



Implementation of Wireless Sensor Technologies using virtual wire library for controlling alternative current appliances

Mujeeb-ur-Rehman Jamali¹, Shazia Samoon², Nadeem A. Kanasro³, Farhan Bashir Shaikh⁴,
Fiaz Ahmed Memon⁵, Ghulam Nabi Rajper⁶

Institute of Mathematics and Computer Science, University of Sindh, Jamshoro, Pakistan
mujeebjamali@usindh.edu.pk¹

Sindh University Campus @ Larkana, University of Sindh, Jamshoro, Pakistan

Shazia.shamoon@usindh.edu.pk²

SUCL & IMCS, University of Sindh, Jamshoro, Pakistan

nadeem.kanasro@usindh.edu.pk³

Faculty of Information & Communication Technology University Tunku Abdul Rehman Kampar Malaysia

farhan.shaikh@usindh.edu.pk⁴

Institute of Mathematics and Computer Science, University of Sindh, Jamshoro, Pakistan

fiazmemon@usindh.edu.pk⁵

Department of Computer Science and Information Technology Shah Abdul Latif University Khairpur Mir's,
Sindh, Pakistan (rajperghulamnabi@gmail.com)⁶

ABSTRACT

Control higher power devices like lights is difficult and possibly dangerous when power line voltages are being controlled. There are significant differences in controlling AC power compared to DC. The purpose of this paper is to control alternative current through inexpensive wireless technologies sensors with Arduino. The Infrared signals range is short and limited therefore we can control in limited range. The Radio Frequency signals range is wide and strength is very high. Therefore, we can control in wide range with better signal strength.

Key words: Wireless Sensor, Virtual Wire, Arduino, Sketch.

1. INTRODUCTION

Pervasive computing is ubiquitous without interaction Internet of Things (IoT) technologies are used 24/7. Both pervasive computing and wireless sensor networks (WSNs) have important applications such as monitoring of remote environmental and target tracking. Monitoring applications include indoor/outdoor environmental monitoring. While, Tracking applications include tracking objects [1].

WSNs in the recent year gain attention of worldwide. But there are various challenges and constraints depend on environment. The deployment and type of it's are under water and ground, land, multi-media, mobile and terrestrial e.g., multi-media type included video and audio in which event is monitored [2]. The standard protocol layer stack of WSNs is shown in Fig.1.

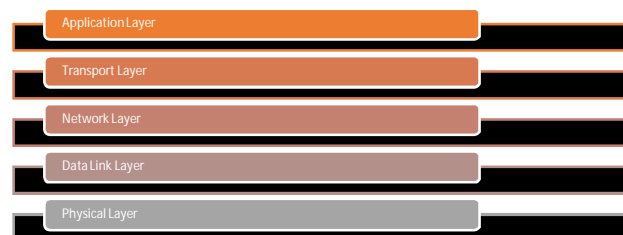


Figure 1: Layer of WSNs

The sensors of WSNs equipped with interface which is used for communication to other sensor / base station which layout the network. In this study, we have developed a project in which controlling electric devices remotely using the sensors of Infrared and Radio Frequency (RF). IR is invisible radiant energy which is 700 nm to 1 mm with 300 GHz to 430 THz. While, RF is electromagnetic wave frequencies which is 1 mm to 100,000 KM lie in the range of 3 Hz to 300 GHz. To send and receive RF signals required an antenna. The strength of signal the radio connection depends on hardware quality that utilized i.e., transmitter/receiver, antenna height etc. Table 1 shows descriptions of complete range of the radio propagation are reproduced:

Table 1: Radio Frequencies Range

RF Description
1. Extremely low frequency (ELF) range of 3–30 Hz
2. Super low frequency (SLF) range of 30–300 Hz
3. Ultra low frequency (ULF) range of 300–3000 Hz

