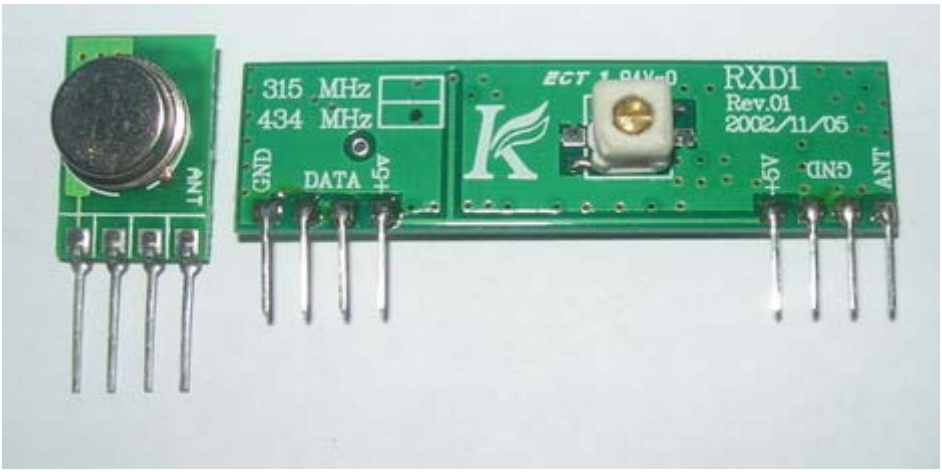
Running TX433 and RX433 RF modules with AVR microcontrollers

admin - Fri, 03/14/2008 - 16:17

Topics:

Sometimes in embedded design you may want to go wireless. Might be you will want to log various readings of remotely placed sensors, or simply build a remote control for robot or car alarm system.

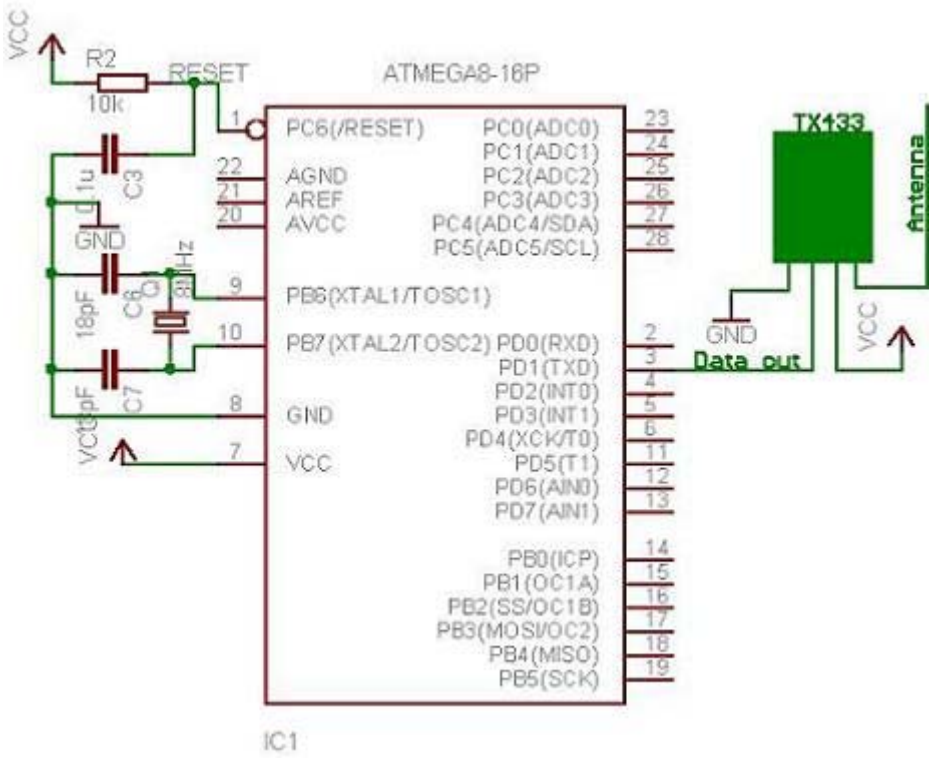
Radio communications between two AVR microcontrollers can be easy when specialized modules are used. Lets try to run very well known RF modules TX433 and RX433 that (or similar) can be found almost in every electronics shop and pair of them cost about ~15 bucks.



