



# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: [www.ijirccce.com](http://www.ijirccce.com)

Vol. 4, Issue 12, December 2016

## Smart Building Prototype using PIC18F4550

MS. Ismael Sánchez Ríos<sup>1</sup>, Dr. Elio Lozano Inca<sup>2</sup>, Victor Santos Uceta<sup>3</sup>, Alexander Ortiz Lozada<sup>4</sup>

Professor, Dept. of Electronics, University of Puerto Rico, Bayamón, Puerto Rico<sup>1</sup>

Associate Professor, Dept. of Computer Science, University of Puerto Rico, Bayamón, Puerto Rico<sup>2</sup>

PhD. Student, Dept. of Computer Science, Purdue University, West Lafayette, Indiana, United States<sup>3</sup>

BS, Dept. of Computer Science, University of Puerto Rico, Bayamón, Puerto Rico<sup>4</sup>

**ABSTRACT:** Smart buildings are valuable for today organizations. They not only save energy, water and operational cost, but also improve security, event response time, and bring the benefit of decreasing pollution in the environment. This paper describes the implementation of a prototype for a smart building system to control switches and monitor different analogical and digital sensors widely used in similar systems. Each data acquisition module of this system is composed of a PIC18F4550 microcontroller with different sensors attached to a board and can manipulate electronic circuit peripherals for measuring physical variables in the environment. The PICKit2 application and the USB.c library were used to program the microcontroller for communication through the USB port. The PC interface program was implemented in Visual Basic with HID native API for communication through the USB port. SQL server was used to implement a database to collect data from data acquisition modules. A web server and a web application interface were implemented using ASP.NET technology to remotely control peripheral devices over TCP/IP protocol. The implemented prototype smart building system can be used in each room of the building to control and monitor temperature, relative humidity, atmospheric pressure sensors, motion detector, smoke detector and luminosity intensity detector and to detect the status of a switch to turn on a light bulb, radio, fan, or any electro domestic device. In conclusion, powerful and versatile information technology based on client/server and web technology was used to implement a prototype smart building providing an efficient and cost effective solution to the current market.

**KEYWORDS:** USB devices, intelligent buildings, human interface devices, HID, physical variables sensors, USB programming.

### I. INTRODUCTION

Smart buildings are an important outcome of information technology and are valuable for today organizations. They not only save energy, water and operational cost, but also improve security, event response time, and provide benefit to the environment performance through decreasing pollution. The PIC18F family [9] of microcontrollers offers cost-effective solutions for general purpose applications used in real time operating systems (RTOS) which require complex communication protocols such as TCP/IP, CAN, USB, ZigBee, LCD and control motors. The prototype system is composed of various data acquisition modules to collect sensor data. Each of these modules uses PIC 18F4550 to communicate with the PC through an USB port. Each client PC sends the collected data to SQL server at periodic intervals. On the server side we implemented a web server to provide a web interface to interact with the modules through the web (Figure4.bshows the developed prototype system).

The main purpose of this research was to describe the implementation of a prototype smart building system composed of various data acquisition modules. Each of these modules were based on PIC 18F4550. The software of this microcontroller was implemented in the C programming language using USB.c library provided by MicroC [10] for communication through an USB port. On the PC side the client data acquisition interface program was implemented in Visual Basic. This program uses HID native API for the communication with the USB port. On the server side, a SQL database server was created to store collected data from PC client modules. In addition, a web server with a web application interface based on ASP.NET technology was implemented to remotely control peripheral devices over TCP/IP protocol.

# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: [www.ijirccce.com](http://www.ijirccce.com)

Vol. 4, Issue 12, December 2016

This microcontroller is used to convert analogue signals from the temperature and humidity sensors to digital signals. In addition it has input and output digital ports capable of receive discrete data and produce discrete output and it is used for the smoke and movement sensors. Temperature, relative humidity, atmospheric pressure sensors, and other sensors such as a motion detector, smoke detector and luminosity intensity detector were used to implement each data acquisition module for the smart building system. This module can also detect the status of a switch by means of digital inputs used in the microcontroller activating events such as turning on light bulbs and turning on radio or any electro-domestic device. The USBTrace program was used to monitor the signals traveling from the computer to the sensors to ensure the correct communication between the software and the circuit. The end product is a combination of the power and versatility of two programming languages extensively used in the industry and the new advances in the web and client /server technologies.

The content of this paper is organized as follows. In section I a literature review regarding smart building systems is presented. Identification and selection of the hardware and software required to develop the proposed system is made in section II. Also in this section preliminary experiments, implementation of the proposed system, analysis and discussion of the results of the experiments are presented. Finally, in section III the conclusions regarding the developed system and future work to be performed are performed.

## II. RELATED WORK

Gutpa et al. [12] used a microcontroller and a wireless module to implement a smart city with a detection system for garbage, water leakage, earthing fault, and text messaging, fire detection, and wireless auto-switching system. Similar systems has been developed by Zhenghua [1], Shawki [2], Edward [3], Abdulzahra [5], Kaur [6], Bhaskar [7] to implement a microcontroller-based smart home. Panna et al. [8] used a PIC 18F4550 microcontroller to develop a smart home for energy saving. Their prototype has various sensors and actuators to control light, fans, and air condition. Schor et al. [13] used a web service-based approach to implement a smart building system. This system integrates sensors and actuator nodes into IP-based networks. Similar work was developed by Levente [14] to implement a web enabled smart building. For our system a basic circuit connected with different sensors and actuators was developed. The initial program for the microcontroller was implemented in C programming language and the PC user interface was implemented in Visual Basic using the program examples provided by Dogan [4].