4. Very low frequency (VLF) range of 3–30 KHz
5. Low frequency (LF) range of 30–300 KHz
6. Medium frequency (MF) range of 300 KHz – 3 MHz
7. High frequency (HF) range of 3–30 MHz
8. Very high frequency (VHF) range of 30–300 MHz
9. Ultra high frequency (UHF) range of 300 MHz –3 GHz
10. Super high frequency (SHF) range of 3–30 GHz
11. Extremely high frequency (EHF) range of 30–300 GHz
12. Tremendously high frequency (THF) range of 300 GHz – 3 THz

SENSORS TECHNOLOGIES

The various types of application can be developed using advanced state-of-the-art sensor technology i.e., very large scale integration (VLSI) and microelectro mechanical systems (MEMS). Due to VLSI technology size of sensor is small, which is cheaper and intelligent. The Physical quantities are converted into signal by sensor which can be observe by electronic instrument. Sensors have limited memory and store capabilities due to that radio frequency is used to transfer data to base station i.e., access point, server, laptop or hand held gadget. The sources of power for sensor node is battery, the secondary source of harvest power from alternative power i.e., solar panel may be used. The sensor may be categories as multi-purpose which is known generic for task of monitoring or measure varies attributed in the environment where it is working / installed. The gateway sensor perform function as bridge to transfer the obtained data to the base station. They are higher processing capabilities and both may deploy to form WSNs.

Wireless system use IEEE 802.15.4 standard communication protocol included XBee, ISA100.11, ZigBee, WirelessHART and IETF 6LoWPAN. While Wibree is IEEE 802.15.3 protocol.

The most well-known and widely used wireless standard is ZigBee. The security model of ZigBee included message integrity, confidentiality and access control and protection of reply [4]. In the standard wireless standard, MAC layer may handle the application with specific requirement with own choice of security. Table 2 describes the protection with different level using the Advanced Encryption Standard (AES) sizes of secret key i.e., 32, 64 and 128-bits [5] with MAC. The block encryption and decryption is performed with CTR and CBC mode. The combination of CTR and CBC mode is performed in CCM mode of ZigBee[3].

Table 2: Security Suit of WSNs

Name	Description
Null	No security
AES – CTR	Encryption only, CTR mode
AES – CBC – MAC - 128	128-bit MAC
AES – CBC – MAC - 64	64-bit MAC

AES – CBC – MAC -32	32-bit MAC
AES – CCM - 128	Encryption and 128-bit MAC
AES – CCM - 64	Encryption and 64-bit MAC
AES – CCM - 32	Encryption and 32-bit MAC

ARDUINO TECHNOLOGY

In the year 2005, the first Arduino was released. Arduino Genuino is a hardware and software company that designs open-source hardware & software. Arduino is a compact and low-cost/inexpensive microcontroller in which have a CPU, memory, and input/output functionality on a single chip. However, microcontroller-based kits for creating digital gadgets and interactive things that can sense and control real devices. It is used to operate the device that is integrated within it. Arduino has its own programming language. The Programming Environment for Arduino (IDE) is distinctive and simple. The Arduino Software (IDE) includes a text editor for writing code. The Serial Monitor displays real-time sensor data.

Sketch – a program writes to run on an Arduino Board, Pin – an input or output connect to something e.g. output to an LED. Digital – value is either High or Low, on or off, one or zero that is switch state. Analog – value ranges, usually from 0-255. For programming the microcontrollers, the Arduino project provides an integrated development environment (IDE) based on a programming language named processing, which also supports the languages C and C++. Arduino programme on the computer, then download it to the Arduino and detach the USB cord.

Programming with Arduino is simple and intuitive and syntax is similar to C language. It is a free, open-source physical computing platform that allow to develop project which is a lot easier. With Arduino, the possibilities are endless.