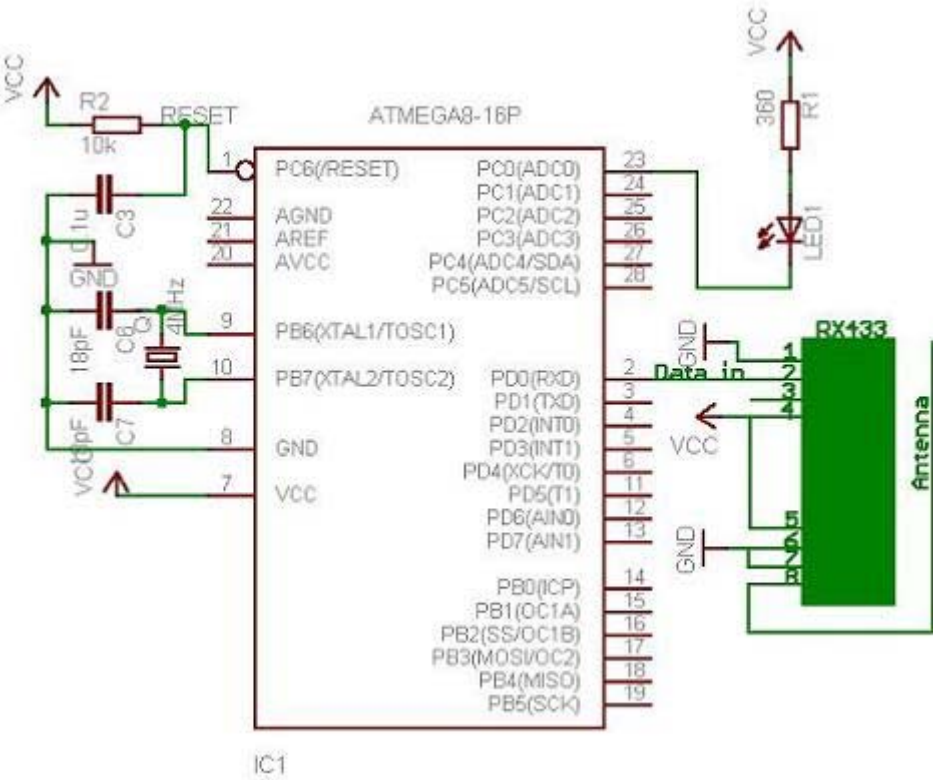
Transmitter and receiver modules are tuned to work correctly at 433.92MHz. Transmitter can be powered from 3 to 12V power supply while receiver accepts 5V. 5V is common for AVR microcontrollers so no problems with interfacing. Modules don't require addition components – just apply power and connect single data line to send information to/from and that's it. For better distances apply 30 – 35cm antennas. Modules use Amplitude-Shift Keying(ASK) modulation method and uses 1MHz bandwidth.

I have constructed two separate circuits for testing on Atmega8 microcontrollers.