## III. EXPERIMENTAL SETUP

### A. DESCRIPTION AND CONNECTIONS OF DATA ACQUISITION MODULE

The following components were used: USB connectors, 1k $\Omega$ , 4.7k $\Omega$  resistors, 100nF, 200nF, (2) 33pF capacitors, 8MHz Crystal Quartz Oscillator, PIC18F4550 microcontroller, sensors, detectors; PGMPIC USB Development Board and T-TECH PCB Prototyping Machine. The description of the sensors and switches are as follows.

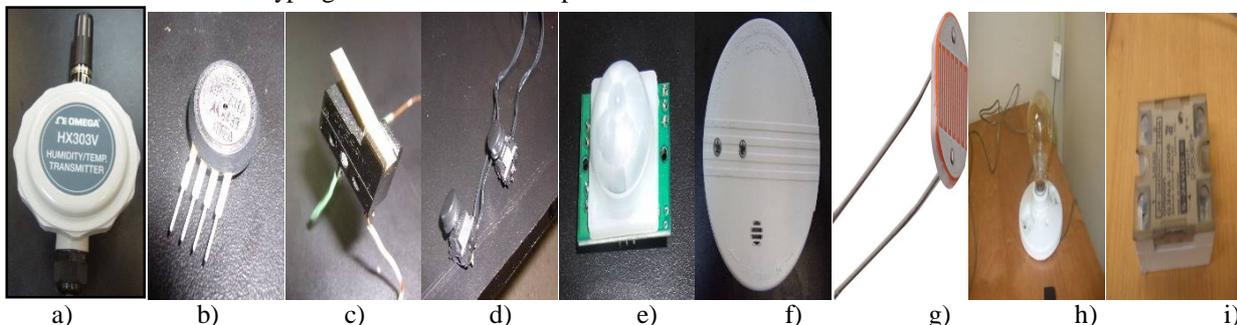


Figure 1. Sensors and switches used in this implementation. From left to right: a) Omega's temperature and humidity transmitter HX302/303; b) Motorola's atmospheric pressure sensor MPX200/201; c) Limit switch; d) Push button; e) Parallax's passive infrared sensor; f) Kidde's ionization smoke alarm; g) Light resistor (LDR); h) Light bulb; i) Solid state relay.

**1. Temperature and Humidity Sensor HX302/303.** This sensor (Figure 1.a) is calibrated to measure the temperature in Celsius degrees in the range 0°C to 100°C and the relative humidity in the range 0% to 100%. An external power

# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: [www.ijirccce.com](http://www.ijirccce.com)

Vol. 4, Issue 12, December 2016

supply of 12v to 24v DC connected to two Vin is required to operate this sensor. The temperature sensor must be connected to an RA0 analogue port and the negative to ground. The humidity sensor must be connected to an RA1 analogue port and the negative to ground. Figure 2.a shows the connections of this sensor.

**2. Atmospheric pressure Sensor MPX200/201.** This sensor (Figure 1. b) is calibrated to measure the pressure in kilo-Pascals (kPa). It needs an external power supply of 5v DC. This power can be obtained from the USB connector. The “output +” must be connected to analogue port RA2 and the “output –” must be connected to ground. Figure 2.b shows the connections of this sensor.

**3. Actuators (Switches).** Switches and push buttons (Figure 1.c and 1.d) are used to indicate the state of a door (closed or open). This switch must be connected to power supply of 5v DC. This power can be obtained from the USB connector. For security reasons the power must have a resistor in serial of 1 kΩ to decrease the current that pass through switches. The node between the switch and the resistor must be connected to digital port RB0. Figure 2.c shows the diagram of how to do the connections for the switch of the door.

**4. Light detector (Photo resistor).** A photo-resistor (Figure 1.g) is used to detect the presence or absence of light in a room. A power supply of 5v DC obtained from USB connector is provided for this detector. To create the light detector, we use a photo resistor (LDR) and a potentiometer of 100kΩ. The potentiometer is connected in serial using two Pot terminals. The Pot will be used to control the light sensibility. Node between Pot and LDR must be connected to digital port RB1. Figure 2.d shows the diagram of how to do the connection.

**5. Movement detector.** This detector (Figure 1. e) works with an infrared emisor. It can detect movement around 20 feet. An external power supply to operate of 3.3v to 5v DC (V+) is required for this detector. The power can be obtained from USB connector. The output signal of this detector is connected to a digital port RB2. Figure 2.e shows the connection of this detector and Figure 2.f shows the connection of an application of this detector.

**6. Smoke detector.** This detector (Figure 1. f) Works along an alarm which is activated when it detect smoke. It requires an external power supply connected to 120v AC. This detector has an internal alarm which is activated when it detects smoke. A relay and a transformer to convert 120v AC voltage to 5v DC is needed to connect the output of this detector as the input to the digital port RB3. Figure 2.g shows the connection of the smoke detector.

