



Arduino Applications for Smart Cities

Shruti S. Gadgil^{1*}, Ruby R. Lobo²

^{1*}Dept. of Computer Science and Engineering, Walchand Institute of Technology, Solapur

²Dept. of Electronics and Telecommunication, D.Y. Patil Institute of Engineering & Technology
Pimpri, Pune,

Available online at: www.ijcseonline.org

Received: May/26/2016

Revised: Jun/02/2016

Accepted: Jun/12/2016

Published: Jun/30/2016

Abstract—As India’s population continues to grow, more citizens will move to cities leading to massive urbanization. India needs to adopt smarter ways to manage complexities by developing smart cities. The Smart Cities projects involve creating an innovation network between governments and academic partners leading to excellence in the development. Devices like Drones are going to be predominant in achieving tasks to be done in these smart cities. Arduino as a supporting software along with required hardware support is a bliss to develop small applications required for smart cities. It is open source and hence supporting to use. Arduino is a company of a different type It is a project and user community that designs and manufactures microcontroller-based kits for building digital devices and interactive objects by sending a set of instructions to the microcontroller that can sense and control objects in the physical world. In this paper we present the Arduino environment and its use to build smart city applications.

Keywords- Arduino;microcontroller;programming;UNO;Integrated Development Environment;Smart City;Serial port

I. INTRODUCTION

India’s population is growing rapidly, more migrations are seen from villages to the city. It is estimated that by the year 2050, the number of people living in Indian cities will touch 843 million. To accommodate this massive urbanization, India needs to adopt smarter ways to manage complexities, reduce expenses, increase efficiency and improve the quality of life. Investments of about USD 1.2 trillion will be required over the next 20 years across areas like transportation, energy and public security to build smart cities in India for smart energy, smart environment and smart governance [1,2]. Indian Budget of 2014-15 has USD 1.2 billion allocated for smart cities and FDI norms relaxed , USD 83 million allocated for Digital India Initiative, PPP Model to be used to upgrade infrastructure in 500 urban areas, Smart City projects to create 10-15% rise in employment. Ministry of Urban Development has plans to develop 2 smart cities in each of India’s 29 states [3,4].

The Smart Cities projects involve creating an innovation network between governments and academic partners leading to excellence in the development and take-up of e-services and e-government, and which is setting new standards for e-service delivery. Since establishing e-services have to be optimized using automation, devices like drones are going to be predominant in achieving tasks to be done in these smart cities [5,6,7].

Arduino is an open-source prototyping platform based on easy-to-use hardware and software. The project is based on microcontroller board designs, manufactured by several vendors, using various microcontrollers. These systems provide sets of digital and analog I/O pins that can be interfaced to various expansion boards ("shields") and other circuits. The boards feature serial communications interfaces, including USB on some models, for loading programs from personal computers. For programming the microcontrollers, the Arduino project provides an integrated development environment (IDE) based on the Processing project, which includes support for the C and C++ programming languages.

The first Arduino was introduced in 2005, aiming to provide an inexpensive and easy way for novices and professionals to create devices that interact with their environment using sensors and actuators. Common examples of such devices intended for beginner hobbyists include simple robots, thermostats, and motion detectors.

Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output – activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the Arduino programming language (based on Wiring), and the Arduino Software (IDE), based on Processing.

Over the years Arduino has been the brain of thousands of projects, from everyday objects to complex scientific instruments [8]. A worldwide community of makers - students, hobbyists, artists, programmers, and professionals -

*Corresponding Author:
Shruti. S. Gadgil
e-mail: gadgil.shruti@gmail.com , Tel.: 0217-2652700

has gathered around this open-source platform, their contributions have added up to an incredible amount of accessible knowledge that can be of great help to novices and experts alike.

