CS COMPLEX ENGINEERING PROBLEM REPORT

TOPIC: "WIRELESS BUZZER SYSTEM"

COURSE TITLE: COMMUNICATION SYSTEM

COURSE CODE: TC-306

SECTION: B

GROUP MEMBERS: -

- EE-21052 (ANAS IRFAN)
- EE-21057 (SIMRAH SALEEM)
- EE-21061 (SYEDA HAMNAH AHMED)
- EE-21062 (IQRA TARIQ)
- EE-21066 (HIBA KHAWAR)
- EE-21070 (MUZNA SAJID HANFI)

WIRELESS BUZZER SYSTEM

INTRODUCTION:

A wireless buzzer system is a convenient and effective means for signal or alerting purposes across distances in various settings such as educational institutions, offices, industries, hospitals, homes, restaurants, etc. They represent a significant advancement in the realm of communication and alert mechanisms. Unlike traditional wired systems, wireless buzzers eliminate the need for physical connections, offering enhanced flexibility, ease of installation, mobility and extended operational range.

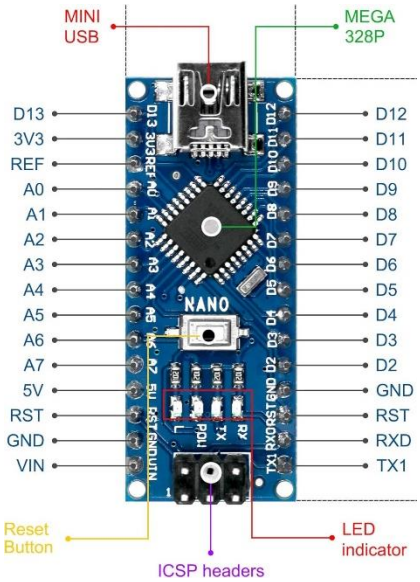
Its primary function is to provide an audible alert or notification. There are various types of these modules that differ in their sound generation mechanisms, operating principles, and utilizations. Some of its common applications are doorbells, kitchen timers, alarm clocks, game show buzzers, classroom participation system, microwave ovens, laundry machines, smoke detectors, parking sensors in vehicles. In this report, we explore the design and implementation of one of its common applications i.e. “Wireless Quiz Buzzer System”.

The core principle behind these systems is the transmission of signals through radio frequencies, allowing for remote activation of buzzers or alarms over considerable distances. One of the most commonly used frequency bands for such applications is 433 MHz, due to its optimal balance between range, data transmission rate, and power consumption. Components like the transmitter and receiver modules operate efficiently within this band, making them ideal for wireless communication projects. Coupled with encoding and decoding integrated circuits (ICs), these modules can securely and reliably transmit data. The system leverages the capabilities of the Arduino Nano microcontroller to handle the programmable logic and control functions, managing inputs and outputs, encoding and decoding data, ensuring precise and responsive communication between the transmitter and receiver.

COMPONENTS:

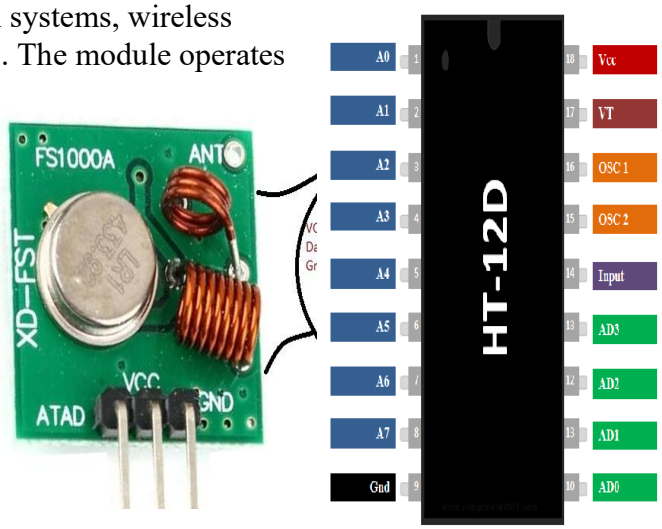
- **ARDUINO NANO:**

The Arduino Nano is a small, compact, and versatile microcontroller board designed for embedded applications. It is based on the ATmega328P microcontroller. This chip has 32 KB of flash memory for storing programs, 2 KB of SRAM, and 1 KB of EEPROM. It consists of 14 digital input/output pins, of which 6 can be used as PWM (Pulse Width Modulation) outputs, 8 analog input pins, each providing 10-bit resolution. It supports UART (serial), SPI (Serial Peripheral Interface), and I2C (Inter-Integrated Circuit) communication protocols. It can be powered via the USB connection or an external power supply. It is programmed using the Arduino Integrated Development Environment (IDE), a user-friendly platform that supports C/C++ programming languages.



