant level from the sensors array and the physical location that is reported from the attached GPS module. The pollutants frame is then uploaded to the General Packet Radio Service Modem (GPRS-Modem) and transmitted to the Pollution-Server via the public mobile network. A database server is attached to the Pollution-Server for storing the pollutants level for further usage by interested clients such as environment production agencies, vehicles regeneration authorities, tourist and insurance companies. The Pollution-Server is interfaced to Google maps to display real-time pollutants levels

and their locations in large metropolitan area. The rest of the paper is organized as follows . Section 2.0 specifies the system functional and nonfunctional requirements. Section 3.0 describes system hardware. The software architecture is described in Section 4.0. The results, implementation and testing are reported and discussed in Section 5.0. Finally, the conclusion is presented in Section 6.0.

## 2. SYSTEM REQUIREMENTS

A system can be characterized according to its functional and nonfunctional requirements. Functional requirements describe the primary functionality of a system while nonfunctional requirements describe attributes like reliability and security, etc. The system's functional requirements are as follows.

- System must support accurate and continuous real-time data collection.
- System needs to store the data and provide access to a location map interface.
- System needs to support mobility.
- System must use minimum power.
- System must be accessible from the Internet 24/7.
- System must be compact.
- System must mostly use off-the-shelf devices, components, and standards.
- System must support two-way communication between the client and the server.
- System must be field-configurable.
- System should be easy to deploy.

Nonfunctional requirements for the system dictate that the system is reliable, portable, accurate, maintainable, secure, accessible, and usable. In addition, the system must support performance standards for an adequate response time and storage space for data.

## 3. HARDWARE ARCHITECTURE

To satisfy the system's functional and nonfunctional requirements,two major building blocks are needed, namely: a Mobile Data-Acquisition Unit (Mobile-DAQ) and a fixed Internet-Enabled Pollution monitoring Server (Pollution-Server). The Mobile-DAQ unit is designed by integrating the following hardware modules shown in Fig. 1. As the figure shows, the Mobile-DAQ consists of a 16-bit single-chip microcontroller integrated with a sensor array using analog ports. The Mobile-DAQ is also connected to a GPS module and a GPRS-Modem using the RS-232 interface. Each of these components is described in the following.

### 3.1. 16-Bit Single-Chip Microcontroller

The microcontroller is a single-chip device that has rich built-in resources for digital input/output ports, 16 channels, 8/10 bits analog-to-digital converter, 8 input/output interrupt- driven timers, 12 Kbytes of RAM, 4 Kbytes of EEPROM, 256 Kbytes of FEEPROM, two RS-232 serial communication ports, 4 Control Area Networks ports, and SPI communication ports[9] . These resources are more than enough for the proposed application.

### 3.2. Sensors Array

The sensor array consists of three air pollutions sensors including Carbon Monoxide (CO), Nitrogen Dioxide (NO2), and Sulfur Dioxide (SO2)[10].As Table I shows, the resolution of these sensors is sufficient for pollution monitoring. Each of the above sensors has a linear current output in the range of 4 mA–20 mA. The 4 mA output corresponds to zero-level gas and the 20 mA corresponds to the maximum gas level. A simple signal conditioning circuit was designed to convert the 4 mA–20 mA range into 0–5 V to be compatible with the voltage range of the built-in analog-to-digital converter in the 16-bit single chip microcontroller described earlier .

### 3.3. GPS Module

The GPS module provides the physical coordinate location of the mobile-DAQ, time and date in National Marine Electronics Association (NMEA) format[11]. NEMA format includes the complete position, velocity, and time computed by a GPS receiver where the position is given in latitude and longitude[12]. The data packet from the GPS-Module includes an RMS Header followed by UTC time, data validity checksum, latitude, longitude, velocity, heading, date, magnetic variation and direction, mode, and checksum. The only information required for the proposed system is date, time, latitude and longitude. The GPS modem is interfaced with the microcontroller using the RS-232 communication standard.