Arduino was born at the Ivrea Interaction Design Institute as an easy tool for fast prototyping, aimed at students without a background in electronics and programming. As soon as it reached a wider community, the Arduino board started changing to adapt to new needs and challenges, differentiating its offer from simple 8-bit boards to products for IoT applications, wearable, 3D printing, and embedded environments [9]. All Arduino boards are completely open-source, empowering users to build them independently and eventually adapt them to their particular needs. The software, too, is open-source, and it is growing through the contributions of users worldwide.

The flow of the paper includes a literature review in section II followed by a description on Arduino board in section III ,Arduino coding in section IV followed by a conclusion of the paper in section V ,an Acknowledgement and References used for the paper.

II. LITERATURE REVIEW

A. M. Gibb in his work examines the history of the Arduino and its precursors, starting with a history of the electronic computer and how it branched into the development of the first microcontrollers used in business, science, and military sectors; and following with a discussion of the needs and events that drove the development of a new group of microcontrollers targeted at artists, designers, and hobbyists [10].

D. Mellis, M. Banzi, D. Cuartielles, and T. Igoe describes a new Beagle Board-based platform for teaching and practicing interaction design for musical applications [8].

J. Boxall says Arduino is a cheap, flexible, open source microcontroller platform designed to make it easy for hobbyists to use electronics in homemade projects. With an almost unlimited range of input and output add-ons, sensors, indicators, displays, motors, and more, the Arduino offers you countless ways to create devices that interact with the world around you [11].

Badamasi, Y.A. analyzes the working principle of an arduino. He remarks that many people try to use the arduino because it makes things easier due to the simplified version of C++ and the already made Arduino microcontroller that can programme, erase and reprogramme at any given time. He discusses the hardware components used in the arduino board, the software used to programme it (Arduino board) with the guide on how to write and construct your own projects, and a couple of examples of an arduino project, This gives the overall view of an arduino [12].

Murtaza Hussain Shaikh investigates how Adriano fits for creative and interactive artwork installations using an artwork installation in the campus [13].

Robinson P. Paul, Ghansyam B. Rathod, Vishwa R. Trivedi, Punit V. Thakkar emphasized on the Persistence Of Vision technology in which energy is the main factor in designing all the applications, maximum and efficient use of the energy is very important. The display has many advantages over a traditional CRT, LCD or LED display, like power savings, less complexity, easy configuration, attractiveness etc. To overcome the drawback of old processor we have decided to implement the same display atop a new and advanced microprocessor, the Arduino duemilanove. This platform brings with it newer coding and a different understanding of peripherals. ARDUINO INTERFACE BOARDS provided them with a low-cost, easy-to use technology to create the project [14].

Sarik, J. ; Kymissis, I. explain experimental setups with Lab kits using the Arduino prototyping platform. Saima Maqbool*, Ulya Sabeel, Nidhi Chandra, Rouf-Ul-Alam Bhat proposed a smart traffic light control and congestion avoidance system during emergencies based on Arduino. The Arduino used in the system is Arduino duemilanove atmega which is 328P family based. Yashi Mishra, Gaganpreet Kaur Marwah, Shekhar Verma proposed a secure system that provides information about authorized and unauthorized persons [1]. This system was an Arduino Based Smart RFID Security and Attendance System with Audio Acknowledgement [15].

III. ARDUINO BOARD

Arduino started in 2005 as a project for students at the Interaction Design Institute Ivrea in Ivrea, Italy [12,13]. At that time program students used a "BASIC Stamp" at a cost of \$100, considered expensive for students. Massimo Banzi, one of the founders, taught at Ivrea. The name "Arduino" comes from a bar in Ivrea, where some of the founders of the project used to meet. The bar, in turn, has been named after Arduin of Ivrea, who was the margrave of Ivrea and King of Italy from 1002 to 1014.

Colombian student Hernando Barragán created the Wiring development platform which served as the basis for Arduino [16]. Following the completion of the Wiring platform, its lighter, less expensive versions were created and made available to the open-source community; associated researchers, including David Cuartielles, promoted the idea. The Arduino's initial core team consisted of Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino, and David Mell

An early Arduino board with an RS-232 serial interface (upper left) and an Atmel ATmega8 microcontroller chip (black, lower right); the 14 digital I/O pins are located at the top and the six analog input pins at the lower right.