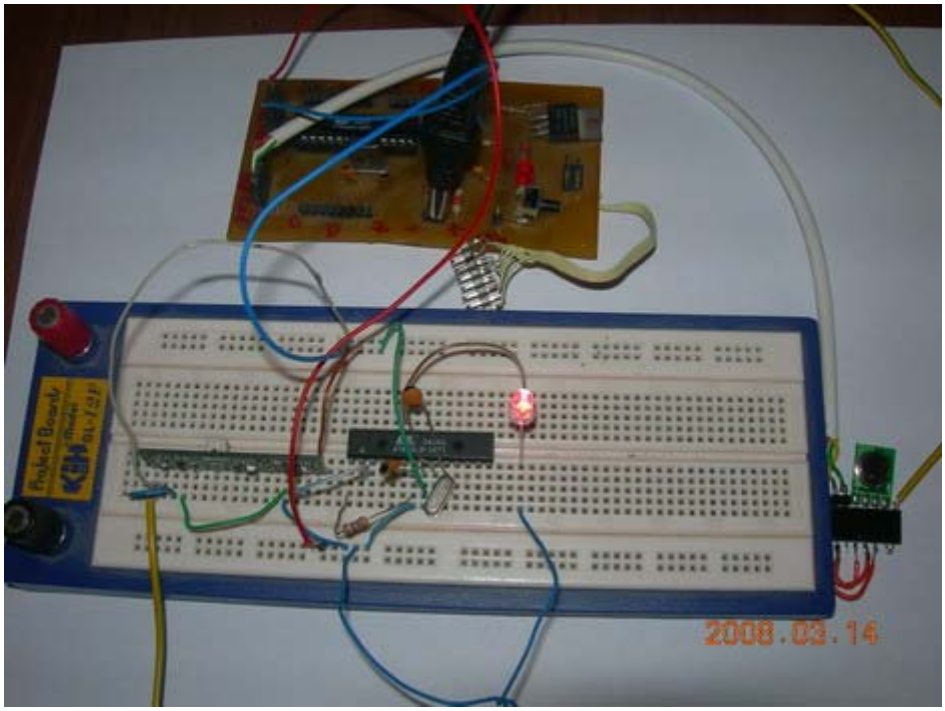
Transmitter



Receiver



For testing I have used a prototyping board and breadboard.



As you can see I have used one LED for indicating RF activity. Ok enough about hardware part – actually there is nothing more to say – circuits are simple.

Lets move on to software part. Radio transmission is a bit more complicated than wired communications because you never know what radio signals are present on air. So all matters how transmitted signal is encoded. And this is a part where you have many choices: use hardware encoding like USART or write your own based on one of many ending methods like NRZ, Manchester etc. In my example I have used AVR USART module to form data packs. Using hardware encoders solves many problems like synchronization, start and stop, various signal checks. But as long as I was practising you cannot rely on plain USART signal. Here you can actually improvize by adding various checks and so on.

I decided to form 4 byte data packages in order to send one byte information. These include:

- one **dummy** synchronization byte (10101010);
- one **address** byte – in case there are more receivers(or transmitters);
- one **data** byte;
- and **checksum** which is actually a sum of address and data(address+data).

Why did I use a dummy byte at the beginning of package. Simply I noticed, that when transmitter doesn't transmit any data – receiver catches various noises that come from power supply or other sources because receiver likes adjust its input gain depending on input signal level. First byte tunes receiver to accept normal signal after then address byte, data and checksum can be read more reliably. Probably with different transmission modules you may exclude this dummy byte.

Ttransmitter program for AVR Atmega8:

```
#include <avr/io.h>
```

```
#include <util/delay.h>
```

```
#ifndef F_CPU
```

```
//define cpu clock speed if not defined
```

```
#define F_CPU 8000000
```

```
#endif

//set desired baud rate

#define BAUDRATE 1200

//calculate UBRR value

#define UBRRVAL ((F_CPU/(BAUDRATE*16UL))-1)

//define receive parameters

#define SYNC 0XAA// synchro signal

#define RADDR 0x44

#define LEDON 0x11//switch led on command

#define LEDOFF 0x22//switch led off command

void USART_Init(void)

{

        //Set baud rate

        UBRRL=(uint8_t)UBRRVAL;         //low byte

        UBRRH=(UBRRVAL>>8);     //high byte

        //Set data frame format: asynchronous mode,no parity, 1 stop bit, 8 bit siz

        UCSRC=(1<<URSEL)|(0<<UMSEL)|(0<<UPM1)|(0<<UPM0)|

                (0<<USBS)|(0<<UCSZ2)|(1<<UCSZ1)|(1<<UCSZ0);

        //Enable Transmitter and Receiver and Interrupt on receive complete

        UCSRB=(1<<TXEN);
```

```
}

void USART_vSendByte(uint8_t u8Data)

{

    // Wait if a byte is being transmitted

    while((UCSRA&(1<<UDRE)) == 0);

    // Transmit data

    UDR = u8Data;

}

void Send_Packet(uint8_t addr, uint8_t cmd)

{

        USART_vSendByte(SYNC);//send synchro byte

        USART_vSendByte(addr);//send receiver address

        USART_vSendByte(cmd);//send increment command

        USART_vSendByte((addr+cmd));//send checksum

}

void delayms(uint8_t t)//delay in ms

{

uint8_t i;

for(i=0;i<t;i++)

        _delay_ms(1);
```

```
}

int main(void)

{

USART_Init();

while(1)

        {//endless transmission

        //send command to switch led ON

        Send_Packet(RADDR, LEDON);

        delayms(100);

        //send command to switch led ON

        Send_Packet(RADDR, LEDOFF);

        delayms(100);

        }

        return 0;

}
```

In my case I used UART 1200 baud rate. It may be increased or decreased depending on distance and environment. For longer distances lower baud rates works better as there is bigger probability for transmission errors. Maximum bit rate of transmitter is 8kbits/s what is about 2400 baud. But what works in theory usually do not work in practice. So 1200 baud is maximum what I could get working correctly.