### 3.4. GPRS-Modem

The general packet radio service (GPRS) is a packet-oriented mobile data service used in 2G and 3G cellular communication systems global system for mobile communications (GSM). The proposed system uses a GPRS-Modem as a communication device to transmit time, date, physical location and level of air pollutants. The modem used for the proposed system has an embedded communication protocol that supports Machine-to-Machine (M2M) intelligent wireless Transmission Control Protocol (TCP/IP) features such as Simple Mail Transfer (SMTP) E-mail, File Transfer Protocol (FTP), and Simple Messaging Service (SMS)

services Protocol. The modem supports an RS-232 interface that allows Serial TCP/IP socket tunneling.

### 3.5. Pollution-Server

The Pollution-Server is an off-the-shelf standard personal computer with accessibility to the Internet. As Fig. 1 shows, the Pollution-Server connects to the GPRS-Modem via TCP/IP through the Internet and the public mobile network. The server requires a private IP address for the GPRS-Modem and communicates over a pre-configured port. The Pollution-Server connects to a database management system (MySQL) through a local area network (LAN). The Pollution-Server runs a WampServer [13] stack that provides the Apache Web Server in addition to the PHP Server-side scripting language. Clients such as the municipality, environmental protection agencies, travel agencies, insurance companies and tourist companies can connect to the Pollution-Server through the Internet and check the real-time air pollutants level using a normal browser on a standard PC or a mobile device. The Pollution-Server can be physically located at the Environmental Protection Agency (EPA) or similar government agencies.
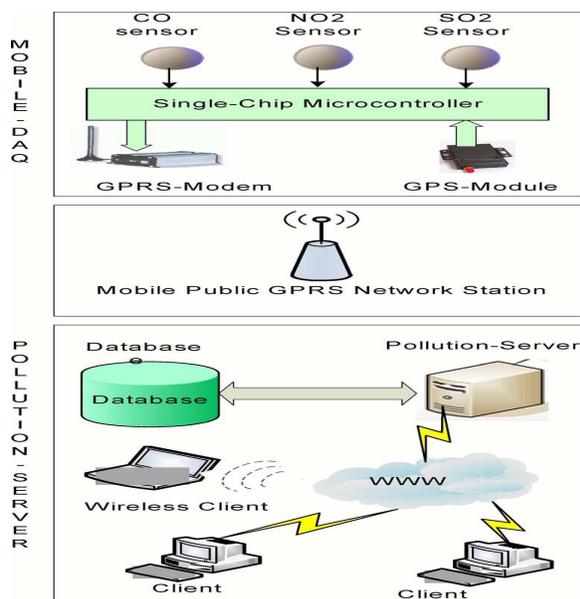


**Fig. 1. System hardware basic building blocks**



**Fig. 2. Photograph of the mobile DAQ unit mounted on the roof top of a car**
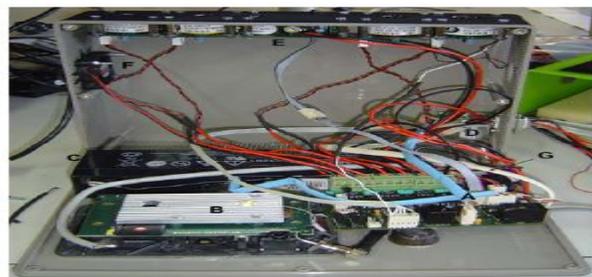


**Fig. 3. Interior of the DAQ. Main Board (A).GPRS modem, (B). Battery (C). GPS Module (D). Gas sensors (E). Cooling fan (F). Solar panel controller**

| Sensor | CO | NO2 | SO2 |
|---|---|---|---|
| Resolution (ppm) | <1.5 | <0.02 | <0.1 |
| Resp. time (t90) (s) | <25 | <60 | <25 |
| Op. range (ppm) | 0-1000 | 0-20 | 0-20 |
| Operating life (yrs) | > 2 | > 2 | >2 |
| Diameter (mm) | 20 | 20 | 20 |

**Table 2  Sensor Array Specification**

## 4. SOFTWARE ARCHITECTURE

The system software architecture is divided into two layers structure: physical layer and application layer.