An Arduino board historically consists of an Atmel 8-,16-or32-bit.AVR (although since 2015 other makers' microcontrollers have been used) with complementary components that facilitate programming and incorporation into other circuits. An important aspect of the Arduino is its

standard connectors, which lets users connect the CPU board to a variety of interchangeable add-on modules known as shields. Some shields communicate with the Arduino board directly over various pins, but many shields are individually addressable via an ICserial bus—so many shields can be stacked and used in parallel. Prior to 2015 Official Arduinos had used the Atmel mega AVR series of chips, specifically the ATmega8, ATmega168, ATmega328, ATmega1280, and ATmega2560 and in 2015 units by other manufacturers were added [14]. A handful of other processors have also been used by Arduino compatible devices. Most boards include a 5 V linear regulator and a 16 MHz crystal oscillator (or ceramic resonator in some variants), although some designs such as the LilyPad run at 8 MHz and dispense with the onboard voltage regulator due to specific form-factor restrictions. An Arduino's microcontroller is also pre-programmed with a boot loader that simplifies uploading of programs to the on-chip flash memory, compared with other devices that typically need an external programmer. This makes using an Arduino more straightforward by allowing the use of an ordinary computer as the programmer. Currently, optiboot bootloader is the default bootloader installed on Arduino UNO.

At a conceptual level, when using the Arduino integrated development environment, all boards are programmed over a serial connection. Its implementation varies with the hardware version. Some serial Arduino boards contain a level shifter circuit to convert between RS-232 logic levels and TTL-level signals. Current Arduino boards are programmed via Universal Serial Bus (USB), implemented using USB-to-serial adapter chips such as the FTDI FT232. Some boards, such as later-model Uno boards, substitute the FTDI chip with a separate AVR chip containing USB-to-serial firmware, which is reprogrammable via its own ICSP header. Other variants, such as the Arduino Mini and the unofficial Boarduino, use a detachable USB-to-serial adapter board or cable, Bluetooth or other methods, when used with traditional microcontroller tools instead of the ArduinoIDE, standard AVR ISP programming is used.

The Arduino board exposes most of the microcontroller's I/O pins for use by other circuits. The Diecimila, Duemilanove, and current Uno provide 14 digital I/O pins, six of which can produce pulse-width modulated signals, and six analog inputs, which can also be used as six digital I/O pins. These pins are on the top of the board, via female 0.10-inch (2.5 mm) headers. Several plug-in application shields are also commercially available. The Arduino Nano, and Arduino-compatible Bare Bones Boar and Boarduino boards may provide male header pins on the underside of the board that can plug into solderless breadboards.

There are many Arduino-compatible and Arduino-derived boards. Some are functionally equivalent to an Arduino and can be used interchangeably. Many enhance the basic Arduino by adding output drivers, often for use in school-level education to simplify the construction of buggies and small robots. Others are electrically equivalent but change the

form factor, sometimes retaining compatibility with shields, sometimes not. Some variants use completely different processors, with varying levels of compatibility.

Arduino boards are available commercially in preassembled form, or as do-it-yourself kits [15]. The hardware design specifications are openly available, allowing the Arduino boards to be manufactured by anyone. Adafruit Industries estimated in mid-2011 that over 300,000 official Arduinos had been commercially produced, and in 2013 that 700,000 official boards were in users' hands. This board has a Microcontroller. Wikipedia says: A micro-controller is a small computer on a single integrated circuit containing a processor core, memory, and programmable input/output peripherals The important part for us is that a micro-controller contains the processor (which all computers have) and memory, and some input/output pins that you can control. (often called GPIO - General Purpose Input Output Pins) as shown in Fig.1



