RESEARCH ARTICLE

On New Approach in Using 433MHz Radio Modules

Prithviraj Shetti[1], Prasad V. Sakharpe[2], Amrut Ubare[3]

[1]Lecturer, Department Electronics & Telecom., Ashokrao Mane Polytechnic, Kolhapur-416 112, India

[2]Lecturer, Department Electronics & Telecom., Ashokrao Mane Polytechnic, Kolhapur-416 112, India

[3]Student, Third year, Ashokrao Mane College of Engg, Kolhapur-416 112, India

[1] prsece@gmail.com; [2] sakharpe_pv@yahoo.co.in; [3] amrutubare@gmail.com

Abstract— Cheap radio modules such as 433MHz Rx/Tx pairs are very popular in hobby projects and readily available in the local market. For reliable data transmission and reception these modules need initial burst training pulses for sync and some encoding scheme for reducing effect of noise. Present approaches make use of VirtualWire library and Manchester library for using these modules with Arduino board. This gives transmission of 3-4 digit sensor values and 5-12 characters text string. In present paper we demonstrate it is possible to send even longer text string and integer data without using these two libraries and using another protocol viz. software serial only. Transmission of long text strings is demonstrated with reliable results.

Keywords— Manchester library; VirtualWire library; SoftwareSerial library; Arduino board; 433MHz radio modules

I. INTRODUCTION

Cheaply available 433MHz radio modules are very popular among hobbyists building wireless projects. Sometimes it is sending some integer sensor value on air and receiving it at another location. And sometimes it is sending text strings on air. Some projects may use both features. This is expected to work over range of 10-90 meters. These cheap radio modules require specially formatted data, with sync patterns, equal balance of 0's and 1's and error checking. Presently there are two libraries that satisfy these criteria and make easy using these modules with Arduino board. One is VirtualWire library [1], with which it is possible to transmit very short strings and integer values. Another is Manchester library [2] which is based on Manchester encoding technique. With this it possible to transmit short length integer values.

In present paper we demonstrate the possibility of transmitting long text strings and integer values without use of these encoding scheme based libraries, but using SoftwareSerial library only. It is possible to transmit and receive more than 50 characters reliably over short distances. The SoftwareSerial library is originally intended for having an extra serial port over the Arduino board for interfacing extra serial devices. This is software based serial port which is different than hardware UART already available on board ready for serial communication. This makes possible interfacing more than one serial device with Arduino board at a time. e.g using GSM module for SMS Tx/Rx and Serial Monitor for debugging at same time. Here the serial monitor uses built-in

UART of Arduino board for communication. We have used the inherent protocol structure of this library to reliably receive long text strings.

II. RELATED WORK

The main intention of using 433MHz radio modules in projects is to transmit and receive integer values and text strings over short distances. Present two approaches use VirtualWire and Manchester libraries which are based on encoding schemes.

A. *VirtualWire library*

This library was written by Mike McCauley[1]. It uses many individual functions. We have to define transmit and receive pins, baud rate for communication, follow simple Tx/Rx protocols to reliably receive data. Following is the summary of some inbuilt ready to use functions.

vw-set_tx_pin(transmit pin) — were we set transmit pin on Arduino board.
vw-set_rx_pin(receive pin) — were we set receive pin on Arduino board.
vw-setup(baud rate) — were we set baud rate for communication.
vw-send(message, length) — were we send message of specified length.
vw-wait_tx() — were we wait for transmitter to complete data transmission.
vw-wait_rx() — were we wait for receiver to complete data reception.

The VirtualWire library is been found too sensitive to clock frequency and ambient temperature. Thus there are found dropouts in data.[3] Another problem with this approach is, it facilitates transmission of only very short text strings and integer data which limits scope of use.

B. *Manchester library*

This library is originally written by carl47-as named on Arduino forum.[3] And further developed by mchr3k-as named on Arduino forum and others.[2] It is based on Manchester encoding technique.[4]
Following is the summary of some inbuilt ready to use functions.

man.setupTransmitter (TX_PIN, MAN_1200) – set Transmitter pin and baud rate.
man.setup Receive (RX_PIN, MAN_1200) – set Receiver pin and baud rate.
man.beginReceive () – to begin Receiving data.
man.beginTransmit (transmit_data) – to begin Transmitting data.
man.receiveComplete () – returns true if data reception complete.

The advantage of using this library is reliable transmission in noisy conditions but message length is too short and that to only numeric data can be transmitted. Not useful for long text string transmission.