### 4.1. Physical Layer

This layer is responsible for acquiring the real-time data from the sensors-array and the physical location, time and date of the sampled pollutants from the GPS module. This information is then encapsulated into a data frame by the microcontroller. The microcontroller then sends each frame to the GPRS-Modem through the RS-232 interface. The GPRS-Modem, in turn, sends each data frame to the Pollution-Server using the publicly available mobile network and the Internet.The physical layer is implemented using ANSI C language which is compiled to native micro controller code. The software implementing the physical layer is composed of five functions, namely:Ports-Read()function,Sensor Acquisition() function, GPS-Position() function, Data-Frame() function, And GPRS Transmit()function. Are called from a main program that is stored on and executed by the Mobile-DAQ microcontroller.

— Ports-Config() function: Developed to configure the digital inputs/outputs in addition to the resolution of the analog-to-digital converters that read the air pollutants level from sensor array outputs.

— Sensors-Acquisition() function: Reads each pollutant level as a voltage from the signal conditioning circuit output via the built-in analog-to-digital converter module of the microcontroller.

— GPRS-Read() function: Communicates with the GPS module through RS-232 and extracts latitude and longitude of the sampled air pollutant along with time and the date.

— Data-Frame() function: Encapsulates the IP address of the Pollution Server, a port number, the three pollutants levels, latitude and longitude of the sampled location, and time and date of the when the samples were taken. The data frame is shown in Table 4.

— GPRS-Transmit() function: Selectively sends the data frame to the GPRS-Modem using the RS-232 interface port. This frame is sent according to the algorithm shown in Fig. 4. As the figure shows, a data frame is only transmitted if the pollutant's level has changed since the last reading.

| Index | Air Quality Description | Band |
|-------|------------------------|------|
| 0-100 | Clean Air | 🟩 |
| 101-125 | Light Pollution | 🟨 |
| 126-150 | Significant Pollution | 🟧 |
| 150 above | Heavy Pollution | 🟪 |

**Table 3. Air Quality Description**

*4.2. Application Layer*

The application layer consists of three primary modules:Socket-Server, Air-Pollution-Index, and Google-Mapper.Socket-Server collects and stores pollutant data from all the Mobile-DAQs. Air Pollution-Index calculates pollution categories based on local pollution policies and regulations.Finally, Google-Mapper, makes this pollution information available over the Internet. Each module is described in the following.

— Socket-Server*:* Multithreaded Java program that uses Berkeley sockets to listen to a pre-configured port (e.g., 8080) for socket connections from the various remote Mobile-DAQs. Upon connecting with a Mobile-DAQ, the Socket-Server spawns a software thread that parses the data frame containing pollutant data along with the sampling time and location, stores the data frame in a database using the mySQL databasemanagement system and closes the connection.

— Air-Pollution-Index: Function to convert the raw pollutant level received from each Mobile-DAQ to pollution standards called air quality index (AQI) using the formula(1)

$$AQI = \left( \frac{\text{Pollution level}}{\text{Pollution Standard}} \right) * 100 \qquad (1)$$

The pollution standard is defined according the air quality standards of a particular region. For example, in the Tamil Nadu, the pollutant standard for CO, NO2, and SO2 are 20, 0.15, and 0.13 ppm, respectively.

| Unit-ID | Pollution-Server IP-Address | Pollution-Server Port # | Time | Date |
|---------|----------------------------|------------------------|------|------|
| *Latitude* | *Longitude* | CO-Level | NO$_2$-Level | SO$_2$-level |