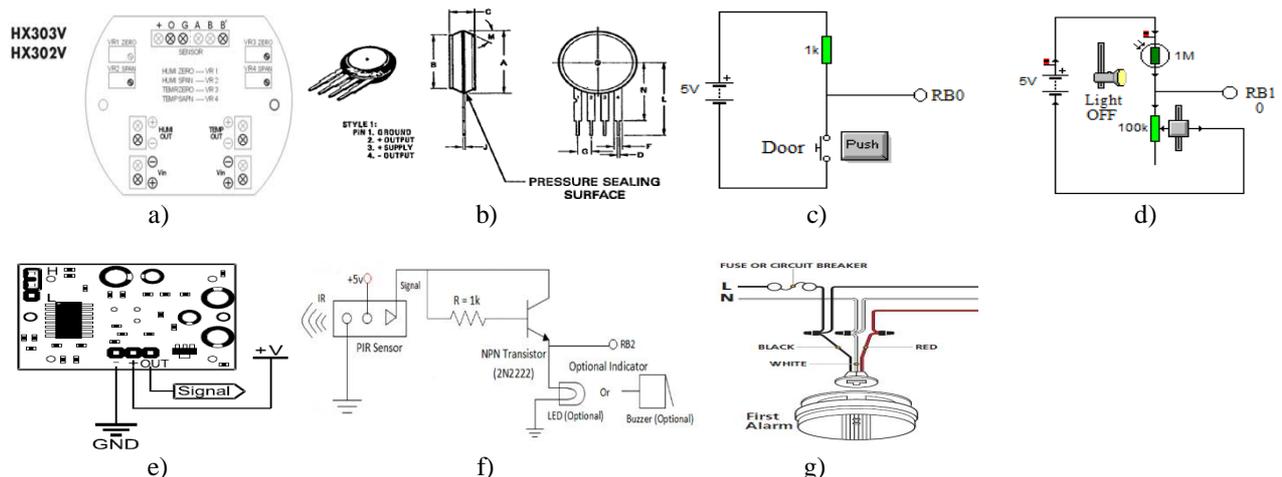


Figure 2. a) Internal diagram of the temperature and humidity sensor. b) Diagram of the atmospheric pressure sensor. c) Diagram of the door’s switch. d) Diagram of the light detector with a photo resistor (LDR). e) Diagram of the movement detector. f) Application of the movement sensor with indicator. g) Diagram of the smoke detector.

## IV. SOFTWARE IMPLEMENTATION

The following software were used to implement the programs for the microcontroller, the PC client interface and the web server: MikroC, mikroElektronika C compiler, PICkit 2 development programmer/debugger, EasyHID USB Wizard, USBTrace (USB Protocol Analyzer), Microsoft Visual Studio 6.0, Microsoft Visual Studio 2008, SQL Server



# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: [www.ijircce.com](http://www.ijircce.com)

Vol. 4, Issue 12, December 2016

Management Studio Express 2005, CadSoft Computer's EAGLE Program (PBC layout and fabrication tool), ASP.NET and LogMeIn.

## A. PROGRAMMING THE MICROCONTROLLER

**1. Initial configuration.** mikroC compiler was used to implement the program for the PIC microcontroller. Device PIC18F5450 must be selected to create a new project. CPU speed of 48MHz is chosen for USB 2.0 interface. The following options of the wizard were used for the implementation: PLLDIV\_2\_1L, CPUDIV\_OSC1\_PLL2\_1L, USBDIV\_2\_1L, FOSC\_HSPLL\_HS\_1H, FCMEM\_OFF\_1H, IESO\_OFF\_1H, PWRT\_ON\_2L, BOR\_ON\_2L, BORV\_43\_2L, VREGEN\_ON\_2L, WDT\_OFF\_2H, WDTPS\_256\_2H, MCLRE\_ON\_3H, LPT1OSC\_OFF\_3H, PBDEN\_OFF\_3H, CCP2MX\_ON\_3H, STVREN\_ON\_4L, LVP\_OFF\_4L, ICPRT\_OFF\_4L, XINST\_OFF\_4L, DEBUG\_OFF\_4L.

The configuration PLLDIV\_ is a pre-scaler function to divide the clock oscillator of the microcontroller by any assigned quantity. For this project a crystal quartz of 8MHz was used. The pre-scaler needs to be of 4MHz, because PLLDIV\_2\_1L was selected to divide the clock speed by 2 producing 4MHz. 96MHz is obtained because the multiplexer (MUX) creates 4:96MHz PLL. This speed is divided by 2 producing 48MHz which is the required frequency to work with the high speed USB 2.0. More details about the internal clock of the microcontroller can be found on the page 24 of PIC reference manual [PIC].

**2. Program description.** Basically the program is executing a set of instructions for the PIC 18F4550 inside an infinite loop. The program must include the library USBdsc.c provided by [10]. This library needs to be assigned input and output buffers. For this program, 32 bytes were used because more than 16 characters were required. This option can be obtained from the menu *Tools* and selecting *HID terminal*. In the Descriptor window, the output and input of the report length must be 32 bytes. Sometimes this configuration is not saved automatically, so the library USBdsc.c needs to be modified manually to enter the buffer values directly.