III. HARDWARE SETUP

Fig1-2 shows the hardware setup which is invariably same if any of the libraries are used. The sketch is developed in Fritzing.[5] We have Arduino board on both transmitter and receiver sides connected to respective transmit and receive modules of 433MHz. The radio modules require +5 volt power supply which is given from Arduino board itself. Fig3,4 show actual setup of our experiment.
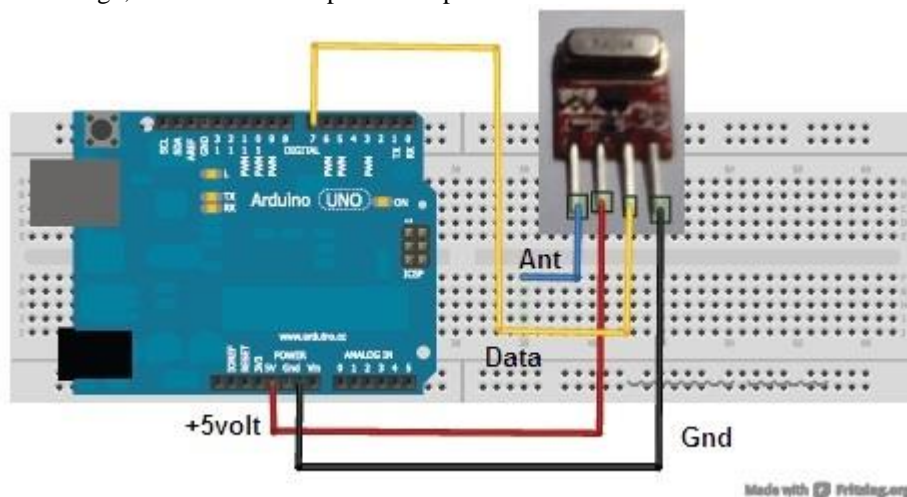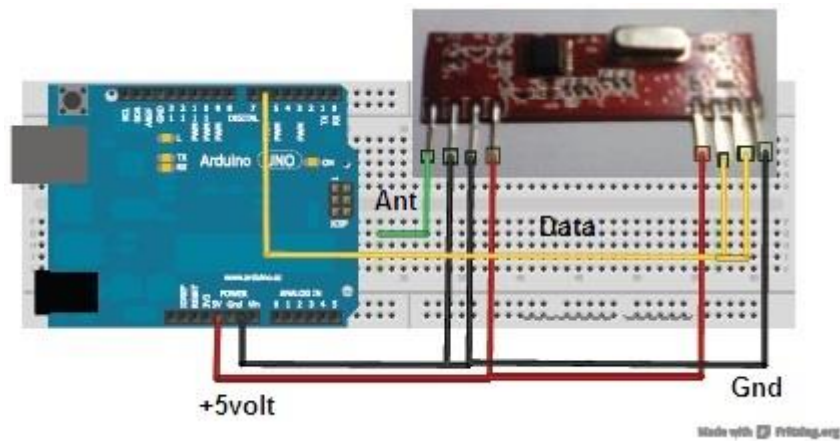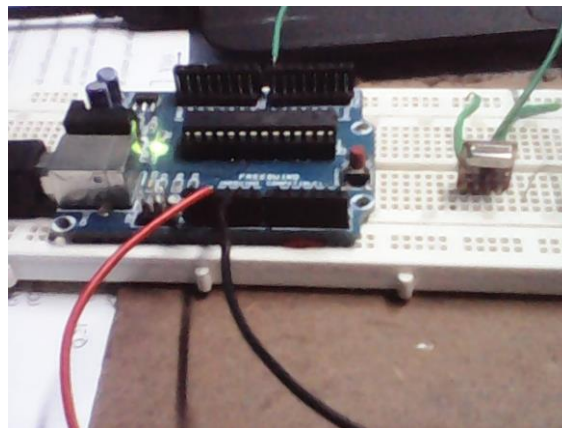