It uses a 9V/12V battery or an external Power Supply. There is other Arduino microcontroller, each types of is different with operating voltage, digital and analog pins etc. Arduino UNO the most popular of the Arduino boards i.e., Microcontroller ATmega1328. Arduino Mega 1280 / 2560, Arduino YUN, Arduino Mini / Mini Lite, Arduino Pro Mini 3.3v / Mini Pro Mini 5v, Arduino Fio, Arduino LilyPad / Simple LilyPad, Arduino Leonardo, Arduino Pro. The snapshot of Arduino UNO is shown in Fig.1.



Figure 1: Arduino UNO Board

The complete description of Arduino Microcontroller is shown in (Table 2):

Table 2:Arduino Microcontroller Description

Microcontroller	ATmega328P
Operating Voltage	5V
Input Voltage	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
PMW Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pin	20 mA
DC Current for 3.3v Pin	50 mA
Flash Memory	32 KB (ATmega328P)
	of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
Length	68.6 mm
Width	53.4 mm
Weight	25 g

VIRTUAL WIRE

This section presents the Arduino Libraries that can be used in the Arduino environment for extension. In programming platform, in the sketches use libraries for additional functionality for working with hardware or manipulating data. In The IDE comes with a variety of libraries to extend the Arduino environment, the same may also download or develop own. Libraries add further functionality for use in sketches, such as the ability to interface with hardware. A library for Arduino is similar to a class with methods and functions which is great to organize code.

WSNs system need data to be transferred from sensor nodes to a centralized base station therefore Ethernet Library i.e., Ethernet Shield and Wi-Fi Shield. SD Libraries included due to sensor nodes have limited storage, thus, SD Library can be used for reading and writing data in SD cards [7].

Virtual Wire is an Arduino communications library that allows several Arduinos to connect with each other using low-cost RF. It is similar to UDP protocol using communication library for a short message to be forwarded over a wireless network without addressing, retransmission, or acknowledgment.

In this project, we have used libraries of IRremote.h, LiquidCrystal.h and VirtualWire.h and their functions/methods for performing the various operations i.e., showing status in LCD, IR and RF transmission and receiving. The implementation of Arduino sketch is shown in Table 3.

Table 3:Arduino Sketch of IR & RF module

1.	#include <IRremote.h>
2.	#include <LiquidCrystal.h>
3.	#include <VirtualWire.h>
4.	int RECV_PIN = 8;
5.	// LCD initialize the library with the numbers of the interface pins
6.	LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
7.	IRrecv irrecv(RECV_PIN);
8.	decode_results results;
9.	int status1 = 0;