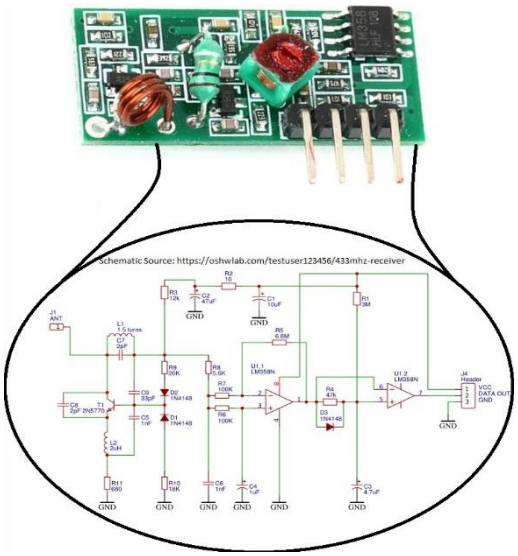
- **FS1000A 433MHz TRANSMITTER MODULE:**

The FS1000A is a 433MHz (radio frequency) transmitter module commonly used for wireless communication in various projects. It is designed to send data wirelessly over relatively short distances, making it suitable for remote control systems, wireless sensor networks, and other similar applications. The module operates at a frequency of 433 MHz, a standard frequency for many wireless communication applications. It uses Amplitude Shift Keying modulation to transmit data. This modulation technique varies the amplitude of the carrier signal to represent binary data. It supports data rates up to 10 kbps. The module usually has a built-in antenna or a pin for connecting an external antenna to enhance its range and signal quality.



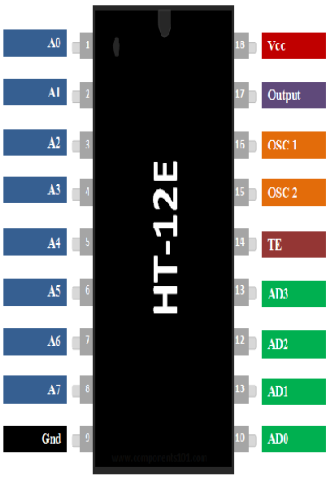
- **MX-RM-5V 433MHz RECEIVER MODULE:**

The MX-RM-5V 433MHz receiver module is a component used to receive radio frequency (RF) signals transmitted at 433 MHz. It is designed to work in conjunction with a 433MHz transmitter module, to enable wireless communication between devices. It uses Amplitude Shift Keying modulation to decode the amplitude variations of the received RF signal into binary data. This modulation technique varies the amplitude of the carrier signal to represent binary data. It supports data rates up to 10 kbps. The module usually has a built-in antenna or a pin for connecting an external antenna to enhance its range and signal quality.



- **HT12E ENCODER IC:**

The HT12E is a 212 series encoder IC (Integrated Circuit) used to encode data for transmission in remote control systems. It can encode 4 bits of data, allowing for up to 16 different signal combinations. It includes 8 address bits, which can be used to set unique addresses for multiple receivers, ensuring that the transmitted data is received by the intended receiver only. The encoded data is output serially, making it suitable for transmission over RF or IR communication links. The HT12E provides a latch mode which ensures that the data is continuously transmitted if the transmission enable pin is active. The status of each address/data pin can be individually pre-set to logic "high" or "low". If a transmission-enable signal is applied, the encoder scans and transmits the status of the 12 bits of address/data serially in the order A0 to AD11 for the HT12E encoder and A0 to D11 for the HT12A encoder.