Fig1: Transmitter side

Fig2: Receiver side
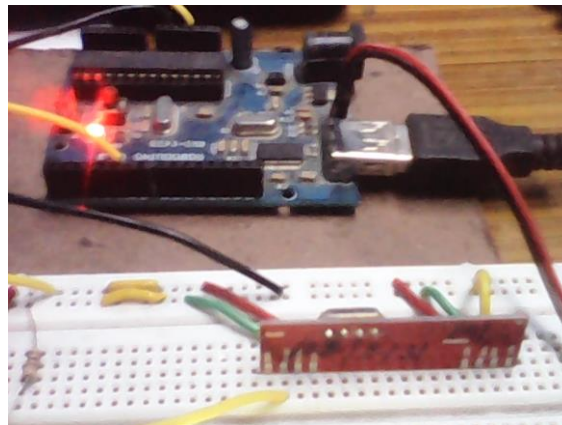

Fig3: Actual Transmitter setup

Fig4: Actual Receiver setup

IV. SOFTWARE DESIGN

A. *Transmitter flowchart*

We have used SoftwareSerial library for writing Transmitter and Receiver code. Fig 5 shows flowchart transmitter code and fig6 shows flowchart for receiver code.
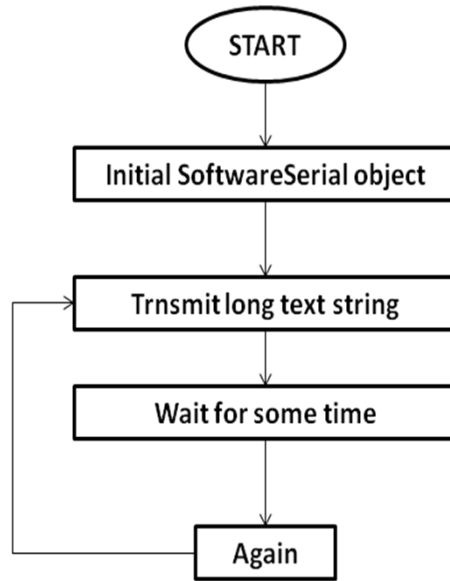
Fig5: flowchart for Transmitter code
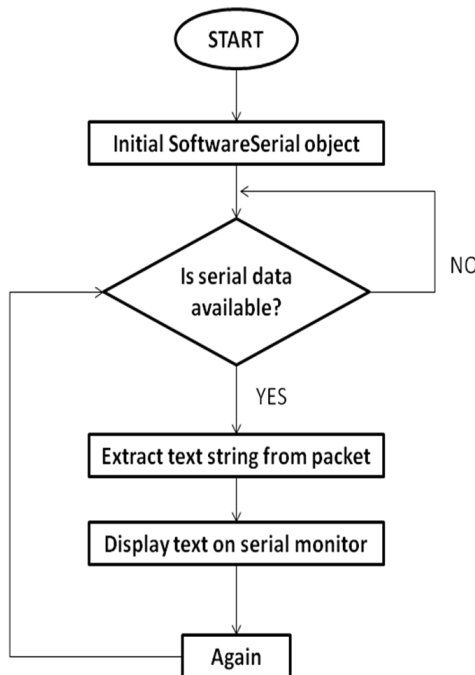
B. *Receiver flowchart*

Fig6: flowchart for Receiver code

V. RESULTS

Fig7,8,9 show the received text message on serial monitor of Arduino IDE. The screen of serial monitor is shown as a series of images due to the inability to fit entire frame in single picture. If combined the text is "Transmitting long text string using 433MHz modules". The length of received character count can also be seen. It is around 50 characters. The setup is tested for distances of 10 to 50 meters with same results. These results are obtained without using any specific error correction or detection for over the air received signal. Also there isn't any kind of acknowledgment scheme used between Transmitter and Receiver, so rate of transmission is also better. Both the transmitter and receiver side codes are most simple in form. Though there are occasional dropouts, improvement can be obtained by using antenna on both sides.
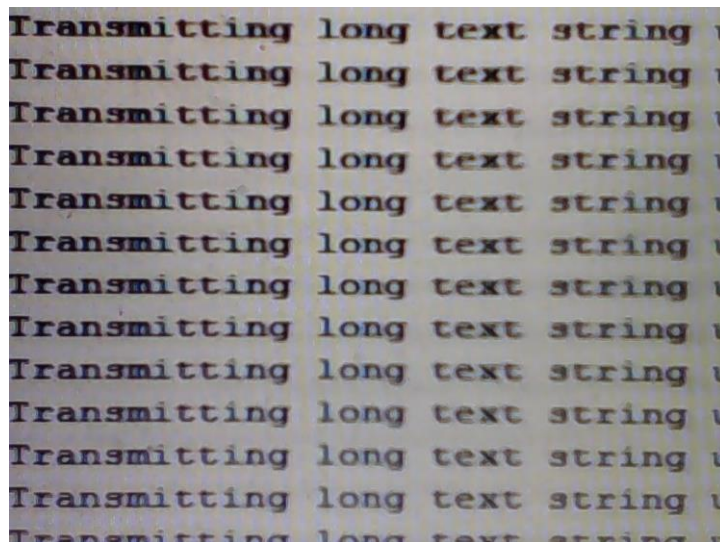
Transmitting long text string u
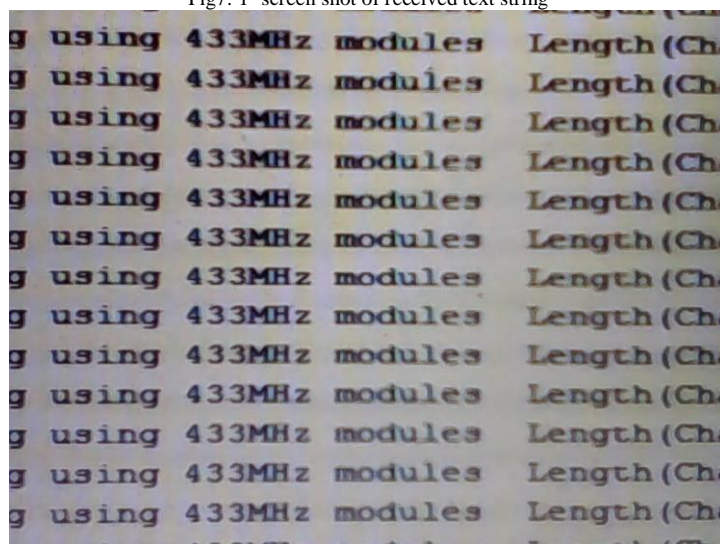Transmitting long text string u
Transmitting long text string u
Transmitting long text string u
Transmitting long text string u
Transmitting long text string u
Transmitting long text string u
Transmitting long text string u
Transmitting long text string u
Transmitting long text string u
Transmitting long text string u
Transmitting long text string u
Transmitting long text string u