Transmitter sends two commands (LEDON and LEDOFF) to receiver with 100ms gaps. Receiver recognizes these commands and switches LED on or off depending on received command. This way I can monitor if data transfer works correctly. If LED blink is periodical – then transmission goes without errors. If there is an error in received data then LED gives shorter blink.

Receiver program code:

```
#include <avr/io.h>
```

```
#include <avr/interrupt.h>

#include <util/delay.h>

#ifndef F_CPU

//define cpu clock speed if not defined

#define F_CPU 4000000

#endif

//set desired baud rate

#define BAUDRATE 1200

//calculate UBRR value

#define UBRRVAL ((F_CPU/(BAUDRATE*16UL))-1)

//define receive parameters

#define SYNC 0XAA// synchro signal

#define RADDR 0x44

#define LEDON 0x11//LED on command

#define LEDOFF 0x22//LED off command
```

void USART_Init(void)

```
{

        //Set baud rate

        UBRRL=(uint8_t)UBRRVAL;         //low byte

        UBRRH=(UBRRVAL>>8);     //high byte

        //Set data frame format: asynchronous mode,no parity, 1 stop bit, 8 bit siz
```

```
        UCSRC=(1<<URSEL)|(0<<UMSEL)|(0<<UPM1)|(0<<UPM0)|
                (0<<USBS)|(0<<UCSZ2)|(1<<UCSZ1)|(1<<UCSZ0);
        //Enable Transmitter and Receiver and Interrupt on receive complete
        UCSRB=(1<<RXEN)|(1<<RXCIE);//|(1<<TXEN);
        //enable global interrupts
}
uint8_t USART_vReceiveByte(void)
{
    // Wait until a byte has been received
    while((UCSRA&(1<<RXC)) == 0);
    // Return received data
    return UDR;
}
ISR(USART_RXC_vect)
{
        //define variables
        uint8_t raddress, data, chk;//transmitter address
        //receive destination address
        raddress=USART_vReceiveByte();
        //receive data
```

```
        data=USART_vReceiveByte();
        //receive checksum
        chk=USART_vReceiveByte();
        //compare received checksum with calculated
        if(chk==(raddress+data))//if match perform operations
        {
                //if transmitter address match
                if(raddress==RADDR)
                        {
                                if(data==LEDON)
                                        {
                                                PORTC&=~(1<<0);//LED ON
                                        }
                                else if(data==LEDOFF)
                                        {
                                                PORTC|=(1<<0);//LED OFF
                                        }
                                else
                                {
                                        //blink led as error
```

```
                                        PORTC|=(1<<0);//LED OFF
                                        _delay_ms(10);
                                        PORTC&=~(1<<0);//LED ON
                                    }
                                }
        }
}
void Main_Init(void)
{
        PORTC|=(1<<0);//LED OFF
        DDRC=0X001;//define port C pin 0 as output;
        //enable global interrupts
        sei();
}
int main(void)
{
        Main_Init();
        USART_Init();
        while(1)
        {
```

```
        }
```

```
        //nothing here interrupts are working
```

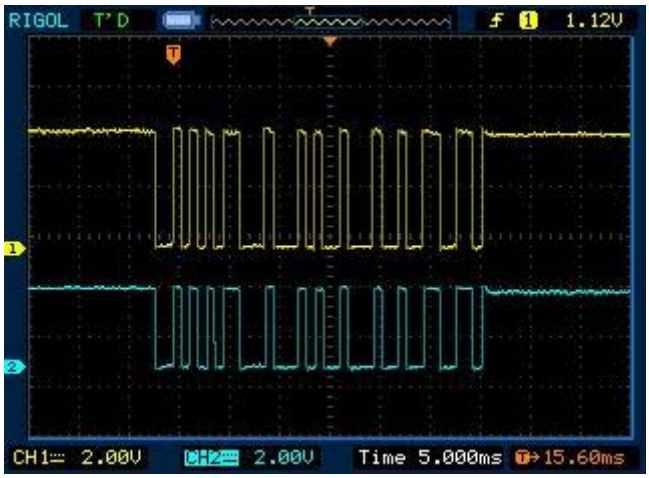
```
        return 0;
```

```
}
```

Receiver program receives all four bytes, then checks if checksum of received bytes is same as received checksum value. If checksum test passes then receiver addresses are compared and if signal is addressed to receiver it analyses data.