**Table.4. Data-frame payload**
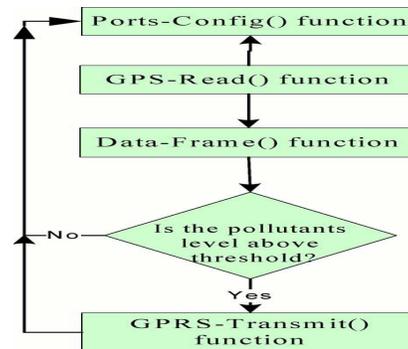


**Fig. 4. Mobile-DAQ software algorithm**

Following (1), the air quality is divided into four categories. An index value of 0–100 corresponds to clean air, 101–125 represent light pollution, 126–150 signify significant pollution, and above 150 means heavy pollution. In summary, the Air Pollution-Index function returns a pollution category from the raw pollutant data.

— Google-Mapper: PHP program running on the Apache web-server that reads the pollutant data from the mySQL database and plots it on a Google Map using the Google Maps API. In specific, an instance of a GMap object from is created using a JavaScript call.AGPolygon object based on latitude, longitude and the level of the pollutant is created for each region in the Map being shown. The color of the polygon follows the pollution category as calculated by the Air Pollution-Index function. For example, for the Chennai, a red color is used to denote Heavy Pollution indicating an AQI value of above 150, as shown in the Table 3. This program allows a user to click on a particular polygon representing an area on the map. Upon clicking, the program shows an information window showing the pollution levels of each of CO, NO2, and SO2 in parts per million (ppm), the time of last sampling and hot links to history of each of the gases. Upon clicking a hot link, a user can view the raw ppm values of a gas for that particular location. Fig. 5 shows these readings for the Bharath University of Chennai in the India. Fig.

7 shows sample CO, NO2, and SO2 readings from one of the locations at the university campus.
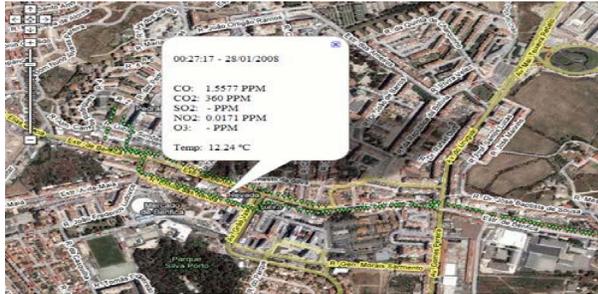


**Fig 5. Google Mapper data**

## 5. IMPLEMENTATION AND TESTING

The Environment Protection and Safety Section (EPSS) in Chennai  has monitored air quality since 1990 . Their current system is based on six static monitoring stations located around the Chennai metropolitan area.



**Fig. 6. Public interface showing actual pollutant values**

These stations send air pollutant data to a central server using fixed line modem connections. The pollution data is also available to the public through their Web site. This system has worked well. However, the data collected is limited to the vicinity of the six monitoring stations. Consequently, a mobile system based on the hardware and software architecture described earlier was built and tested in the India. The designed sensor array consisting of CO, NO2, and SO2 was interfaced through a signal conditioning circuit through analogue channels 5, 6, and 7 of the HCS12microcontroller, as shown in Fig. 1. The sensor output voltages representing the level of gas for each pollutant ( ) were converted to a ppm value for each gas. The GPS module was connected to COM0 and the GPRS-Modem was connected to COM1 of the microcontroller. Fig. 4 shows a typical data frame being transmitted from GPRS-Modem to the Pollution-Server. The Mobile-DAQ was mounted on a University bus that was driven around the campus of the BharathUniversity of Chennai to collect pollutant data. The Mobile-DAQ was mounted on top front of the bus

to avoid contamination from the bus exhaust. The pollutant data was collected for 12 h. Fig. 7 shows how a user can use the Internet to access pollutant levels in a location covered by the bus. As the figure shows, Google Maps is used as the primary interface. Pollutant data is shown using different colored polygons that are superimposed on the map. The color code used for these polygons was consistent with the AQI index of the Chennai Municipality. As the figure shows, different areas within the University campus have different levels of pollutants. The yellow polygon shows light pollution while the green polygons show clean air according to the AQI index. As Fig. 7 shows, a user can click any of the polygons to retrieve details of the various pollutant levels. A user can further drill down by clicking to view the past data for any of the gases for this location.
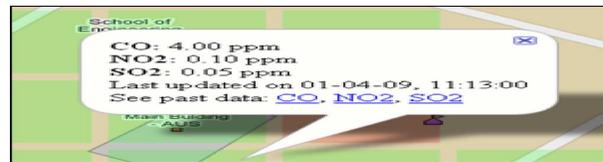