Fig7: 1st screen shot of received text string

g using 433MHz modules   Length(Cha
g using 433MHz modules   Length(Cha
g using 433MHz modules   Length(Cha
g using 433MHz modules   Length(Cha
g using 433MHz modules   Length(Cha
g using 433MHz modules   Length(Cha
g using 433MHz modules   Length(Cha
g using 433MHz modules   Length(Cha
g using 433MHz modules   Length(Cha
g using 433MHz modules   Length(Cha
g using 433MHz modules   Length(Cha
g using 433MHz modules   Length(Cha

Fig8: 2nd screen shot of received text string

ltes   Length(Charachters)= 50
les   Length(Charachters)= 50
les   Length(Charachters)= 50
les   Length(Charachters)= 50
les   Length(Charachters)= 50
les   Length(Charachters)= 50
les   Length(Charachters)= 50
les   Length(Charachters)= 50
les   Length(Charachters)= 50
les   Length(Charachters)= 50
les   Length(Charachters)= 50
les   Length(Charachters)= 50
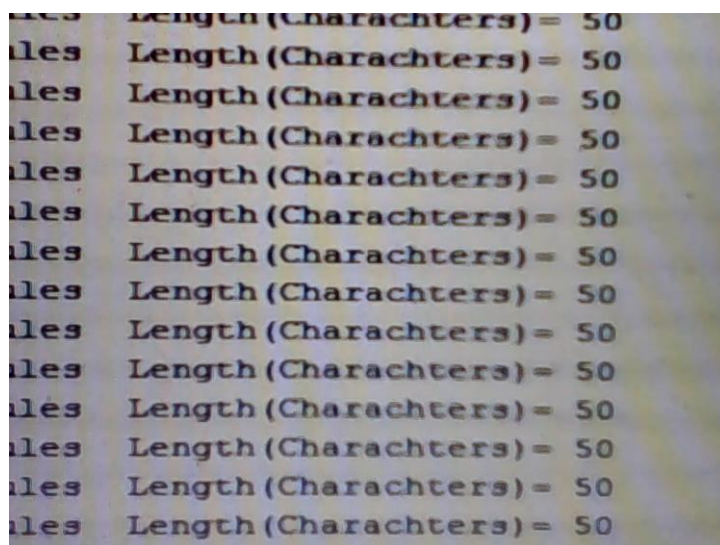les   Length(Charachters)= 50
les   Length(Charachters)= 50

Fig9: 3rd screen shot of received text string

VI. CONCLUSIONS

433MHz radio modules are certainly most cheap solutions for hobby wireless projects and low end industrial applications. Given we have some mechanism to deal with noise interference and tools to increase the scope of them. We observed that long text strings with around 50 characters or more can be transmitted and received successfully. Here we made use of inherent communication protocol of the SoftwareSerial library and did not device any extra Error detection or correction for noise affecting the signal on air. Yet it works for distances upto 50 meters which is sufficient for low end applications.

ACKNOWLEDGMENT

I would like to thank all people around internet for their work on Arduino with 433MHz radio modules. The discussions on forums are especially helpful and are duly acknowledged with various references throughout the paper.

REFERENCES

[1]   Virtual at airspayce.com. [Online]. Available: http://www. airspayce.com/mikem/Arduino/virtualwire.pdf
[2]   Manchester library. [Online].Available: http://mchr3k.github.io/Arduino-libs-manchester/
[3]   Mchr3k at Arduino forum. [Online].Available: http://www.forum.arduino.cc/index.php/topic,63755.0.html
[4]   Manchester. [Online.] Available: http://www.atmel.com/dyne/resources/prod_documents/doc9164.pdf
       Fritzing. [Online.] Available: http://www.fab.fritzing.org