After all I have noticed that without antennas transmission is more erroneous even if modules are standing near by. Of course with all my power chords around the room I was getting lots of noises that receiver was catching between data transmissions.

In the last pictures you can see data packets of 4 bytes seen on the oscilloscopes. Yellow signal is from transmission data line(TX) while blue is taken from receiver data line(RX):



Transmitted and received signals matches

There is no noise between data packages(ideal case)

Good luck if you will decide to go wireless.

File:

RF433.zip

Comments

About dummy byte

admin - Fri, 03/14/2008 - 17:21

If you have any questions or suggestions feel free to comment.

Query for connection

Anonymous (not verified) - Sat, 06/16/2012 - 06:14

Sir,
In datasheet of ASK 434 transmitter and receiver ,It is written that supply current should around 2 mA. So how we should connect to ensure a 2mA current. Sir plz reply me as I am facing problem for receiving module.

Thanks

Not getting response from RX

Anonymous (not verified) - Sat, 06/16/2012 - 09:51

Sir
I did whatever written by you in this article but I didn't get any response. I think receiver is not receiving ant data. Is there any problems due to supply current to ASK module.

Modular programming

mrx23 - Tue, 03/25/2008 - 10:54

Please write a simple article about how to do modular programming with WinAVR + AVRstudio in C.With multiple .C source files, headers, global variables, prototypes, common #define-s.I couldn't find any useful resources on the net about this.Thank you!

Modular programming

admin - Tue, 03/25/2008 - 17:49

I will consider this. Thank you for suggestion.

Help with a TX433

bfrye - Tue, 06/17/2008 - 03:56

I just bnought a pair of the RX/TX 433 pair. I am trying to get the things running... My very first step was to plug in the receiver and look at the data line. It is all over the place! THen I hooked up the tranmistter and just hard wired the data line to +5V. I expected to see the receiver signal doing something reasonable. It seems like no matter what I am doing I have garbage on the data line. I have an antenna and everthinh... Any input?Brandon

tx/rx433

bfrye - Tue, 06/17/2008 - 17:07

(Hopefully this is not a re-post). I am trying use these modules in my wireless applicatoin. I am having a lot of trouble. I seem to be getting nothing but noise on the data pin... 30 cm antenna, modules are several feet apart, proper grounding etc... Any ideas? When I put the scope on the data line of the receiver, I get garbage. The transmitting/powered down/not transmitting seems to make little difference. Am I missing something?

Rx Tx $22Mhz Atmega128

naveen89 - Sun, 02/21/2010 - 06:00

Sir,I have Atmega128. I m trying to run your code on my Microcontroller but m not able to get any output.I modified code for Atmega128, Since your code was for Atmega8.F_CPU to 16000000BAUDRATE 19200For TxUBRR1L=0x67;//Lower UBRR1H=0x00;//Higher bits for UBRR=103 UCSR1A=0x00; UCSR1B=0x18;//Enable Reciever and Transmitter UCSR1C=0x06;For RxUBRR1L=0x67;//Lower UBRR1H=0x00;//Higher bits for UBRR=103 UCSR1A=0x00;

UCSR1B=0x18;//Enable Reciever and Transmitter UCSR1C=0x06;But still I m not getting any output.Please help.Regards

ATMEGA 128

fran52 - Mon, 08/11/2014 - 04:56

not suitable –please select a different chip App notes state they Inow about the communication problemsThis is a dead chip–why are you insisting to use it?????Stop wasting everyones time- get rid of this chip (at least for UART projects)pick a different chip (like those mentioned in the project) and stop spewing the webwith crap about the atmega128

Non-inverted TXD line.

kazkowicz - Mon, 05/03/2010 - 15:33

Idle TXD line is held high. IMHO this will cause continuous carrier transmission when no data is being sent. TXD signal should be inverted before feeding to the transmitter and inverted once again after receiver to restore correct logic levels. Simple transistor inverters will do the job. If for some reason receiver is connected to PC COM port, the second inversion is not necessary.

