AUTOMATIC INTRUDER DETECTION SYSTEM WITH THE FACILITATION OF ANDROID APP

Sachin Goel¹, Sonya Khurana¹, Babita Rani¹ and Chander Mohan²
¹Scholars, ACE, Ambala, Haryana, India
sachingoel1993@gmail.com
sonyakhurana34@gmail.com
khilanbabita@gmail.com
²Assistant Professor, ACE, Ambala Cantt, India
ambitiouschander@gmail.com

Abstract—Intruder detection is a process which involves the detection of the arrival of an unwanted person/intruder in the house and the execution of necessary actions required to be performed so as to alert the members of the house and increase the security of the house simultaneously. It was observed that thereafter many such incidents took place in the absence of the family members/residents of that house and thus, there is a great need of such system which continuously monitor its surroundings and give a alert to the owner if some intrusion activity monitor by the system even if owner of the house is not near by the home. In this paper, we proposed an Automatic Intruder Detection System with the facilitation of Android App which results in ease of operation for user as the system is provided with Android App through which a person can get updates of his/her home even he/she is not at their residence.

Keywords—Intruder, Residents, Security.

1. INTRODUCTION

Intruder detection is a process which involves the detection of the arrival of an unwanted person/intruder in the house and the execution of necessary actions required to be performed so as to alert the members of the house and increase the security of the house simultaneously. It was observed that thereafter many such incidents took place in the absence of the family members/residents of that house and thus there was nobody to be alerted in those cases in which all the residents were out for sometime giving the intruders a fair chance to have access to the house and do what they planned to. Considering the fact that the same situation could occur any number of times with anyone a security system involving a cell phone calling system has been worked upon thus updating the system of intruder detection and giving a better solution to the residents of a house.

As the name ‘AUTOMATIC INTRUDER DETECTION CALL’ suggests, this project will include the process of detection of an incoming intruder and simultaneously making an automatic call on the cell phone(s) of the residents giving them a chance to call the local police and neighbors as well. It would also include a simultaneous increase in the protection of the house along with an alarm alert. Section II includes the major researches related to core of this article. Section III briefly introduced technology used for hardware potion of this work. Section IV exhibits the software part of the work including the android application development phase. Section V describes the Algorithm used. In Section VI, we introduced results and conclusion of the proposed system. Finally, Section VII concludes this paper.

2. LITERATURE SURVEY

Yuansheng Liu [4] had proposed a sample house environment monitor and control system that is one branch of the Smart home is addressed in this paper. The system is based on the embedded system and can act as a security guard of the home. The system can monitor the temperature, humidity, gas density, water immersion of the house and have infrared sensor to guarantees the family security. The system also has network and telephone connection to receive the owner's command and send the alert to the owner. The whole system includes a main control unit and input/output unit. The main chip of the main control unit is the S3C44B0X that is a 16/32 bits RISC processor and based on ARM7 core. In the main control unit, the use of 5 inch LCD and the touch screen provide a well user interface for handlers, the LAN/telephone interface provide the tools to communication. The µC/OS-II software core manages the whole unit work as a whole system. The input unit includes many sensors and its circuit, the information from the input unit is a base of the main control unit. The output is the action part of the main control unit it drives the alert and the switch of the electric appliance.
Sachin Kishor Khadke [2] had proposed Home Security System Based on Android Smartphone. A user logs into the smart Android phone interface, and clicks the buttons gently to send message commands from the GUI which will be transmitted to home information center through the GSM network. Then the AVR ATmega processor recognizes the specified command, and controls the home appliance switches in the wireless radio frequency manner to achieve remote control of appliances ultimately. This research focuses on the design of Android terminal, the communication between PIC and GSM module, the realization of the wireless module device’s driver, the difficulty in supplying the appropriate low-voltage DC for MCU and wireless module just by a single live wire. The users can manipulate appliances anywhere, anytime, letting our houses become more and more automated and intelligent. There are some problems in the PC monitor terminal, such as its great bulk, inconvenience to carry, high cost, limited monitoring range and so on. Therefore, it’s a good choice to design a terminal based on phone.