**Fig. 7. Details of pollutant data shown in Fig. 6**.

## 6. CONCLUSION

A wireless distributed mobile air pollution monitoring system was designed, implemented and tested using the GPRS public network along with GPS. The system utilizes city buses to collect pollutant gases such as CO, NO2, and SO2. The pollution data from various mobile sensor arrays is    transmitted to a central several that make this data available on the Internet through a Google Maps interface. The data shows the pollutant levels and their conformance to local air quality standards. It is worth mentioning that much more work is required to commercialize the system and to bring the system to be used for the general and specialized works.

## REFERENCES

[1] N. Kularatna and B. H. Sudantha, "An environmental air pollution monitoring system based on the IEEE 1451 standard for low cost requirements," IEEE Sensors *J.*, vol. 8, pp. 415– 422, Apr. 2008.

[2] F. Tsow, E Forzani, A. Rai, R. Wang, R. Tsui, S. Mastroianni, CKnobbe, A. J. Gandolfi, and N. J. Tao, "A wearable and Wireless sensor system for real-time monitoring of toxic environmental volatile organic compounds," IEEE Sensors *J.*, vol. 9, pp. 1734– 1740, Dec. 2009.

[3] Y. J. Jung, Y. K. Lee, D. G. Lee, K. H. Ryu, and S. Nittel, "Air pollution monitoring system based on geosensor network," in Proc. IEEE *Int* Geoscience Remote Sensing Symp.*, 2008, vol. 3, pp.1370–1373.

[4] C. J. Wong, M. Z. MatJafri, K. Abdullah, H. S. Lim, and K. L. Low,"Temporal air quality monitoring using surveillance camera," in *Proc. IEEE Int.* Geoscience and Remote Sensing Symp.*, 2007, pp.2864–2868.

[5] M. Gao, F. Zhang, and J. Tian, "Environmental monitoring System With wireless mesh network based on embedded system," in Proc. 5th IEEE Int. Symp. Embedded Computing, 2008, pp. 174–179.

[6] W. Chung and C. H. Yang, "Remote monitoring system with wireless sensors module for room environment," *Sens. Actuators B*, vol. 113, no. 1, pp. 35–42, 2009.

[7] J. W. Kwon, Y. M. Park, S. J. Koo, and H. Kim, "Design of air pollution monitoring system using ZigBee networks for ubiquitous-city," in Proc. Int. Conf. Convergence Information Technology,2007, pp. 1024–1031.

[8] M. AbuJayyab, S. Al Ahdab, M. Taji, Z. Al Hamdani, and F. Aloul, "Pollumap: A pollution mapper for cities," in *Proc. IEEE* Innovations in Information Technology Conf.*, Dubai, UAE, Nov.2006, pp. 1–5.

[9] H. W. Huang, The HCS12/9S12: An Introduction to Hardware And Software Interfacing, 1st ed. Florence, KY: Thomson Delmar Learning, 2006.

[10] Alpha Sense Gas Sensor Datasheets and Speculations. [Online]. Available: http://www.alphasense.com/alphasense_sensors/sulfur_dioxide_sensors.html

[11] National Marine Electronics Association Data. [Online]. Available: http://www.gpsinformation.org/dale/nmea.htm

[12] A. R. Al-Ali, Imran Zualkernan, and Fadi Aloul,"A Mobile GPRS-Sensors Array for Air Pollution Monitoring"