It's your baud rate

NOVAfarmer - Sun, 07/25/2010 - 15:28

I didn't look through your whole code, but the first problem I saw was that you are trying to use a baud rate of 19200. These radio modules can only be used at 2400 max.

adding different remotes?

sherazi11 - Sun, 08/15/2010 - 09:36

i am starting a fixed code remote project ... and i am really thankfull to you for this great tutorial... but i have some querries? and that is that i want the recever to be locked to a number of transmitters... so how i am going to do it...should i program each transmitter with different address? and place a push button if pressed the reciever takes the transmitted signal and stores the address in its eeprom.. and afterwards it matches each signal recieved with that codes and makes a valid or in valid decession...can u help me to make a professional approach regards

No output from the receiver

piyushpandey - Sun, 04/03/2011 - 12:57

Hellol am using the Atmega 16 fro receiver and transmitter is connected to the Atmega 8 and trying to light the LED like you to check whether my transmitters and receivers are working correct or not.But what I have got is that the transmitter is transmitting the data continuously as the led connected to the transmitter pin is blinking continuously and the receiver pin led is just on throughout, and I am not getting the output I mean the led connected to my C0 pin is not blinking as hoped , so can you give me any idea why is this so.Thank you :cry:

I am getting same problem

amitkumar.singal - Tue, 02/21/2012 - 10:24

Dear friend, I am getting same problem & using atmega16. Did you find solution for your problem. If got, please tell me.

3.3V for the reciever

fran52 - Mon, 08/11/2014 - 04:58

Hope you didn't burn out your chip by hooking it to 5V

fst-4 czs-3 modules 315 MHz

Anonymous (not verified) - Fri, 09/09/2011 - 07:16

please, i'd like to know if that program work for fst-4 and czs-3 modules 315 MHz.

433mhz topic --NOT 315mhz

fran52 - Mon, 08/11/2014 - 04:45

Start a new topic DO NOT STEAL CURRENT THREAD

timing

Anonymous (not verified) - Fri, 09/09/2011 - 16:22

I've just bought these and the timing from tx to rx using 1Khz square wave is not 50/50 on the Rx pin. It's 60 /40...

HELLO!!

Anonymous (not verified) - Mon, 09/12/2011 - 19:48

HELLO!!
I AM MAKING A HOMEMADE HELICOPTER BUT WANT TO OPERATE THR TRANSMITTER AND RECEIVER PLS TELL ME HOW TO INSTALL RECEIVER ON HELICOPTER ,AND HOW TO CONFIGURE RF CONTROLLER

proticall

abpccpba - Wed, 04/03/2013 - 05:51

When you type in all caps means that your screaming.

I was searching for this...

Anonymous (not verified) - Thu, 09/29/2011 - 14:35

I was searching for this... This is handy when you are developing real time systems....Thankx for the Post!!.. :)

noise appear on data pin

Anonymous (not verified) - Wed, 10/12/2011 - 10:40

hi everybody!
I use RX-315,TX-315 when I test Rx,Tx separately no problem and everything work good,but when I place my micro(mega8) on bread board (and use shared power supply)I get noise on data pin of also I test it with PT2272 & PT2262 the noise still on data pin but PT2272 work properly and the noise not make any problem!I used Schmitt-trigger(4093) on input of micro but nothing changed and the problem not solved so I think its better to use span/zero and amplifier circuit on input of micro!any idea or suggestion?
tnx 4 all
md.electronic@gmail.com

There is some thing called RF

Anonymous (not verified) - Sat, 11/05/2011 - 11:21

There is some thing called RF layout practices which you need to follow to get noiseless operation with better range. The supply to the module should be passed through a choke and 100Nf caps inorder to filter any garbages in the supply which could have been picked up by the receiver and looped on. Also connect a ground image (9" for a 315Mhz module) Mean the active antenna and the ground points of the module must be of the same length where the ground section provides an image for the receiving antenna.
Cheers

Amplifier ??
fran52 - Mon, 08/11/2014 - 05:04

In the USA we are regulated on how we transmit. Go over 100 meters and you could be fined.The ARRL (America Radio Relay League) has an excellent handbook that explains about these little transmitters.Download their examples and LEARN (check the datasheets and app notes too)

RF Communication Using Controller 8051
Anonymous (not verified) - Thu, 10/13/2011 - 14:31

Thank you for kind information! Their is no need any HT12E/D which makes data transmission rate very slow!!!!!