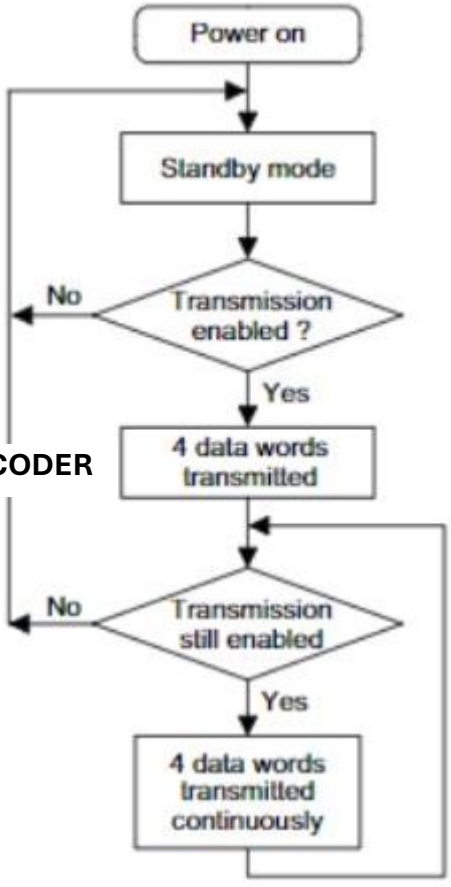


- **HT12D DECODER IC:**

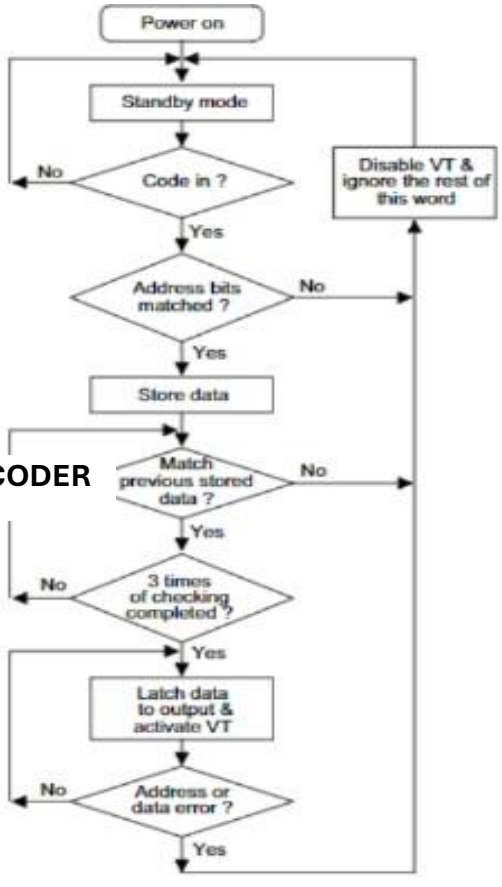
The HT12D is a 212 series decoder IC (Integrated Circuit) designed to decode data received from a remote-control transmitter. The HT12D can decode 4 bits of data, corresponding to the 4 data bits encoded by the HT12E encoder IC. It includes 8 address bits which must match the address set in the transmitter (HT12E) for successful decoding of data. The decoded data is received serially, compatible with the serial data output from the HT12E encoder IC. The HT12D provides a latch mode where the received data remains valid as long as the transmission enable (TE) pin is active. It includes an internal oscillator circuit, requiring only a single external resistor to set the frequency.

Following is the flow chart diagram of encoder and decoder circuit which may better help in understanding their working principle;

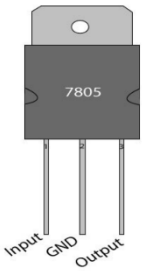
ENCODER



DECODER

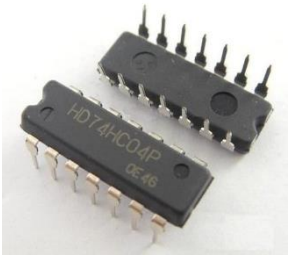


- **VOLTAGE REGULATOR IC:**



The 7805 IC is used to regulate and stabilize the output voltage at +5V DC, regardless of variations in the input voltage and load conditions.

- **NOT GATE IC:**



The not gate 7404 IC in our case is used to make the low-level output to high-level. If the input is 1 (high), the output is 0 (low), and if the input is 0 (low), the output is 1 (high). It's often used to reverse a digital signal.