The PORT A was programmed for analog inputs. The instruction *TRISA = 0xFF* starts *PORT A* as inputs. Channels *AN0*, *AN1*, *AN2*, ..., and *AN12* are assigned for analog inputs because they need to be converted internally in the microcontroller from analog to digital (A/D). Channel *AN0* was assigned to port *RA0*, channel *AN1* to port *RA1* and channel *AN2* to port *RA2*. *RA0* and *RA1* are connected to temperature and humidity sensors respectively. The port *RA2* is connected to atmospheric pressure sensor. The temperature sensor is calibrated to measure in degree Celsius (°C), the humidity sensor in percent (%), and the pressure sensor in kilo-Pascals (kPa). The A/D converter works with 10 bits ( $2^{10} = 1024$ ). The output voltage of the A/D is multiplied by the reference voltage of USB (+5v) and then is divided by 1024. All the voltages must in milli-volts (mV) units. The initial step to convert the voltage of a sensor is given by the formula:

$$\text{Conversion Volt. A/D} = (V_{\text{sensor}} * 5000 \text{ mV}) / 1024$$

Following the calibration specifications, the conversion of the temperature and humidity sensor is multiplied by 100 obtaining a temperature in °C and a humidity in %. The conversion of the pressure sensor is obtained in kPa. To convert kPa to milli-Bars (mB) we use the following formula:  $\text{mB} = (2.0 * V_{\text{conversion}} + 0.95) / 0.009$ . Details of the A/D converter and the localization of the channel and the name of the instructions appear in the PIC18F4550 datasheet, pages 259 – 265 [PIC].

The PORT B is programmed half to be digital inputs (*RB0*, *RB1*, *RB2*, *RB3*) and the other half digital outputs (*RB4*, *RB5*, *RB6*, *RB7*). The instruction *ADCON1 = 0xFF* initializes the *PORT B* to be used as input and/or output. The instruction *TRISB = 0x0F (0000 1111)* initializes half of the *PORT B* as input and the other half as output. More details about *PORT A* and *PORT B* can be found in the PIC18F4550 datasheet, pages 111 – 116 of the PIC reference manual [PIC].

In the program, all the data of output ports (*PORT A* y *PORT B*) are assigned to specific arrays: *Write\_buffer[4]*, *Temperature[8]*, *Humidity[8]*, *Pressure[8]*. All these arrays are concatenated in a Buffer of 32 characters (bytes) declared as *Cat\_Temp\_Hum\_Press\_Digital[32]*. The arrays: *op[12]*, *op2[12]*, *op3[12]*, *op4[12]*, are used to save data in string format.

Because buffers are used in excess, the program accumulates some blank spaces that result in garbage. The *op* arrays are used to remove the blank spaces. All the buffers are saved in the concatenate array called *Cat\_Temp\_Hum\_Press\_Digital*.

# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: [www.ijirccce.com](http://www.ijirccce.com)

Vol. 4, Issue 12, December 2016

The instruction `Hi d_Wri te(&Cat_Temp_Hum_Press_Di gi tal , 32) i s` used to send all the 32 bytes of the buffer to the PC through USB interface. The instruction `Del ay_ms( )` assigns a delay in mili-seconds (ms) to the program.

## B. COMPILATION

PICkit2 development programmer/debugger [9] and development board ET-PGMPIC USB were used to program the microcontroller PIC18F4550.



Figure 3. a) ET-PGMPIC USB Development Board (left). b) Connection of the microcontroller with the ET-PGMPIC USB Development Board (middle). c) Programming process (right).

When the Project is created using the microC wizard, the compiler creates various programs in different formats. One of these programs are in hexadecimal format (.hex). Before programming the microcontroller ET-PGMPIC USB development board needs to be connected to the computer using the USB cable. Then the microcontroller needs to be connected to the socket of the development board. Then, the program PICkit 2 verifies that PIC device is found. If the information doesn't appear, the family of microcontrollers PIC18F4550 must be selected in the **Device Family** menu. Then, Import Hex must be selected in menu **File** to search the hexadecimal file. Then, the button **Erase** must be pressed to delete the microcontroller's program. Finally, the **Write** button must be pressed to program the microcontroller.

## C. PROGRAMMING THE PC CLIENT INTERFACE

EasyHID [11] USB wizard, developed by Mecanique was used to develop a Project in Visual Basic 6.0 to work with the interface USB 2.0. In the first window of the wizard, the options **Company Name**, **Product Name** and **Serial Number** are optional. In the next window the options **Vendor ID** and **Product ID** used are VID = 4660 and PID = 1. The VID y PID are authentic numbers assigned by the USB implementers (USB Implementers Forum, Inc.) and has a cost ([www.usb.org](http://www.usb.org)). In the next window the options are: **Polling (Input)**- data required by the computer of the USB device (default: 10 ms); **Polling (Output)** - data that the computer sends to the USB device (default: 10 ms); **Bus Power**- maximum power consumed by the USB device (Default: 50 x 2 mA = 100mA); **Buffer (Input)**- buffers that are sent from the USB device to the computer, maximum of 64 bytes (32 bytes were used); **Buffer (Output)**- buffers that outs from the computer to the USB device, maximum of 64 bytes (32 bytes were used). In the next Windows the following options were chosen: **Project Name** (optional), **Location** (hard drive), **Compiler (microEngineeringLabs PICBASIC PROTM)**, **Microcontroller (18F5450)**, and **Application Compiler (Microsoft Visual Basic 5.0)**. The project created using EasyHID includes a module called HIDDLLInterface (mcHIDInterface.bas), Visual Basic uses this code to manage instructions of the USB interface. In addition, the created project includes a MainForm (FormMain.frm) to include objects to interact with the USB device.