Megalingam et al. [3] had proposed a Low Power Intelligent, Wireless, and Home Security System (IWHSS) for Elders. The elders can avail the services of this IWHSS, conveniently within their room. If the elder is uncomfortable with a stranger trying to get into the home, services of IWHSS can be availed. The strangers can be identified and in emergency situations where the elder feels the necessity of the external help, the police and relatives can be informed via IWHSS. In addition to informing the police, the system also activates an emergency alarm to alert the neighbors. The elders can also see the stranger at the door and make a conversation with them. The system uses GPRS Modem which is interfaced directly with PIC 16F877A microcontroller to make the system power efficient. We have used low cost RF modules, PIC 16F877A microcontroller, IR-TSOP modules and Image Capture module as part of the hardware implementation of the system. The simulation software’s used were MPLAB IDE and PROTEUS.

Narayanan et al. [1] had proposed Wireless Home Security System using PIC Microcontroller. This research introduces the intelligent home security system (IHAM) which is developed using PIC microcontroller with the Zig-Bee wireless communication technology, speech recognition technique and GSM network technology that control the home appliance. The automation centers on recognition of voice commands and uses low-power RF Zig-Bee wireless communication modules which are relatively cheap. The home automation system is used to control all lights and electrical appliances in a home or office using voice commands with help of HM2007 chip that is widely used for such appliances. The proposed system gives the overall framework of hardware and software design, and describes ways to implement the system. The paper also explains the security system for fire hazards that may occur through smoke sensor and GSM Module that is controlled by the same controller that sends the SMS to the user if the smoke is detected.

Palaniappan et al. [6] had proposed GSM Based Home Security System. The aim of this project is to design an embedded system for remote monitoring of the laboratory environment. The remote monitoring the laboratory and its building is necessary for safety and security purpose, which also help us to know the environmental status of the laboratory. The environmental parameters inside the laboratory, such as presence of alcohol, gas and fire can be detected using respective sensors and the sensed data are then transferred to the microcontroller. The microcontroller takes the control action of activating an alarm whenever the presence of these parameters is found. In turn, the Voice alarm and alert message as SMS through GSM are also sent to the remote area. The advantage of this automated detection and alarm system is that, it offers faster response time and accurate detection during an emergency.

Chuah et al. [8] had proposed A Multilevel Home Security System (MHSS). MHSS is basically a multilevel security system which consists of different sensor nodes as the input elements while the output elements react to the signal received from the input elements. The sensor nodes consist of a thief alarm, presence detecting circuit and the break-in camera. A UART is applied as the communication tool between the hardware and the computer. A graphic user interface (GUI) is developed and configured which enables the function of capturing images and sending emails. The captured images are delivered to the house owners and the police forces. The task is performed in order to prevent the thieves’ invasion.

Shanmugasundaram et al. [5] had proposed Intelligent Smart Home System. In this paper a low cost, low power smart home system using ZIGBEE, GSM, sensors (smoke, IR motion sensors) and RFID has been presented. ZIGBEE is a new short distance, low data rate wireless network technology. It is built on the IEEE
802.15.4 low rate wireless Personal Area Network standard. PIC16F877A controller is used in a predominant way because it is rich in peripherals and hence many devices can be interfaced at ease, it is also very cheap and can be easily assembled and programmed. The PIC controller controls the devices and sends the sensor values to the PC via ZIGBEE module. Although Bluetooth is better than ZIGBEE for transmission rate, ZIGBEE has lower power consumption. Hence, ZIGBEE is generally used for 24 hours monitoring of communication transmission systems. At the PC terminal, LABVIEW is used to control the GSM and ZIGBEE operations. The system can detect fire and theft situations and reacts accordingly by turning on the buzzer and sending alert SMS through GSM module. This system also monitors the building occupants and saves the energy by switching off certain devices like fan and lights when the room is unoccupied.

3. TECHNOLOGY USED

ATmega328 Arduino

Today the ATmega328 is commonly used in many projects and autonomous systems where a simple, low-powered, low-cost micro-controller is needed. Perhaps the most common implementation of this chip is on the popular Arduino development platform, namely the Arduino Uno and Arduino Nano models.

Basic Architecture

Figure 3.1: ATmega 328 IC [10]

Basic architecture of ATmega 328 is shown in figure 3.1. These are based on advanced RISC architecture. RISC stands for Reduced Instruction Set Computing. Today the ATmega328 is commonly used in many projects and autonomous systems where a simple, low-powered, low-cost micro-controller is needed. Perhaps the most common implementation of this chip is on the popular Arduino development platform, namely the Arduino Uno and Arduino Nano models.