Figure 1. General Purpose Input Output Pins

An Arduino Uno board combines a micro-controller along with all of the extras to make it easy for you to build and debug your projects [10]. We will be use a breadboard . This is a relatively easy way to make circuits quickly. Breadboards are made for doing quick experiments. They are not known for keeping circuits together for a long time. When you are ready to make a project that you want to stay around for a while, you should consider an alternative method such as

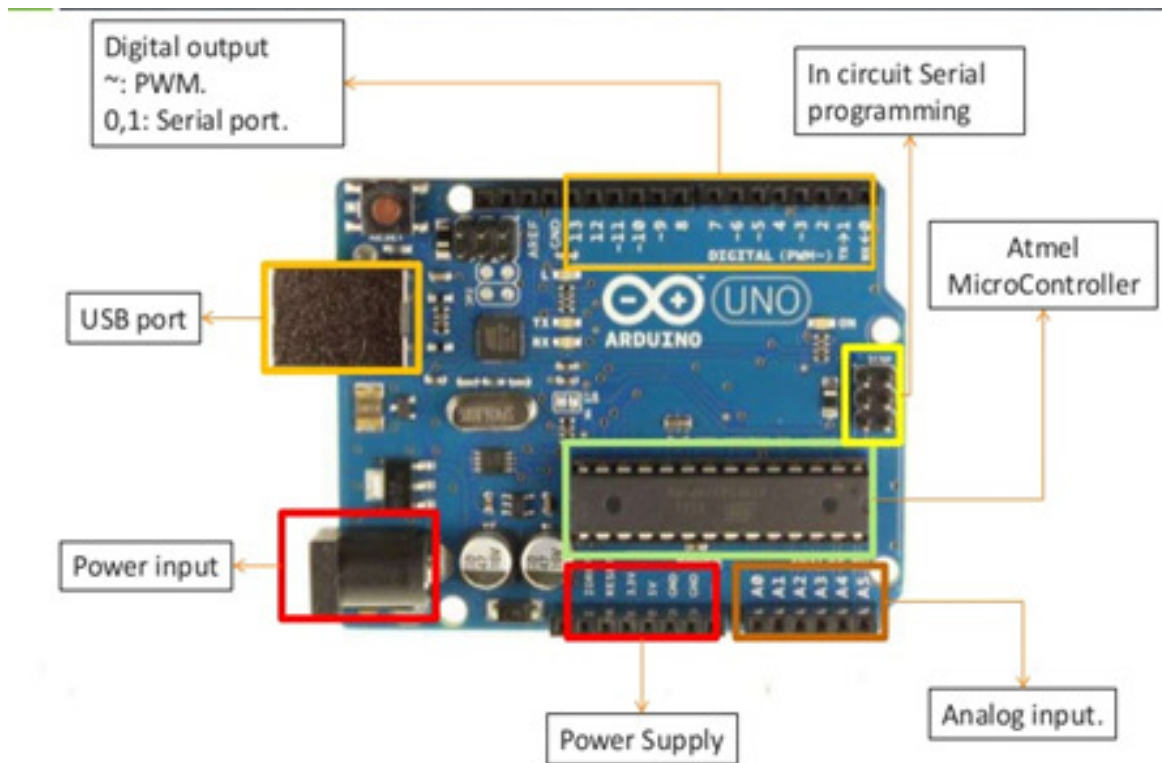


Figure 2. The

Arduino UNO

wire-wrapping or soldering or even making a printed circuit board (PCB)

The first thing you should notice about the breadboard is all of the holes. These are broken up into 2 sets of columns and a set of rows (the rows are 2 divided in the middle). The columns are named a, b, c, d, e, f, g, h, i, and j (from left to right). The rows are numbered 1 - 30. (from top to bottom). The columns on the edges do not have letters or numbers. The columns on the edges are connected from top to bottom inside of the breadboard to make it easy to supply power and ground. (You can think of ground as the negative side of a battery and the power as the positive side.) Power will be +5 volts. Inside of the breadboard, the holes in each row are connected up to the break in the middle of the board. For Example: a1,b1,c1,d1,e1 all have a wire inside of the breadboard to connect them. Then f1, g1, h1, i1, and j1 are all connected but a1 is not connected to f1. This may sound confusing now, but it will quickly come to make sense as we wire up circuits. Thus it is an open source hardware where anyone can get details of a design and modify it as per his requirement.