Monster Beats Turbine headphone
Anonymous (not verified) - Tue, 11/15/2011 - 11:27

d use shared power supply)I get noise on data pin of also I test it with PT2272 & PT2262 the noise still on data pin but PT2272 work properly and the noise not make any problem!I used Schmitt-trigger(4093) on input of mi

TX Code
Anonymous (not verified) - Sun, 01/08/2012 - 20:02

Whenever i am opening the Tx code in Win Avr i am getting the comments and the code is not compiling...
/usr/bin/sh: test : Code:binary operator expected
make.exe: *** No rule to make target 'Code', needed by elf.Stop
Please help..

I would suggest to reuse
admin - Mon, 01/09/2012 - 19:28

I would suggest to reuse source codes and create new projects for tx anr rx on your working environment.

Width of bits received data is different from Transmitted data
Anonymous (not verified) - Wed, 01/11/2012 - 18:14

I'm not using a microcontroller yet , i am transmitting the UART frame from PC to the module with a level shifter (MAX 232 in between).

The data received by the Receiver module looks similar to Transmitter module , but it is not of the same width. But the moment i connect the ground pin of Transmitter and Receiver module the data received is perfect. The receiver doesnt seem to receiver signal over RF.

Any ideas , why this behavior ?

can i connect the micrphone
Anonymous (not verified) - Sat, 01/21/2012 - 10:12

can i connect the micrphone with the above transmitter section .if yes, then at which pin..help me out

doubts
Anonymous (not verified) - Tue, 02/07/2012 - 08:25

I read your receiver code, but this code did not receive the sync byte, my question is why?

RECEIVER PROGRAM

Anonymous (not verified) - Sun, 02/12/2012 - 07:42

can you help me?, I have a doubt, where did you read the sync byte in the receiver programa?

Not getting output on PC0

amitkumar.singal - Tue, 02/21/2012 - 10:26

Hellol am using the Atmega 16 fro receiver and transmitter is connected to the Atmega 8 and trying to light the LED like you to check whether my transmitters and receivers are working correct or not.But what I have got is that the transmitter is transmitting the data continuously as the led connected to the transmitter pin is blinking continuously and the receiver pin led is just on throughout, and I am not getting the output I mean the led connected to my C0 pin is not blinking as hoped , so can you give me any idea why is this so.Thank you

flicker observed in seven segment display

Anonymous (not verified) - Tue, 03/20/2012 - 08:58

I have slightly modified this to transmit and receive numbers and observe flicker at the receiving stage seven segment display.Its like if number 18 to be displayed it displays 1 for few seconds and 8 for few seconds . Any suggestions how to solve this problem?

in my case i've many

Anonymous (not verified) - Tue, 03/27/2012 - 15:49

in my case i've many transmitters and just one receiver i just wish to know if all the transmitters start transmitting at the same in what order receiver will receive it and i wish to catch all the signals but ali the transmitter will transmit only once.is there ne solution for dis other than using a transceiver....???please help...

Non fonctionnal on atmega 168

Anonymous (not verified) - Fri, 03/30/2012 - 21:32

Hello ! I try for a school project to create this project with 2 ATMEGA168 and its non functional ...
I have modify the USART to USART1 but it's non functional, the TX signal is inverted !
You have a solution ?

thanks !

Cap values

Anonymous (not verified) - Fri, 04/13/2012 - 22:17

I think I can read the other cap values but the C7 val is overlapped with VCC. What's the value of C7?

Thanks.

Oh sorry for that. C7 = 18pF

admin - Fri, 04/20/2012 - 14:04

Oh sorry for that. C7 = 18pF

request for spi tx and rx

Anonymous (not verified) - Tue, 04/24/2012 - 14:23

can u plz help me coding in c for tx and rx module interfacing with atmega8 using spi protocol.

so many people

Anonymous (not verified) - Thu, 06/28/2012 - 13:15

are too lazy to do their own homework and research!

Hello Sir!
Anonymous (not verified) - Tue, 10/16/2012 - 23:59

Hello Sir!
Please Tell Me, How To Connect This RF433 To My Wall Socket?