Figure 3.2: Basic Architecture of ATmega 328

Pin Description

Pin description is given in figure 3.13. Atmel’s ATMEga328P 8-Bit Processor in 28 pin DIP package.

VCC
Digital supply voltage

GND
Ground

Port B (PB7:0) XTAL1/XTAL2/TOSC1/TOSC2

Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins
that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running. Depending on the clock selection fuse settings, PB6 can be used as input to the inverting Oscillator amplifier and input to the internal clock operating circuit. Depending on the clock selection fuse settings, PB7 can be used as output from the inverting Oscillator amplifier. If the Internal Calibrated RC Oscillator is used as chip clock source, PB7.6 is used as TOSC2.1 input for the Asynchronous Timer/Counter2 if the AS2 bit in ASSR is set.

![Figure 3.3: Pin Diagram of ATmega 328](10)

**Port C (PC5.0)**

Port C is a 7-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The PC5.0 output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active.

**PC6/Reset**

If the RSTDISBL Fuse is programmed, PC6 is used as an I/O pin. Note that the electrical characteristics of PC6 differ from those of the other pins of Port C. If the RSTDISBL Fuse is un-programmed, PC6 is used as a Reset input. A low level on this pin for longer than the minimum pulse length will generate a Reset, even if the clock is not running. Shorter pulses are not guaranteed to generate a Reset.

**Port D (PD7.0)**

Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running.

**AVCC**

AVCC is the supply voltage pin for the A/D Converter, PC3:0, and ADC7:6. It should be externally connected to VCC, even if the ADC is not used. If the ADC is used, it should be connected to VCC through a low-pass filter. Note that PC6.4 use digital supply voltage, VCC.

**AREF**

AREF is the analog reference pin for the A/D Converter.

**ADC7:6 (TQFP and QFN/MLF Package Only)**

In the TQFP and QFN/MLF package, ADC7:6 serve as analog inputs to the A/D converter. These pins are powered from the analog supply and serve as 10-bit ADC channels.
Specifications of ATmega 328 Arduino

The Atmel 8-bit AVR RISC-based microcontroller combines 32 KB ISP flash memory with read-while-write capabilities, 1 KB EEPROM, 2 KB SRAM, 23 general purpose I/O lines, 32 general purpose working registers, three flexible timer/counters with compare modes, internal and external interrupts, serial programmable USART, a byte-oriented 2-wire serial interface, SPI serial port, 6-channel 10 bit converter (8-channels in TQFP and QFN/MLF packages), programmable watchdog timer with internal oscillator, and five software selectable power saving modes. The device operates between 1.8-5.5 volts. The device achieves throughputs approaching 1 MIPS per MHz.

**LCD**

LCD Stands for Liquid Crystalline Display. To run it via PIC16F73, we need command signals and supply voltage. The signal that is required to display character is produced by an IC which is already embedded on it which is HD44780. A liquid crystal display (LCD) is a flat panel display, electronic visual display, video display that uses the light modulating properties of liquid crystals (LCs). LCs does not emit light directly.
LCD module LM015 displays one line of 16 characters.
- LM015 is initialized with some command words through its control register
- The data to be displayed is written into its data register in ASCII format
- RS pin distinguishes the control and data registers when E is logic high

**Pin Description**

**VCC, VSS and VEE**

The voltage VCC and VSS provided by +5V and ground respectively while VEE is used for controlling LCD contrast.

**RS (Register Select)**

The RS pin is used for their selection as follows. If RS=0, the instruction command code register is selected, then allowing to user to send a command such as clear display, cursor at home etc.. If RS=1, the data register is selected, allowing the user to send data to be displayed on the LCD.

**EN (Enable)**

The enable pin is used by the LCD to latch information presented to its data pins. When data is supplied to data pins, a high power, a high-to-low pulse must be applied to this pin in order to for the LCD to latch in the data presented at the data pins.

**R/W (read/write)**

The R/W (read/write) input allowing the user to write information from it. R/W=1, when it read and R/W=0, when it writing.