10.	int Relay1=A2;
11.	void setup()
12.	{
13.	lcd.begin(0, 3);
14.	Serial.begin(9600);
15.	pinMode(A0, OUTPUT);
16.	pinMode(A1, OUTPUT);
17.	pinMode(Relay1, OUTPUT);
18.	// IRStart the receiver
19.	irrecv.enableIRIn();
20.	// RF
21.	vw_set_rx_pin(10);
22.	vw_set_ptt_pin(9);
23.	vw_setup(2000); // Bits per sec
24.	vw_rx_start();
25.	lcd.clear();
26.	lcd.setCursor(0, 2);
27.	lcd.print(" IR MODE ");
28.	lcd.setCursor(0, 1);
29.	lcd.print("WIRELESS PROJECT");
30.	}
31.	void loop() {
32.	if (irrecv.decode(&results) != 0) {
33.	switch(results.value)
34.	{
35.	case 0x10:
36.	Serial.println("1");
37.	lcd.clear();
38.	lcd.setCursor(0, 2);
39.	lcd.print("IR MODE ");
40.	lcd.setCursor(0, 1);
41.	analogWrite(Relay1, 255);
42.	analogWrite(A0, 0);
43.	delay(2000);
44.	break;
45.	//code start changing mode for RF
46.	case 0xA50:
47.	lcd.clear();
48.	lcd.setCursor(0, 2);
49.	lcd.print("RF MODE ");
50.	lcd.setCursor(0, 1);
51.	lcd.print("RELAY-1..... ");
52.	while(true){
53.	uint8_tbuf[VW_MAX_MESSAGE_LEN];
54.	uint8_tbuflen = VW_MAX_MESSAGE_LEN;
55.	vw_wait_rx();
56.	Serial.println("Waiting for RF Response");
57.	if (vw_get_message(buf, &buflen) // Non-blocking
58.	{
59.	if(buf[0]==0){
60.	lcd.clear();
61.	lcd.setCursor(0, 2);
62.	lcd.print("RF MODE ");
63.	lcd.setCursor(0, 1);
64.	lcd.print("RELAY-1 OFF ");
65.	analogWrite(A0, 0);
66.	analogWrite(Relay1, 255);
67.	Serial.println("0 Received");
68.	}
69.	if(buf[0]==1){
70.	lcd.clear();
71.	lcd.setCursor(0, 2);
72.	lcd.print("RF MODE ");
73.	lcd.setCursor(0, 1);
74.	lcd.print("RELAY-1 ON ");
75.	analogWrite(Relay1, 0);
76.	analogWrite(A0, 255);
77.	Serial.println("1 Received");
78.	}
79.	if(buf[0]==2){
80.	lcd.clear();

```

81. lcd.setCursor(0, 2);
82. lcd.print("IR MODE ");
83. lcd.setCursor(0, 1);
84. lcd.print("RELAY-1 OFF");
85. analogWrite(A0, 0);
86. analogWrite(Relay1, 255);
87. Serial.println("2 Received");
88. gotoendloop;
89. }
90. } // end of out if body
91. } // end of while loop
92. lcd.print("RELAY-1..... ");
93. // irrecv.enableIRIn(); // Start the receiver
94. if (irrecv.decode(&results) != 0){
95. if (results.value == 0xA50){
96. irrecv.resume(); // Receive the next value
97. }
98. } // end of while body
99. break;
100. case 0xA90:
101. if(status1 == 0){
102. analogWrite(A0, 255);
103. analogWrite(Relay1, 0);
104. Serial.println("POWER ON");
105. lcd.clear();
106. lcd.setCursor(0, 2);
107. lcd.print("IR MODE ");
108. lcd.setCursor(0, 1);
109. lcd.print("R-1,2&LED ON ");
110. delay(2000);
111. status1=1;
112. }else{
113. analogWrite(A0, 0);
114. analogWrite(Relay1, 255);
115. Serial.println("POWER OFF");
116. status1=0;
117. lcd.clear();
118. lcd.setCursor(0, 2);
119. lcd.print("IR MODE ");
120. lcd.setCursor(0, 1);
121. lcd.print("R-1,2&LED OFF ");
122. delay(2000);
123. }
124. break;
125. default:
126. lcd.clear();
127. lcd.setCursor(0, 2);
128. lcd.print("IR MODE ");
129. lcd.setCursor(0, 1);
130. lcd.print("UNKNOWN BUTTON ");
131. analogWrite(A1, 255);
132. analogWrite(A0, 255);
133. delay(500);
134. analogWrite(A1, 0);
135. analogWrite(A0, 0);
136. }
137. endloop:
138. irrecv.resume(); // Receive the next value
139. }
140. } // end of loop function
141. // RF Transmitter
142. #include <VirtualWire.h>
143. int P6=6;
144. int P7=7;
145. int P8=8;
146. char *controller;
147. void setup() {