How to write fuse bits on Atmega8
Anonymous (not verified) - Fri, 10/19/2012 - 03:09

Hi, I need your help how to write fuse bits (Low fuse and high fuse) on Atmega8 (8Mhz and 4Mhz).

rf
Anonymous (not verified) - Thu, 12/06/2012 - 19:19

well when i design rf receiver & atmega16 on one PCB it wont work .. separately it works ..is there any special care has to take...

help about decode
bluepawn - Tue, 02/19/2013 - 13:24

I want to know how do I use a 4-channel remote control which key is presed
thank so much

help about decode
bluepawn - Tue, 02/19/2013 - 13:26

I want to know how do I use a 4-channel remote control which key is presed
thank so much

my email address technologypawn@gmail.com

RF [car alarm]
Boliveira2 - Thu, 04/04/2013 - 17:29

Sir,

I have a remote control (4 buttons) from my older car alarm and it works on 433kHz and i want to do a homemade alarm... so i never work with rf and i want to know how i can "read" and "understand" what remote control "sais". what baud rate? i know only the Freq that it oppers...
Can you help me please.
Thank you.

oscilloscope
Peio47 - Sat, 04/27/2013 - 17:20

How do you manage to see data packets with oscilloscope ?
On the oscillosope, the frequency is 1Mhz instead of 433.

multiple transmitters working???
anil99e - Wed, 02/05/2014 - 08:21

sir,does this work if there are more than one transmitters and a single receiver? will interference be a problem?

Multiple transmitters of this

admin - Thu, 02/06/2014 - 18:25

Multiple transmitters of this type tend to interfere. So this one isn't a good choice for multi solution.

RF SEND AND RECEIVE SIMULTANEOUSLY

ketan - Sun, 02/09/2014 - 20:31

Sir , i am having problem in usart communication through rf (both sender and receiver connected to a single atmega32). such that i can have two way communication between two such micro controllers ie. if a command is send to another then a acknowledgement command should be received to sender. but i am having some coding issues. If both transmitter and receiver is connected to same micro controller then it behaves in different manner. SIR PLZ HELP IN FIXING THIS ISSUE....
THE CCODE IS :-

```
// usart_text_transmit.c

#include <avr/io.h>
#include "usart.h"
#include <avr/interrupt.h>
#include <avr/delay.h>
#include <string.h>

int main(void)
{

USART_Init(51);

char str_s1="1234";
char str2[]="data 1 received!";//first string to send
char str_s[]={"data 2 received!"};//second string to send
char str1[100]; //our buffer to record received bytes
_delay_ms(200);
while(1)
{

WriteStringData(str2);//usart commad to send string
_delay_ms(1000);//a little wait for other chip execute received bytes

WriteStringData(str_s);
//_delay_ms(1000);

//if we are waiting data dont use delay as we wait delay we cant read received bytes
//actualy usart waiting for data

ReadStringData(str1);//usart command to receive string
//LCD_clear();
//LCD_write_string(str1);

WriteStringData(str1);
_delay_ms(1000);

}
}
/*********THE USART HEADER FILE IS – */

#include <avr/delay.h>

void USART_Init( unsigned int ubrr)//initialize usart with ubrr value
{
/* Set baud rate */
UBRRH = (unsigned char)(ubrr>>8);
UBRRL = (unsigned char)ubrr;
/* Enable receiver and transmitter */
UCSRB = (1<<RXEN)|(1<<TXEN);
/* Set frame format: 8data, 2stop bit */
UCSRC = (1<<URSEL)|(1<<USBS)|(3<<UCSZ0);
}
```

```
char ReadData( void )// this function read onyl one byte
{
/* Wait for data to be received */
while ( !(UCSRA & (1<<RXC)) )
;
/* Get and return received data from buffer */
return UDR;
}

void WriteData(char data)//this function write only one byte
{
//Wait For Transmitter to become ready
while(!(UCSRA & (1<<UDRE)));

//Now write
UDR=data;
}

//this function receive long string chars
//make sure buffer size is enough
//firsly read from usart the data and write to string array
//after compare last data with '\0' if it is null teminate the reading
void ReadStringData(char *str){

char c;
do{
c=ReadData();
*str=c;
str++;
}
while(c!='\0');
// LCD_clear();

return;

}

//this function send string daha via usart byte by byte
//firsly read from byte pointer adress
//make sure readed char is not null char
//send data and increase pointer to read next char
//if readed char is null ('\0') end the transmisson after sent terminater null char
void WriteStringData(char *strData)
{
while(*strData!='\0')
{
WriteData(*strData);
strData++;
}
WriteData('\0');
return;

}
```

THERE IS SOME CODING ISSUE .. AS WHEN TESTED SEPARATELY THEY DO WORK FINE :-/
PLZZ HELP