Different types of Arduino boards are as follows

- Arduino UNO.
- Arduino Mega
- Arduino LilyPad.
- Arduino BT
- Arduino Nano
- Arduino Mini

The basic components of the Arduino UNO are as shown in Fig.2 They are Power supply input, USB Port, Microcontroller, In circuit serial programming component and digital output serial port.

To install the software and hardware using Arduino UNO follow the following steps: Check out: <http://arduino.cc/en/Guide/HomePage>

1. Download & install the Arduino environment (IDE)
2. Connect the board to your computer via the USB cable
3. If needed, install the drivers
4. Launch the Arduino IDE
5. Select your board
6. Select your serial port
7. Open the blink example
8. Upload the program

A. The Integrated Development Environment (IDE)

You use the Arduino IDE on your computer (picture following) to create, open, and change sketches (Arduino calls programs “sketches”). Sketches define what the board will do. You can either use the buttons along the top of the IDE or the menu items. The IDE is shown in Fig.3.

1) Parts of the IDE(from left to right,top to bottom)

- a) *Compile* : Before the program “code” can be sent to the board, it needs to be converted into instructions that the board understands. This process is called compiling.
- b) *Stop*:This stops the compilation process.
- c) *Create new sketch*: This opens a new window to create a new sketch.
- d) *Open Existing Sketch*: This loads a sketch from a file on the user’s computer.
- e) *Save Sketch*: This saves the changes to the sketch one is working on.
- f) *Upload to Board* : This compiles and the transmits over the USB cable to the board.
- g) *Tab Button*: This lets one create multiple files in the sketch.
- h) *Sketch Editor*: This is where one writes or edit sketch.
- i) *Text Console*: This shows what the IDE is currently doing and is also where error messages are displayed.
- j) *Line Number*: This shows on what line number the cursor is on.



Figure 3. The Integrated Development Environment [17].

IV. ARDUINO CODING

An Arduino program is written and then compiled and uploaded to Arduino board as shown in Fig.4.

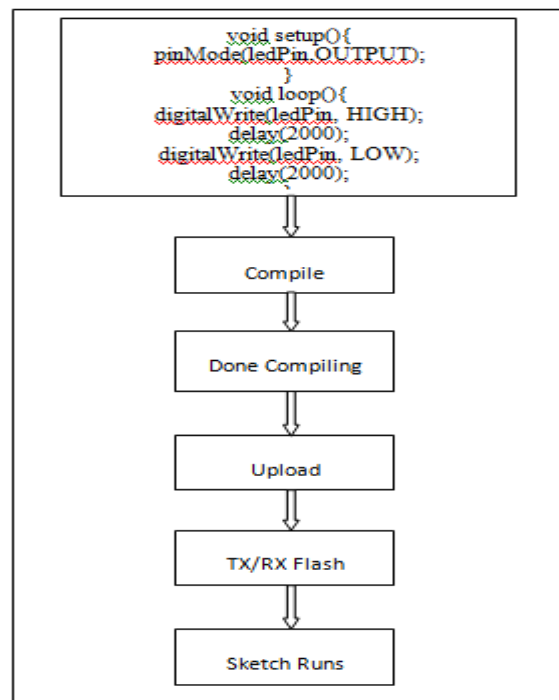


Figure 4. Flow of Arduino Program

Arduino programs may be written in any programming language with a compiler that produces binary machine code. Atmel provides a development environment for their microcontrollers, AVR Studio and the newer Atmel Studio.