**D0-D7 (data lines)**

The 8-bit data pins, D0-D7, are used to send information to the LCD or read the contents of the LCD’s internal registers. To displays the letters and numbers, we send ASCII codes for the letters A-Z, a-z, and numbers 0-9 to these pins while making RS =1. There are also command codes that can be sent to clear the display or force the cursor to the home position or blink the cursor. We also use RS =0 to check the busy flag bit to see if the LCD is ready to receive the information. The busy flag is D7 and can be read when R/W =1 and RS =0, as follows: if R/W =1 and RS =0, when D7 =1(busy flag =1), the LCD is busy taking care of internal operations and will not accept any information. When D7 =0, the LCD is ready to receive new information.

4. **SOFTWARE USED**

- EAGLE(For PCB Designing)
- Arduino (AVR)
- MIT App Inventor 2(for designing Android application)
4.1 EAGLE (For PCB Designing)

EAGLE stands for Easily Applicable Graphical Layout Editor. It is software used to design an electronic schematic and layout of a printed circuit board. It consists of a schematics editor, a PCB editor and an auto router module.

Schematic provides the functional flow and the graphical representation of an electronic circuit. Schematic mainly consists of Electrical connections (nets), Junctions, Integrated circuits symbols, discrete components Eg: Resistors, I/O connectors, Power &ground symbols.

PCB layout: To start laying out the printed circuit board, open the schematics in Eagles schematic editor and click on the board button. Once you have created a board for a schematic, you should always have both files open when working with either the schematic or the circuit board layout.

The Eagle PCB Design Suite is wholly unique in offering the ability to co simulate both high and low-level micro-controller code in the context of a mixed-mode SPICE circuit simulation.

The Eagle PCB Design Suite is wholly unique in offering the ability to co simulate both high and low-level micro-controller code in the context of a mixed-mode SPICE circuit simulation.

![Eagle Board](image)

Figure 4.1: Eagle Board

With this Virtual System Modeling facility, you can transform your product design cycle, reaping huge rewards in terms of reduced time to market and lower costs of development. If one person designs both the hardware and the software then that person benefits as the hardware design may be changed just as easily as the software design. In larger organizations where the two roles are separated, the software designers can begin work as soon as the schematic is completed; there is no need for them to wait until a physical prototype exists. In short, Express PCB improves efficiency, quality and flexibility throughout the design process.

Steps for PCB Designing

The following checklist uses the major area of concern in the process of PCB design: a) Optimum size and shape of board should be ensured. The substrate should be selected properly by taking into consideration its cost, mechanical properties and electrical properties) Layout of conductor pattern should be taken care from cross talk, leakage, shielding, numbers of jumpers required and their placement.) Selection of conductor width, thickness and spacing should be done after analyzing their placement.) Proper productive coating should be selected. Thermal consideration
should be analyzed properly) Proper mounting of heavy and unstable component should be ensured) Easy maintainability should be built in.

4.2 Arduino (AVR)

Arduino is an open-source electronics prototyping platform based on flexible, easy-to-use hardware and software. It's intended for artists, designers, hobbyists, and anyone interested in creating interactive objects or environments. Arduino can sense the environment by receiving input from a variety of sensors and can affect its surroundings by controlling lights, motors, and other actuators. The microcontroller on the board is programmed using the Arduino programming language (based on Wiring) and the Arduino development environment (based on Processing). Arduino projects can be stand-alone or they can communicate with software running on a computer (e.g. Flash, Processing, and Max-MSP).

![Arduino Uno Board](image)

Figure 4.2: Arduino Uno Board [8]

Power

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically. External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm Centre-positive plug into the board's power jack.

Leads from a battery can be inserted in the GND and Vin pin headers of the POWER connector. The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts. The power pins are as follows:

Vin

It is the input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.

5V

The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.

3V3

A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
Memory
The Atmega328 has 32 KB of flash memory for storing code (of which 0.5 KB is used for the boot loader); it has also 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the EEPROM library).

Input and Output
Each of the 14 digital pins on the Uno can be used as an input or output, using pin Mode(), digital Write(), and digital Read() functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 k ohms. In addition, some pins have specialized functions:

Serial
0 (RX) and 1 (TX) are used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.

External Interrupts: 2 and 3
These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attach Interrupt() function for details.

PWM: 3, 5, 6, 9, 10, and 11
Provide 8-bit PWM output with the analog Write() function.

SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK)
These pins support SPI communication, which although provided by the underlying hardware, is not currently included in the Arduino language.