The microcontroller was programed to send the buffer of 32 bytes to the computer through the USB port. The client interface program reads the data and presents it in the texboxes to the user. In the same manner, the computer sends data to the microcontrollers using buffers. The `BufferIn()` array was declared to read data and the `BufferOut()` array was declared to send data. Each element of the buffer represent an ASCII character. For instance, the temperature 23.5 °C is stored as `BufferIn(1) = 2`, `BufferIn(2) = 3`, `BufferIn(3) = .` and `BufferIn(4) = 5`. This method is used to get the data sensed by each sensor. Data of the other sensors are stored in the next buffer elements. Digital data of the PORT B has 8 bits, for this reason PORT B was taken as a character or an element of the array of buffers. The following conversion of the measures coming from the microcontroller to different units are made. For temperature  $^{\circ}\text{F} = (^{\circ}\text{C} * 1.8) + 32$  and  $^{\circ}\text{K} = ^{\circ}\text{C} + 273$ , and for pressure  $1 \text{ kPa} = 10 \text{ millibars}$ ,  $1 \text{ kPa} = 9.869 \times 10^{-3} \text{ atmand}$   $1 \text{ kPa} = 0.14503 \text{ PSI}$  were used.

An additional form as part of the client interface program was implemented to send the received data from the acquisition device to the database server. This program can control different devices through the internet using the

# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: [www.ijirccce.com](http://www.ijirccce.com)

Vol. 4, Issue 12, December 2016

database. In this way the system can detect the temperature, humidity, light on/off and movement in different rooms. This program has the capability to send messages to the server to execute some events, such as to turn on or off a light bulb. The figure 4.a shows the implemented client interface program.

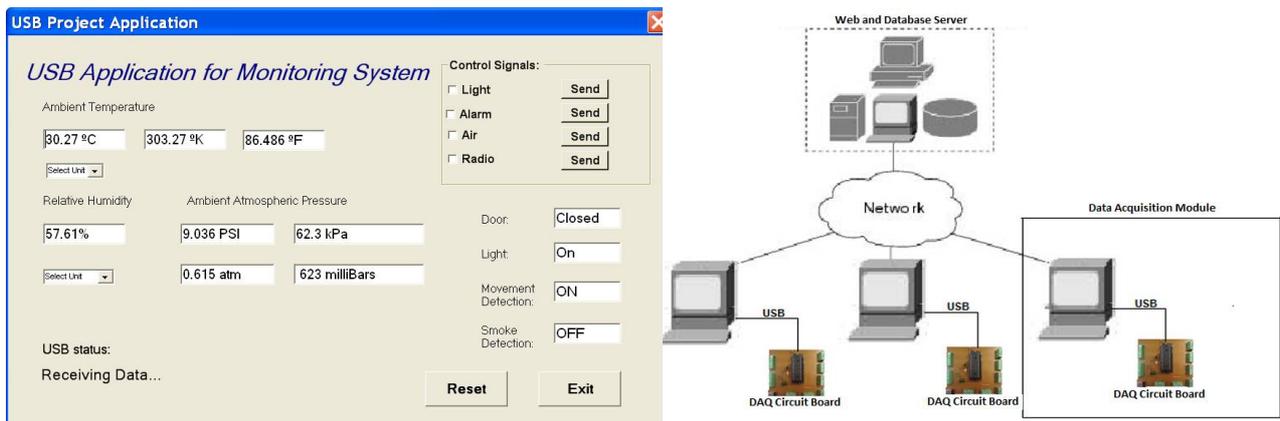


Figure4. a) Client interface data acquisition program.b) Prototype of the smart building system

## D. DATABASE IMPLEMENTATION

The structure of the database was developed using SQL server.Each table of the database was created to store data for each monitored sensor: temperature, pressure, humidity, smoke detector, movement detector, state of the door's switch, and the state of the presence or absence of light. Microsoft SQL Server and SQL Management Studio Express was used to create this database. Figure 5.a shows the diagram how the database is integrated in the prototype system.