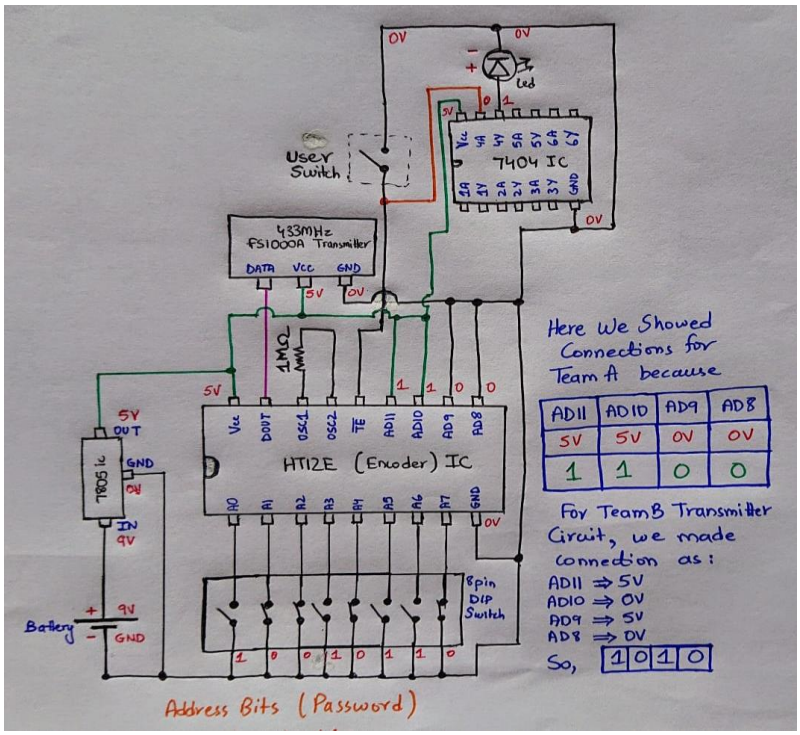
- **BUZZER AND LEDs:**

Buzzer and LEDs are integrated to provide auditory and visual feedback, enhancing the user interface with signals and indicators.

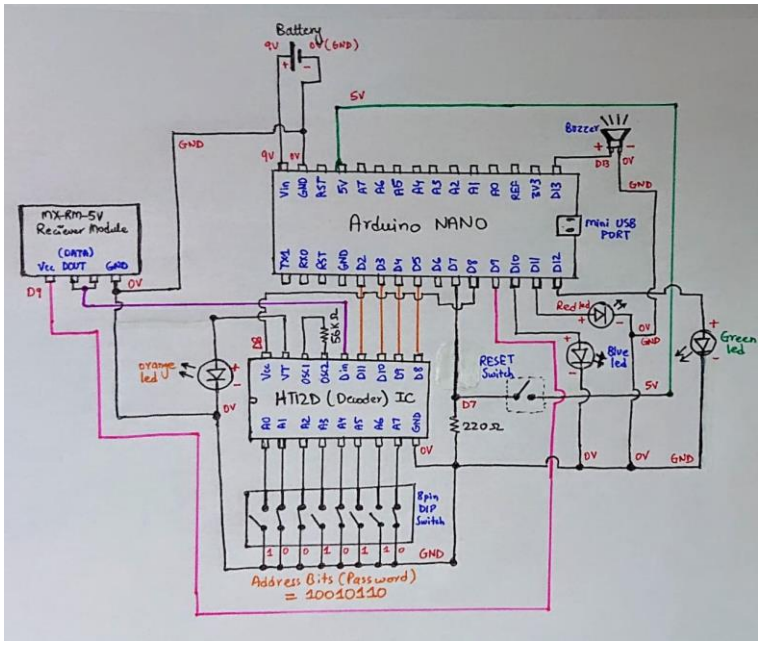
WORKING PRINCIPLE:

The circuit is powered through a 9V battery source which first goes into the voltage regulator IC that coverts and regulate the voltage to 5V which is then connected to the Vcc pins and ground is connected to the GND pin of all the ICs and modules.

Now, A0 to A7 pins of HT12E encoder IC are address bits which are connected to 8-pin dip switch, this switch operates on the condition such that when it is closed, the 8 pins get connected to ground because by default these pins of encoder IC are high-level, its 4 data bits AD8 to AD11 are connected to Vcc and ground in such a way that making address code i.e. 1100, its TE(bar) pin is connected to the user switch and the other leg of user switch is grounded, so when the user presses the switch, the circuit is closed, and 0 appears on TE(bar) but since this pin will invert its input so IC will receive 1, hence it will continuously transmit the address and data bits together (sending a combined 12 bit address – starting 8 bits are address and last 4 bits are data) through Dout pin to Data input pin of 433MHz FS1000A Transmitter module. To indicate that user has pressed the switch and data is being transmitted, we connected a NOT gate's input (IC7404) through a user switch that is going into TE(bar) pin, and its output is connected to led, so when the switch is closed, the IC will receive 0 and its output would be 1, so the led gets lit indicating that data is transmitted.