LED: 13
There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Uno has 6 analog inputs, each of which provides 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though it is possible to change the upper end of their range using the AREF pin and the analog Reference() function. Additionally, some pins have specialized functionality:

I2C: 4 (SDA) and 5 (SCL)
Support I2C (TWI) communication using the Wire library. There are a couple of other pins on the board:

AREF:
This is the reference voltage for the analog inputs. Used with analog Reference.

Reset:
Bring this line LOW to reset the microcontroller. It is typically used to add a reset button to shields which block the one on the board.

Communication:
The Arduino Uno has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega8U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The 8U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, on Windows, an *.inf file is required. The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1).
**PIN MAPPING OF ARDUINO WITH ATMEGA 328**

![Arduino Pin Mapping](image)

**Automatic Software (Reset)**

Rather than requiring a physical press of the reset button before an upload, the Arduino Uno is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2 is connected to the reset line of the ATmega328 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the environment. This means that the boot loader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload. This setup has other implications. When the Uno is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the boot loader is running on the Uno. The Uno contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line; see this forum thread for details.

**USB Over-current Protection**

The Arduino Uno has a resettable poly fuse that protects your computer's USB ports from shorts and over current. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

**Physical Specification**

The maximum length and width of the Uno PCB are 2.7 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Three screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100 mil spacing of the other pins.

### 4.3 MIT Application Inventor2 (for designing Android application)

**App Inventor 2 for Android** is an open-source web application originally provided by Google, and now maintained by the Massachusetts Institute of Technology (MIT). It allows newcomers to computer programming to create software applications for the Android operating system (OS). It uses a **graphical interface**, which allows users to drag-and-drop visual objects to create an application that can run on Android devices. App Inventor includes:

**App Inventor Designer:**

A designer is what in which a program's components are specified. This includes visible components, such as buttons and images, which are placed on simulated screen, and non-visible components, such as sensors and web connections. Design the App's User Interface by arranging both on and off-screen components.
**Figure 4.4: App Inventor Designer [14]**

**App Inventor Blocks Editor:**
A blocks editor is what in which the program's logic is created. Program the app's behaviour by putting blocks together.

**Figure 4.5: App Inventor Blocks Editor [11]**

**Connecting Android Phone through AI Companion**
A compiler is based on the JAVA language framework. An app used for real-time debugging on a connected Android device.

**Figure 4.6: Connecting Android Phone through AI Companion [11]**
5. FLOW CHART AND ALGORITHM

5.1 ALGORITHM USED

Algorithm for Automatic Intruder Detection Call

```
#include Header file
Define the pins
SoftwareSerialmySerial(0,1);
LiquidCrystallcd(10, 11, 6, 5, 8, 7);
int IRPin1 = 12;
int buzzer = 9;
int IRPin1State=0
Set up the pin mode (input or output)
Define the Baud rate for serial communication
Wait for port to connect and receive data
Void loop
{
  Check the status of IR pin
  If low
    {
      Print “System Secured”
      Buzzer low
    }
  
  If high
    {
      Print “Intruder Detected”
      Buzzer high
    }
Through Bluetooth send “Intruder Detected” to the android phone
}
```
6. RESULT AND DISCUSSION

Proteus Simulation

The software we are using is Proteus. Our project needs to be connected with IR sensors and buzzer but these components are not available in Proteus so in place of IR sensors we have used a switch.

![Figure 6.1: Schematic view of Project](image1)

In the above design Arduino in interfaced with LCD, switch and Bluetooth.

![Figure 6.2: System Secured](image2)

In the above design, when switch is not pressed i.e. when IR sensors detect no intrusion then LCD display “system secured” and through Bluetooth we send noting for making an efficient use of it.

![Figure 6.3: Intrusion detected](image3)
In the previous design, when switch is pressed i.e. when IR sensors detect the intrusion then LCD display “intrusion detect” and through Bluetooth [HC-05] we send “id” to the Bluetooth of android phone and the application in android phone makes the call to the two numbers that are feed by the users.

7. CONCLUSIONS

Efficient and reliable intrusion detection system is always a big challenge for system designers. In this work many of the problems faced by existing Intrusion Detection Systems (IDS) had been sort out. As most of the IDSs require expert personal to operate the system but with the proposed system with the facilitation of android app one can easily operate this system.

8. REFERENCES