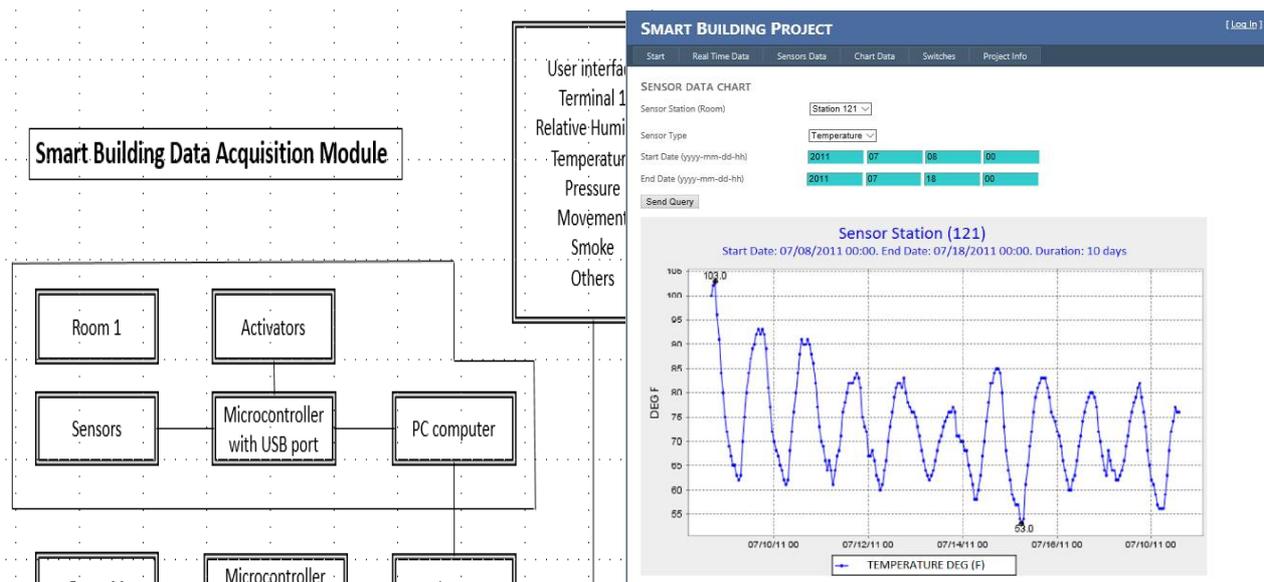


Figure5. a) Prototype of the Smart building (left). b) Web page interface (right)

## E. WEB INTERFACE

A web interface program was implemented in ASP.NET to read the database and to show the stored data in a time specified by the user. This program shows the data in tabular and chart form (Figure 5.b). The Chart tools of ASP.NET were used to implement chart form. Action events such as to turn on the light bulb can be sent using sockets.

# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: [www.ijircce.com](http://www.ijircce.com)

Vol. 4, Issue 12, December 2016

Component w3sockets of ASP.NET can be used to send these action events as a character string from the server to the client interface program. Once the message arrives to the client interface program, it generates an event to send the message to the module which turn on or off the light bulb. For this purpose threads are used to generate an event in the client interface program. Component w3sockets need to be registered in the same folder where the files of the IIS server are located to publish the application. This program and the database were designed to store sensed data connected to the board and these data are displayed constantly in the client interface program. At each hour the program generates an event timer. The data displayed in the client interface program is used to generate and execute an SQL query to store data in the respective table and an instance is created in the report table for each hour (hourly report). When the report button is pressed a new windows appears to display the stored data in a specified hour. When the windows appears the drop down list must be completed with the desired hour and date.

## V. SIMULATION RESULTS

### A. PROTOTYPE DATA ACQUISITION MODULE

The data acquisition module is composed of the data acquisition circuit board and the client PC. Figure 6.a shows the prototype data acquisition circuit board. Figure 6.b shows the schematic of the developed circuit board including the microcontroller. The analogue inputs are used to measure temperature, relative humidity and atmospheric pressure. Also, the prototype displays the digital outputs which are used to execute commands to activate the radio, the light bulb and, air conditioner, etc. Digital inputs are used to measure the presence of movement, smoke, light and door state. This prototype must be enclosed in a container box. In this box is installed an exclusive OR to have a manual method to activate the outputs locally making the system transparent for the user. In this manner, the user can manipulate the light, the radio, and the alarms without the need of a computer.

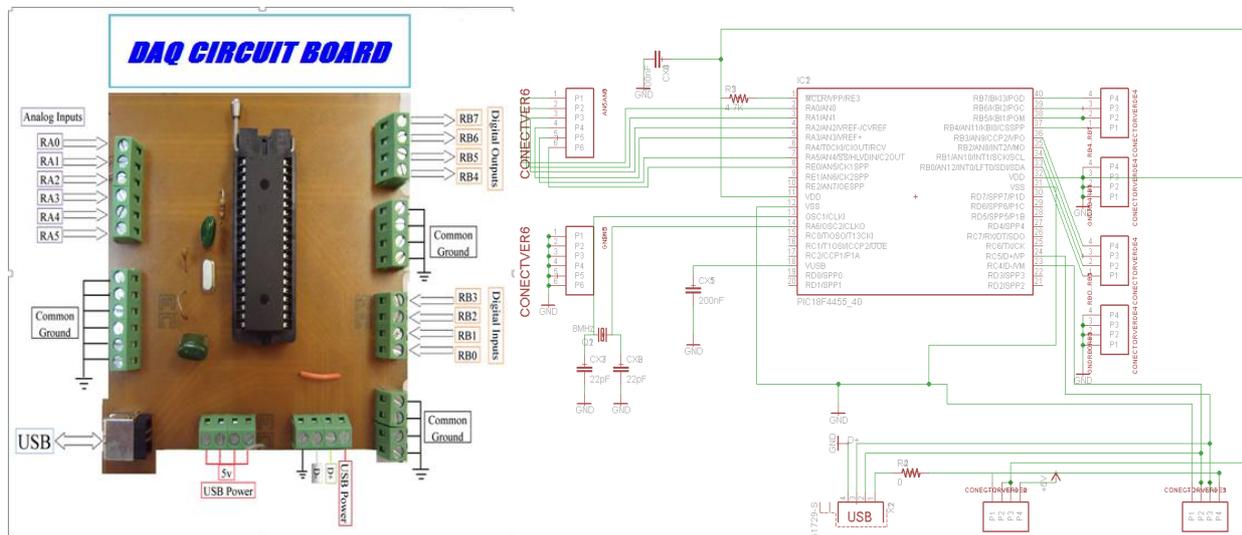


Figure6. a) Prototype data acquisition circuit board. b) Schematic of the circuit with the microcontroller.