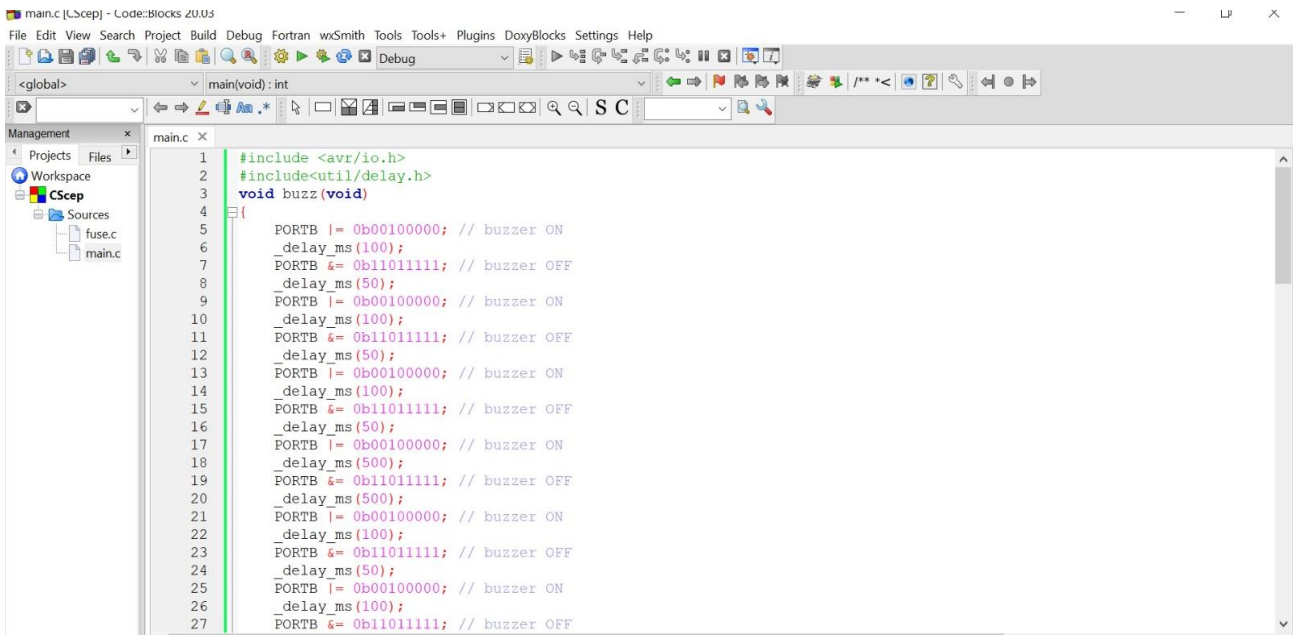


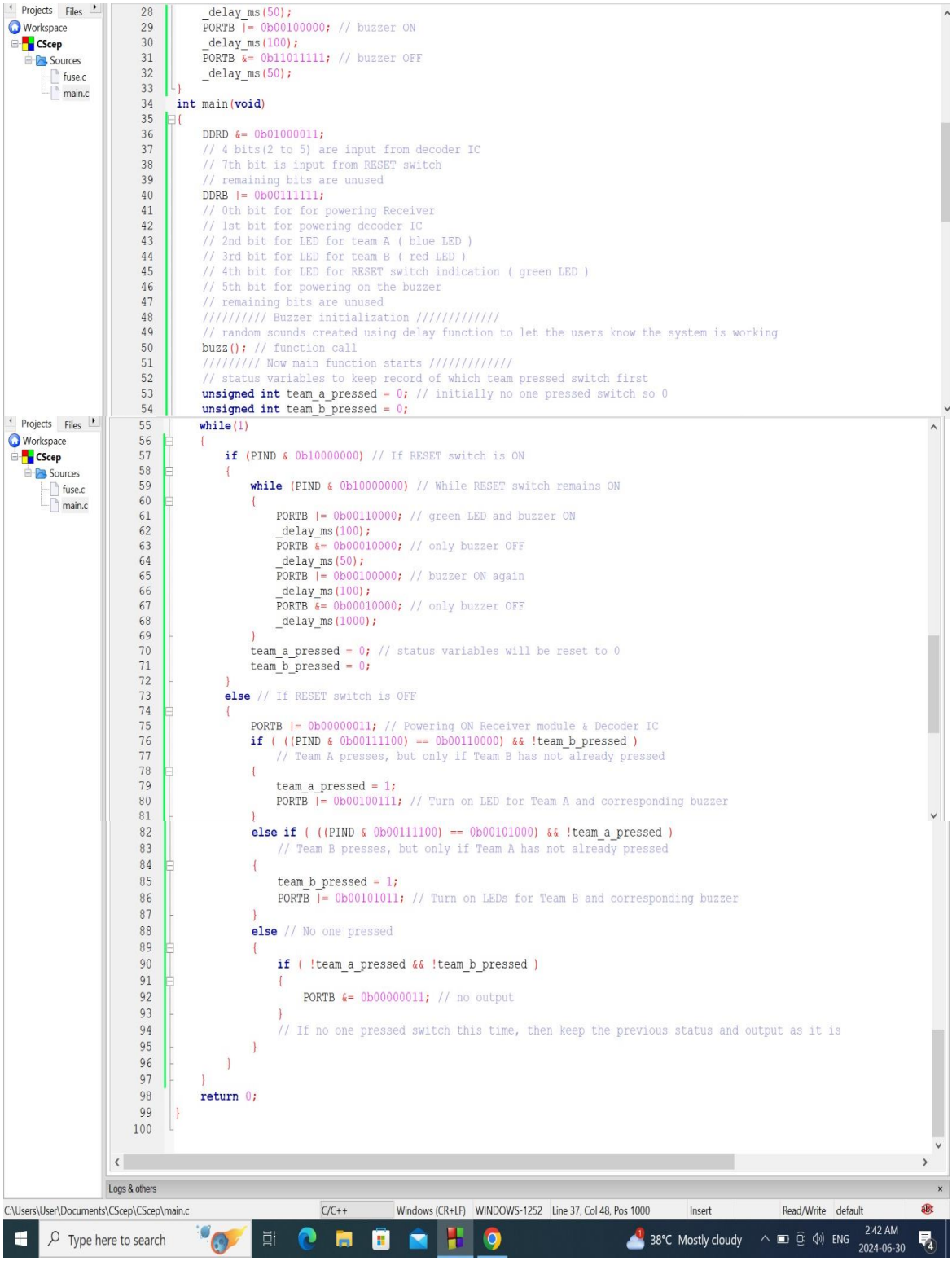
Then, the receiver circuit works similarly as the transmitter circuit. The 433MHz FS1000A receiver module receives the data through antenna and then send it to decoder IC through connection of Dout pin of receiver to the Din pin of decoder. The HT12D decoder IC should have the exact same address and dita bits as encoder IC and similarly its address bits are connected to a dip switch. The data pins D8 to D11 of HT12D are connected to digital pins of Arduino Nano (D2, D3, D4, D5). The Arduino compare these bits with the already given data code bits of teams (contestants), to display that which of the teams (i.e. A or B) have pressed the buzzer. The Arduino is programmed in such a way that it compares the data by using if else statement, then sends the logic high to one of its output pins, D10 and D11 which are connected to different colors of LEDs (blue and red) indicating 2 teams. The buzzer I connected to pin D13 of Nano which will be turned on when the user switch or reset switch is pressed. The pin D9 of Arduino is used for resetting condition by a reset switch, when the circuit is reset, the green led is turned on continuously until the reset switch is turned off and specific type of buzzer sound is heard to indicate that the buzzer system is functional.



Since, we integrated the wireless quiz buzzer system for two contestants therefore, there are two transmitter and receiver circuit with same logic but specific data bits or password for both. So, the code of data bits in transmitter circuit for team A is 1100 and for team B is 1010. When any of the teams presses the buzzer, there would be a buzzer sound and specific color led would be turned on i.e. blue led for team A and red led for team B.