The Arduino project provides the Arduino integrated development environment(IDE), which is a cross-platform application written in Java. It originated from the IDE for the Processing programming language project and the Wiring project. It is designed to introduce programming to artists and other newcomers unfamiliar with software development. It includes a code editor with features such as syntax highlighting, brace matching, and automatic indentation, and provides simple one-click mechanism for compiling and loading programs to an Arduino board. A program written with the IDE for Arduino is called a "sketch".

The Arduino IDE supports the C and C++ programming languages using special rules of code organization. The Arduino IDE supplies a software library called "Wiring" from the Wiring project, which provides many common input and output procedures. A typical Arduino C/C++ sketch consist of two functions that are compiled and linked with a program stub *main()* into an executable cyclic executive program:

- *setup()*: a function that runs once at the start of a program and that can initialize settings.
- *loop()*: a function called repeatedly until the board powers off.

After compilation and linking with the GNU tool chain, also included with the IDE distribution, the Arduino IDE employs the program *avrdude* to convert the executable code into a text file in hexadecimal coding that is loaded into the Arduino board by a loader program in the board's firmware.

A. Sample Program

A typical program for a beginning Arduino programmer blinks a light-emitting diode (LED) on and off. This program is usually loaded in the Arduino board by the manufacturer. In the Arduino environment, a user might write such a program as shown:

```
#define LED_PIN 13

void setup() {
  pinMode(LED_PIN, OUTPUT);    // Enable pin 13
  for digital output
}
void loop() {
  digitalWrite(LED_PIN, HIGH); // Turn on the LED
  delay(1000);                 // Wait one second (1000
  milliseconds)
  digitalWrite(LED_PIN, LOW);  // Turn off the LED
  delay(1000);                 // Wait one second
}
```

B. A Smart City Application using Arduino

Some of the applications [11][13][14] that can be done using Arduino are listed below

- Hand Gesture Recognition based on Shape Parameters.
- Home Automation system using Arduino. Ultrasonic Target Range Estimation using SO* Technology using Arduino.
- Petrochemical Level Indicator and Controller for Automation of cotton purification process in spinning mills Using Arduino.
- RFID security access control system using Arduino.
- RFID based electronic passport system for easy governance using Arduino.
- RFID based Airport Luggage security scanning system using Arduino.
- Arduino Based Industrial appliances control system by decoding dual tone multi frequency signals on GSM / CDMA network.
- Arduino Based Wireless DC Motor Speed & Direction Control using IR .
- Arduino Based Wireless Electrical Apparatus Control System using IR communication.
- Arduino Based Temperature Monitoring and Display on 7-segment Display.
- Arduino Based Foot step power generation system for rural energy application to run AC and DC loads.
- Solar based mobile charger for rural areas with battery voltage analyzer using Arduino.
- Solar highway lighting system with auto turns off on day time with LCD display using Arduino.
- IR remote based Human – computer interface mouse control for PC Using Arduino.

V. CONCLUSION

Arduino with its supporting Hardware and software environments can be used for various Smart city applications. It can be used in Drones for monitoring. It can be used in agricultural applications to give predictions to farmers. In section III the Arduino board is introduced with its specifications. Section IV introduces Arduino coding along with applications that can be developed for Smart cities.

ACKNOWLEDGMENT

We would like to express our sincere thanks to our guides from both our colleges who have taken great interest in motivating us, supervising our work and giving us suggestions from time to time. We also extend our thanks to the heads of our departments and principals of our college who have permitted us to carry out this work.