### B. DESCRIPTION OF THE SYSTEM.

The implemented system gives intelligence to a building by adding several data acquisition modules connected over Ethernet to a server. Each data acquisition module is composed of sensors, activators, and a microcontroller. The intelligence is added in blocks, room by room. Each room has a data acquisition module that uses sensed data in the process of activating outputs giving intelligence to the room and the entire building. The data read by the circuit boards and sensors are collected by the PIC 18F4550 microcontroller which send these data to the client computer through the USB port. The client program sends this data to sever through the Ethernet port. These sensors measures temperature, humidity, and pressure, the activation of door's sensors, light, movement and smoke detectors.

# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: [www.ijircce.com](http://www.ijircce.com)

Vol. 4, Issue 12, December 2016

Each data acquisition module is capable of turning on or off an alarm, a radio, a light bulb or an air conditioner. For this purpose, the client program send a command to the microcontroller through USB port. The microcontroller uses the port B as an output port. The port B is a parallel port with 8 bits. Four of these bits are for output and the other four are for input. A client program implemented in Visual Basic manage the information in the personal computer. A program implemented using the C language manage the information in the microcontroller. A web application implemented in ASP.NET manages the entire smart building system.

## VI. FINDINGS AND CONCLUSIONS

### A. RESULTS

A prototype circuit boards with different digital and analogue sensors was implemented. A program in C programming language was implemented for the microcontroller PIC 18F4550 to interact between the sensors and actuators with the PC client interface program. This interface program was implemented in Visual Basic to communicate data between the client computer and the data acquisition module through the USB port. A database server using Microsoft SQL Server 2005 was implemented. This database has tables to store data coming from client interface program.



Figure 12. Final design of the prototype circuit board

Figure 12 shows the final design of the circuit board inside the box connected to the input commands circuit board. This board allows to the user to control digital outputs locally in addition to the client computer. For instance, the client interface program controls the light bulb to turn on or off remotely. But a local control is needed to allow the user to turn on or off the light bulb. This command circuit board has four exclusive OR gates capable to execute the necessary logic to turn on or off the light bulb locally. The system has four logic local controls. The smoke and movement sensors (Figure 6.a) are mounted in the top of the room because they work better in that place. The temperature and humidity sensors are two sensors in one. In one side the temperature is read and in the other side the relative humidity is read. The light bulb is driven by a solid state relay to control the necessary voltage to the bulb. Only one solid state relay is able to manage twelve light bulbs. In this prototype a relay was used to manage the light bulbs, turning on or off the radio, air conditioning and alarms. The switches are used to detect if the door is open or not. Also, it is used to activate the logic commands coming from the logic OR circuit. The light sensor is used to detect if the light bulb is turning on or off, it is the reason why it was used in digital mode. The house acquires more intelligence when more sensors are added to it. Many experiments was made to proof the operation of the prototype system. Finally sensor's voltage adjustment to convert the raw data in standard units for the user was performed.

### B. CONCLUSIONS.

A complete software and hardware system was developed to monitor physical environment variables and actuators. This system is composed of a data acquisition module connected to a client PC. This prototype system was implemented in C, Visual Basic, and SQL. This system can be used for advance technologies like intelligent buildings, security monitoring, natural disaster prediction and handling. It can be accomplished with only a few software libraries



# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: [www.ijirccce.com](http://www.ijirccce.com)

Vol. 4, Issue 12, December 2016

and a small sensor board. This system can be enhanced to handle a large number of monitoring system devices without need of complicated software modifications. The software developed could provide to the scientific community an open tool to create complex and advanced ideas with an outstanding layer of abstraction.

## C. FUTURE WORK

As a future work other sensors and detector can be added to the developed system. These sensors could be: Gas sensor CH<sub>4</sub> (Metanol), CO (Carbon Monoxide), CO<sub>2</sub> (Carbone Dióxide), LPG (Propane), C<sub>2</sub>H<sub>5</sub>OH (Alcohol/ Benzina), ultrasonic sensor, liquid level sensor, sound impact sensor, strain gauge or weight measure, metallic detector and other. This work will be the base for a smart building system. It will require to be implemented a database server to supply sensed data via Internet. A web application will remotely control these peripheral devices over TCP/IP protocol. It will be part of a smart building project, which is currently in development.

## ACKNOWLEDGMENTS

This project was sponsored by the University of Puerto Rico at Bayamón, NASA, and the Alliance for Minority Participation, and NFS Grant CNS-0837556. We also thanks to students from Electronics Department for collaborating in this project: José R. De León Cacho, and Christian A. Arroyo Ovalles, Jesús M. Rodríguez Carriles and the student Marcel Rivera Ayuso from Computer Science Department. This document was edited by Prof. Antonio Huertas from Computer Science Department of University of Puerto Rico. Special thanks to Florian Leitner Fisher for provide us the HID library.