CODE FOR THE APPLICATION:





```
    _delay_ms(50);
    PORTB |= 0b00100000; // buzzer ON
    _delay_ms(100);
    PORTB &= 0b11011111; // buzzer OFF
    _delay_ms(50);
}
int main(void)
{
    DDRD &= 0b01000011;
    // 4 bits(2 to 5) are input from decoder IC
    // 7th bit is input from RESET switch
    // remaining bits are unused
    DDRB |= 0b00111111;
    // 0th bit for for powering Receiver
    // 1st bit for powering decoder IC
    // 2nd bit for LED for team A ( blue LED )
    // 3rd bit for LED for team B ( red LED )
    // 4th bit for LED for RESET switch indication ( green LED )
    // 5th bit for powering on the buzzer
    // remaining bits are unused
    ////////// Buzzer initialization //////////
    // random sounds created using delay function to let the users know the system is working
    buzz(); // function call
    //////// Now main function starts //////////
    // status variables to keep record of which team pressed switch first
    unsigned int team_a_pressed = 0; // initially no one pressed switch so 0
    unsigned int team_b_pressed = 0;
    while(1)
    {
        if (PIND & 0b10000000) // If RESET switch is ON
        {
            while (PIND & 0b10000000) // While RESET switch remains ON
            {
                PORTB |= 0b00110000; // green LED and buzzer ON
                _delay_ms(100);
                PORTB &= 0b00010000; // only buzzer OFF
                _delay_ms(50);
                PORTB |= 0b00100000; // buzzer ON again
                _delay_ms(100);
                PORTB &= 0b00010000; // only buzzer OFF
                _delay_ms(1000);
            }
            team_a_pressed = 0; // status variables will be reset to 0
            team_b_pressed = 0;
        }
        else // If RESET switch is OFF
        {
            PORTB |= 0b00000011; // Powering ON Receiver module & Decoder IC
            if ( ((PIND & 0b00111100) == 0b00110000) && !team_b_pressed )
                // Team A presses, but only if Team B has not already pressed
            {
                team_a_pressed = 1;
                PORTB |= 0b00100111; // Turn on LED for Team A and corresponding buzzer
            }
            else if ( ((PIND & 0b00111100) == 0b00101000) && !team_a_pressed )
                // Team B presses, but only if Team A has not already pressed
            {
                team_b_pressed = 1;
                PORTB |= 0b00101011; // Turn on LEDs for Team B and corresponding buzzer
            }
            else // No one pressed
            {
                if ( !team_a_pressed && !team_b_pressed )
                {
                    PORTB &= 0b00000011; // no output
                }
                // If no one pressed switch this time, then keep the previous status and output as it is
            }
        }
    }
    return 0;
}
```

EXPLANATION OF THE RESULTS:

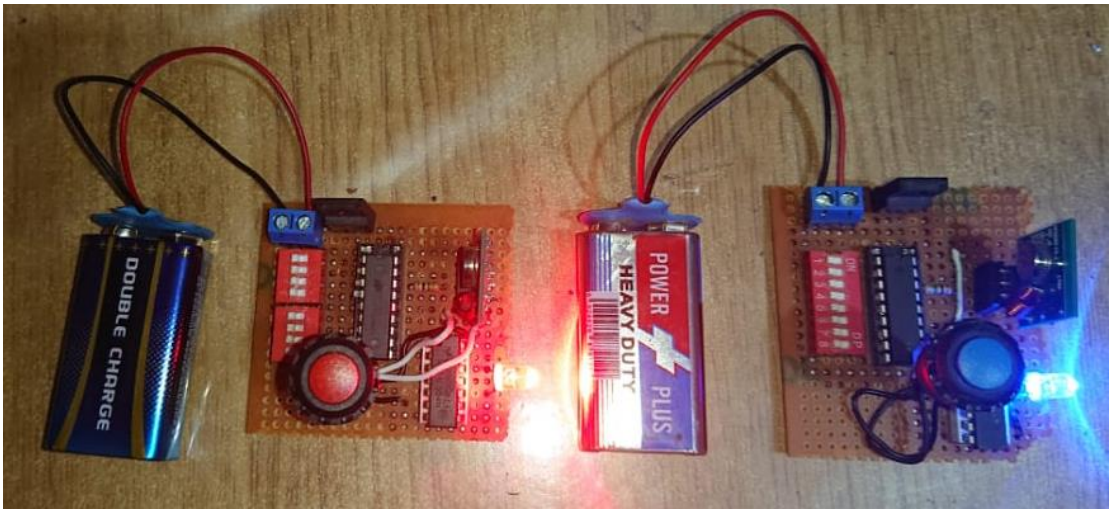
HARDWARE CIRCUIT:

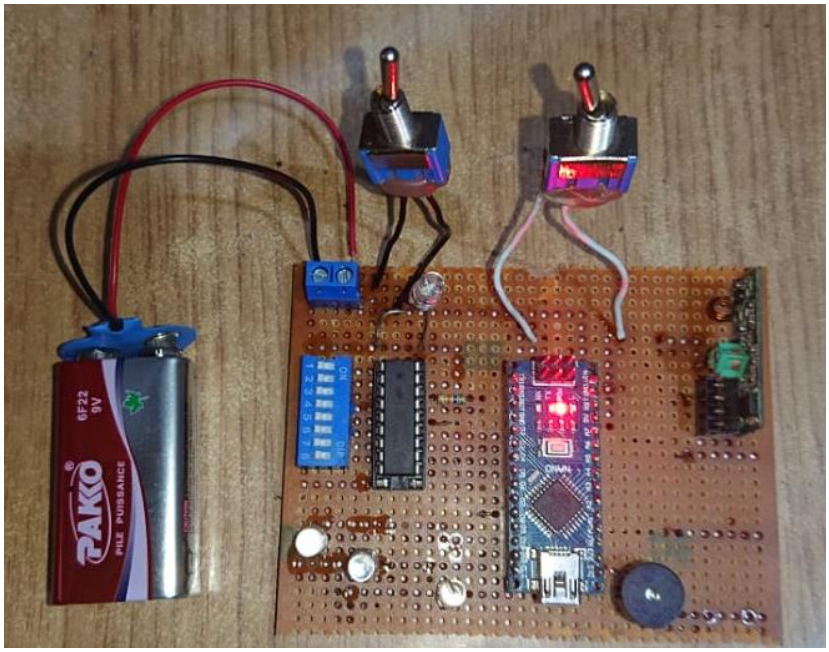
RS = Reset Switch

Rx = Receiver	Buzzer beeps = Buzzer is On with cont. delay
Dec = Decoder IC	Buzzer ON = Buzzer is On cont. w/o delay
G led = Green led	
B = Blue led	
R = Red led	

	RS	Data bits								Buzzer	G Led	B led	R led	Dec	Rx

Case	Data bits @ Decoder	PORTD (input)								PORTB (output)						Comments	Output
		D7	D6	D5	D4	D3	D2	D1	D0	-	-	D13	D12	D11	D10	D9	D8		
1	1010	0	0	1	0	1	0	0	0	0	0	1	0	0	1	1	1	Team B pressed	Red led & buzzer ON
2	1100	0	0	1	1	0	0	0	0	0	0	1	0	1	0	1	1	Team A pressed	Blue led & buzzer ON
3	1111	0	0	1	1	1	1	0	0	0	0	0	0	0	0	1	1	When no one presses, all bits get high after some time	No Output
4	0	1	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	Data bits are cleared	Supply cutoff for Decoder IC & Receiver Green led is ON & Buzzer beeps





CONCLUSION:

Wireless buzzer systems provide a robust and versatile solution for remote alerting and signaling needs. Through the integration of the 433MHz FS1000A transmitter module, MX-RM-5V receiver module, Arduino Nano, HT12D decoder IC, HT12E encoder IC, and a NOT gate IC, we have demonstrated the practical application of wireless communication principles in a simple yet effective design. The system's ability to operate without physical connections not only simplifies installation and maintenance but also extends its usability to diverse environments where wiring might be impractical or impossible. The use of radio frequencies ensures reliable transmission over significant distances, making it suitable for applications in quiz shows, class participation, and game shows. In conclusion, the successful implementation of this wireless buzzer system underscores the potential of wireless technology in enhancing communication and control systems. As technology continues to evolve, further innovations in wireless communication are expected to expand the capabilities and applications of such systems, contributing to more efficient and connected environments.

REFERENCES:

- https://youtu.be/OoUSLSvTJdY?si=Xm0-4VQsR_KCze01
- https://youtu.be/b5C9SPVlU4U?si=ncXmRoh9Q7bKVvla
- https://pdf1.alldatasheet.com/datasheet-pdf/view/64409/HOLTEK/HT12E.html
- https://pdf1.alldatasheet.com/datasheet-pdf/view/64409/HOLTEK/HT12D.html
- https://forum.arduino.cc/t/create-a-wireless-16-player-quiz-buzzer-game/610397