REFERENCES

- [1] Mishra Y.;Marwah G. K.;Verma S.,”Arduino Based Smart RFID Security and Attendance System with Audio Acknowledgement”, International Journal of Engineering Research & Technology (IJERT) Vol. 4, Issue 01,2015.
- [2] Khurshid, J.,”Military robots - a glimpse from today and tomorrow”Control, Automation, Robotics and Vision Conference,ICARCV ,Volume:1,IEEE,ISBN-7803-8653-1,2004
- [3] Smart Cities India 2015 Brochure,” Smarter Solutions for a Better Tomorrow” Pragati Maidan, New Delhi 20th -22th May, 2015.
- [4] Molina C. P.;Ortego R. G.;Pérez F. M., “Perspectives on Technological Developments Applied to Robotics” , The Robotics Divide,2013.
- [5] Lin Patrick ; Bekey George ; Abney Keith, “Autonomous Military Robotics: Risk, Ethics, and Design”, ADA534697, California Polytechnic State Univ SAN LUIS OBISPO,2008.
- [6] Howlader D.;Giordano J.,” Advanced Robotics: Changing the Nature of War and Thresholds and Tolerance for Conflict - Implications for Research and Policy”,The Journal of Philosophy, Science & Law Volume 13, pages(1-19), May 24 2013.
- [7] Da Silva F. B.;Scott S. D.;Cummings M. L., ” Design Methodology for Unmanned Aerial Vehicle (UAV), Prepared For Boeing Phantom Works”,HAL,May 2007.
- [8] Mellis D.;Banzi M.;Cuartielles D.;Igoe T., ” Arduino: An open electronic prototyping platform”, in Proc. CHI, vol. 2007.
- [9] Tariq Ahmad Sherasiya ; Hardik Upadhyay ; Hireen B. Patel , ” A survey: intrusion detection system for internet of things”, issn(p): 2278-9960; issn(e): 2278-9979 vol. 5, Issue 2, Feb - Mar 2016.
- [10] Gibb M, “ New media art, design, and the Arduino microcontroller: A malleable tool ”,PhD thesis, Pratt Institute, 2010.
- [11] Boxall J,”Arduino Workshop: A Hands-on Introduction with 65 Projects”,2013.
- [12] Badamasi, Y.A. ,”The working principle of an Arduino” 11th International Conference on Electronics, Computer and Computation (ICECCO) 2014.
- [13] Shaikh M. H.,”Arduino Tool: For Interactive Artwork Installations”, IJCTE,Vol.3(6): 760-769 ISSN: 1793-8201 DOI: 10.7763,Dec 2011.
- [14] Paul R. P.; Rathod G. B.;Trivedi V. R.;Thakkar P. V.(2014), Persistence of Vision Control Using Arduino, I.J. Intelligent Systems and Applications,vol.1,Page No (102-111) DOI: 10.5815/ijisa,Jan 2014.
- [15] Sarik, J.; Kymissis, I., “ Lab kits using the Arduino prototyping platform” , Publication , Page No.(1-5) , IEEE conference ,2010.

- [16] Maqbool S.;Sabeel U.; Chandra N.;Bhat R., “Smart Traffic Light Control and Congestion Avoidance System During Emergencies Using Arduino and Zigbee 802.15.4”, International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 6, June 2013.
- [17] Getting Started with Arduino , www.arduino.cc/en/Guide.

Authors Profile

Ms. Shruti S. Gadgil pursued Bachelor of Engineering in Computer Science and Engineering from University of Solapur, Solapur in 2013 and is pursuing Master of Engineering from Walchand Institute of Technology, Solapur affiliated to Solapur University . She is presently working for her dissertation on the topic “Associate rule mining for Social Network data using MapReduce” in the area of Data mining involving lot of intelligence under the guidance of Prof. Lobo L. M. R. J. Her areas of interest include designing of software and hardware, data mining and Genetic Algorithms. She had presented a paper in the seminar Recent Trends in Data Mining which is selected by IJCA for publication.



Ms Ruby R. Lobo is pursuing Bachelor of Engineering in Electronics and Telecommunication from D. Y. Patil Institute of Engineering and Technology, Pimpri, Pune affiliated to Pune University. Her areas of interest include designing hardware and software. She has designed a mini project on “Traffic light control using ARM processor Microcontroller”. Her areas of interest include system design, software programming, microcontroller and its applications.