## REFERENCES

1. X. Zhenghua, C. Guolong, H. Li, Q. Song, L. Hu, C. Lei, M. Youwen, X. Yexiang, *The Smart Home System Based on the Iap15f2k61s2 and Gsm*, International Journal On Smart Sensing And Intelligent Systems, Vol. 7, No. 4, pp. 1789-1806, 2014.
2. F. Shawki, M. E. Dessouki, A. I. Elbasiouny, A. N. Almazroui, F. M. R. Albeladi, *Microcontroller Based Smart Home With Security Using Gsm Technology*, International Journal of Research in Engineering and Technology, Vol. 4 No. 06, pp. 20-28, 2015.
3. O. Edward, O. Olabode, *A Microcontroller Based Building Automation System for real time Sensing and Control*. International Journal of Innovation and Scientific Research, Vol. 2 No. 2, pp. 275-280, 2014.
4. Ibrahim Dogan, *Advanced PIC Microcontroller Projects in C: from USB to RTOS with the PIC18F Series*, Newnes Publications, 2008.
5. Z. Abdulzahra, R. S. Kawitkar, *Implementation of Smart Home Control by Using Low Cost Arduino & Android Design*. International Journal of Advanced Research in Computer and Communication Engineering, Vol. 5, No. 2, pp. 248-256, 2016.
6. I. Kaur, *Microcontroller Based Home Automation System with Security*. International Journal of Advanced Computer Science and Applications, Vol. 1, No. 6, pp. 60-65, 2010.
7. B. Bhaskar, R. Swarnalatha. *Smart Home Automation System Using AVR Microcontroller*. International Journal of Advanced Technology in Engineering and Science, Volume No.03, Issue No. 02, pp. 234-242, 2015.
8. R. Panna, R. Thesrumluk, C. Chantrapornchai, *Development of Energy Saving Smart Home Prototype*. International Journal of Smart Home, Vol. 7, No. 1. pp. 47-66, 2013.
9. Microchip Technology Inc, *PIC18F2455/2550/4455/4550 Data Sheet*. 28/40/44-Pin, High Performance, Enhanced Flash, USB Microcontrollers with nanoWatt Technology, 2007.
10. MikroC, *MicroC Compiler*, Available: <http://www.mikroe.com/>, [Accessed : 10 – Oct. 2016].
11. Mecanique, *Easy HID*, Available: <http://www.mecanique.co.uk/products/usb/easyhid.html>, [Accessed : 10 – Oct. 2016].
12. G. Gupta, R. Deshmukh, P. Thorve, K. Bhande, G. Gawas, *Microcontroller & Wireless Based Smart City*. International Journal on Recent and Innovation Trends in Computing and Communication, Vol. 4 No. 1, pp. 87-89, 2016.
13. L. Schor, P. Sommer, R. Wattenhofer, *Towards a zero-configuration wireless sensor network architecture for smart buildings*. BuildSys '09 Proceedings of the First ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings, pp. 31-36, 2009.
14. S. Levente, *Improving energy usage efficiency in web enabled smart buildings*, International Journal of Advanced Research in Computer Engineering & Technology, Vol. 1, No. 9. pp. 169-174, 2012.

## BIOGRAPHY



M. S. Ismael Sanchez Rios received B.S. in Electrical Engineering at University of University of Puerto Rico at Mayaguez Campus in 1986. He received M. S. in Electrical Engineering at Purdue University in 1992. M.S. Sanchez is with the department of Electronic, University of Puerto Rico at Bayamón, San Juan, PR, 00959 USA. His research interest is in robotics, microcontrollers, and digital and analog electronics.



ISSN(Online): 2320-9801  
ISSN (Print): 2320-9798

# International Journal of Innovative Research in Computer and Communication Engineering

*(An ISO 3297: 2007 Certified Organization)*

Website: [www.ijircce.com](http://www.ijircce.com)

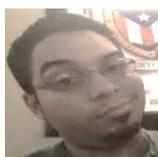
**Vol. 4, Issue 12, December 2016**



Dr. Elio Lozano received B.S. in mathematics at National University of San Antonio Abad of Cusco, Perú in 2000. He received M.S. in Scientific Computing and Ph.D. in Computer and Information Science and Engineering at University of University of Puerto Rico at Mayaguez Campus in 2003 and 2006 respectively. Dr. Lozano is with the department of Computer Science, University of Puerto Rico at Bayamón, San Juan, PR, 00959 USA. He has developed several software and firmware for different architectures.



PhD. Student Victor Santos Uceta received B.S. in Computer Science of the University of Puerto Rico at Bayamón. He is currently pursuing a Ph.D. degree in Computer Science at Purdue University with focus on Distributed Systems, Software Engineering, and applications of machine learning in interdisciplinary fields.



Alexander Ortiz Lozada received B.S. in Electronics Technology of the University of Puerto Rico at Bayamón. He is currently pursuing a B.S. degree in Electrical Engineering at University of Puerto Rico at Mayaguez. He is focused on microcontrollers and controls.