```

```

148. Serial.begin(9600);
149. pinMode(13,OUTPUT);
150. pinMode(P7,INPUT);
151. pinMode(P8,INPUT);
152. pinMode(P6,INPUT);
153. vw_set_tx_pin(10);
154. vw_set_ptt_pin(9);
155. vw_set_ptt_inverted(true); //
156. vw_setup(2000); // speed of data transfer Kbps
157. }
158. void loop(){
159. if(digitalRead(P7) == LOW)
160. {
161. controller="1" ;
162. vw_send((uint8_t *)controller,strlen(controller));
163. message is gone
164. Serial.println("1 Sent");
165. digitalWrite(13, HIGH);
166. delay(500);
167. }
168. if(digitalRead(P8)==LOW){
169. controller="0";
170. vw_send((uint8_t *)controller, strlen(controller));
171. message is gone
172. Serial.println("0 Sent");
173. digitalWrite(13, LOW);
174. delay(500);
175. } // End of if body
176. // For Mode Change
177. if(digitalRead(P6) == LOW){
178. controller="2";
179. vw_send((uint8_t *)controller, strlen(controller));
180. message is gone
181. Serial.println("2 Sent");
182. } // End of if body
183. } // end of loop function body

```

2. METHODOLOGY

The special properties of electric current oscillate at RF by direct and alternative current not shared. AC light required 220 v power which cannot directly controlled Arduino Board. The Relay provide the facility to connect with Board to control high voltage n amp which require or electronic equipment. IR Receiver is connected with Arduino. Arduino receives encoded signal coming from IR Remote and then decodes by using IR Remote.h library. After decoding the signal, compare decoded signal with defined value, if the match occurs then perform related function or switching on/off related relay by using digital input/output. Here AC load is used for demonstration. Alternative current controlling appliances are used for demonstration. In which one zero-watt bulb indicated by light 1 and second is 100-watt bulb indicated by light 2 and third and last is LCD indicated. LCD is also added in this system for displaying the current status of Technology in use. Fritzing open source tool for making simple wiring diagram for your hardware wiring. Fig.3 presents IR and RF Design Model pins in Development Board with connection to Arduino ports and pins [6].

Button 1-3 Control		RF 433 MHz Transmitter		RF 433MHz Receiver		IR Receiver		Buzzer		LCD Pin	
Pin	Arduino	Pin	Arduino Pin	RF Pin	Arduino Pin	IR Pin	Arduino Pin	Pin	Arduino Pin	LCD Pin	Arduino Pin
Button 1 OFF Relay-1		1.DA	10	1.VCC	5V+	1.VCC	5V+	1.GND	GND	1. VSS	GND
1.GND	GND	2.VCC	5V+	2.GND	GND	2.GND	GND	2.DA	A1	2. VDD	5V+
2.DA	8	3.GND	GND	3.DA	10	3.DA	7			3. VO	3
Button 2 ON Relay-1										4. R/W	GND
1.GND	GND									5. RS	12
2.DA	7									6. E	11
Button 3 Change RF to IR Mode										7. D4	5
1.GND	GND									8. D5	4
2.DA	6									9. D6	3
										10. D7	2
										11. A	5V+
										12. K	GND

Figure 3: Design Model of Transmitter and Receiver

3. RESULTS AND DISCUSSION

The working prototype of proposed IR and RF designed model using inexpensive Arduino Board and Relay to Control 220 Watts electric appliance (High Voltage current from AC light through Arduino 5 V).

Relay is an electronic component used to control AC voltage to DC voltage. Relay is connected Arduino UNO Board. Switches of relay are NO, NC and COM. Normally Closed Connection and Normally Open Connection (End Connection control through Arduino 5 v).AC light (high voltage) through common connection controlled.

Three Button for controlling the alternative current appliance connected. Button 1 OFF Relay-1 connected to Bread Board pins GND and DA with Arduino Board pins GND and port 8 respectively. Button 2 ON Relay-1 connected to Bread Board pins GND and DA, while, Arduino pins GND and port 7 used respectively. Button 3 change RF and IR Mode connected with Bread Board pins GND and DA, while Arduino ports GND and 6 used respectively.

Showing current status LCD was used. The LCD device I/O pins are used i.e., VSS, VDD, VO, R/W, RS, E, D4, D5, D6, D7, A, K connected to Arduino I/O pins and ports i.e., GND, 5V+, 3, GND, 12, 11, 5, 4, 3, 2, 5V+ and GND respectively.

RF 433MHz receiver is connected using RF pin using VCC, GND and DA port, while Arduino Pin i.e., 5V+, GND and 10 respectively. IR receiver connected with IR Pin of VCC, GND and DA, while, Arduino Pins are used 5V+, GND and 7 port of Arduino.

4. HARDWARE AND SOFTWARE REQUIREMENT

In this paper we have controlled 220 Watts electric current appliances through IR and RF sensors. The hardware and software used in experiment environment that is Arduino UNO. We have used for controlling using the inexpensive sensors, Arduino UNO, IR Transmitter (Remote Control) and Receiver, 2. RF Module (433 Mhz) Transmitter and Receiver, Relay Module, 220 Watts Bulb, Buzzer, LCD, 2x2, and Buttons. Arduino-Using Solderless Breadboard. Development Board (A printed circuit board designed to facilitate, work with a particular microcontroller) components include: microcontroller, power circuit, programming interface, basic input; usually buttons and LEDs, I/O pins.

The prototype of system can be build using the solderless Board which is fast and provide facility without wasting of time connect all connection in initial stage thereafter soldering Board are used to install/fixed all connection for real time working of Syndicate.

The software specification of the used for Arduino-1.6.0-Windows, Virtual Wire External Library, IR Remote External Library and Fritzing.

5. CONCLUSION

In this paper, we have designed an efficient circuit using Arduino UNO board for controlling alternative current using WSN technology inexpensive/low cost sensors i.e., RF and IF and Relay to control alternative current (220 watts electric appliance) using high voltage current from AC light through Arduino 5v. The working prototype performing required task efficiently.

6. ACKNOWLEDGEMENT

We would like to express my profound humble and gracious thanks to those who help and guide make it possible to complete this work. Thanks to all for supporting us throughout this work.

REFERENCES

- [1] WalteneagusDargie and Christian Poellabauer (2010). "Fundamentals of Wireless Sensor Networks Theory and Practice", Wiley Series on Wireless Communications and Mobile Computing, Wiley, A John Wiley and Sons, Ltd., Publication
- [2] Jennifer Yick, Biswanath Mukherjee, DipakGhosal, (2008). "Wireless sensor network survey", 52 (2008) 2292–2330, 2292–2330
- [3] Boyle, D., and Newe, T, (2008). "Securing wireless sensor networks: Security architectures", Journal of Networks 3 (1), 65–77.
- [4] Karlof, C., Sastry, N. and Wagner, D, (2004), "TinySec: A link layer security architecture for wireless sensornetworks" Proc. of the 2nd International Conference on Embedded Networked Sensor Systems, Baltimore, MD.

- [5] Mujeeb-ur-RehmanJamali, Abdul GhafoorMemon, Nadeem A. Kanasro, Mujeeb-u-Rehman Maree, (2021). “Data integrity issues and challenges in next generation non-relational document-oriented database outsourced in public cloud”, International Journal of Emerging Trends in Engineering Research, Volume 9. No. 4, April 2021, pp- 416-420. ISSN 2347-3983. <https://doi.org/10.30534/ijeter/2021/13942021>.
- [6] Nadeem Ahmed Kanasro, Mujeeb-U-RehamJamali, Ghulam Muhammad and Farhan Ali Surahio, (2020). “Controlling of Home Appliances using Arduino UNO Board with Infrared (IR) and Radio Frequency (RF) Technologies”, International Journal of Advanced Trends in Computer Science and Engineering, Volume 9, No.5, September – October, 2020, ISSN 2278-3091, <http://www.warse.org/IJATCSE/static/pdf/file/ijatcse317952020.pdf>. <https://doi.org/10.30534/ijatcse/2020/317952020>,pp- 9113- 9116.
- [7] <https://www.arduino.cc/en/Hacking/LibraryTutorial>