

```
//
// exercise 13.4rx - http://tronixstuff.com/tutorials > Chapter 13
//
// control two digital outputs from wireless transmitter system - this sketch uses example 13.2tx for
// a transmitter
// each output will be controlled by an on and off button
//
// based on code by Mike McCauley 2010 http://www.open.com.au/mikem/arduino
//

// Need these lines below //////////////////////////////////////
#include <VirtualWire.h>
#undef int
#undef abs
#undef double
#undef float
#undef round
uint8_t buf[VW_MAX_MESSAGE_LEN]; // this is an array of unsigned integers 8-bits long. In other words,
bytes between 0 and 65535
uint8_t buflen = VW_MAX_MESSAGE_LEN;

int c1 = 0; // these will store the status of the channels (low = 0, high = 1)
int c2 = 0;

////////////////////////////////////
void setup()
{
  // wake up the wireless receiver

  vw_set_ptt_inverted(true); // need this line
  vw_setup(2400); // sets speed of data reception.
  vw_set_rx_pin(0); // this is the RX pin number - 0 on a Duemilanove
  vw_rx_start(); // start the receiver!
  pinMode(4, OUTPUT); // LED 4
  pinMode(6, OUTPUT); // LED 3
  pinMode(8, OUTPUT); // LED 2
  pinMode(9, OUTPUT); // LED 1
}

void loop()
{
  // check to see if there is received data in the buffer, and that it came through correctly.
  // if the message didn't come through completely, it will be ignored
  if (vw_get_message(buf, &buflen))
  {
    switch(buf[0])
    {
      case 'b':
        if (c1==0)
        {
          digitalWrite(9, HIGH); // if off, turn it on
          c1=1;
        }
        break;
      case 'd':
        if (c1==1)
        {
          digitalWrite(9, LOW); // if on, turn it off
          c1=0;
        }
        break;
      case 'f':
        if (c2==0)
        {
          digitalWrite(8, HIGH); // if off, turn it on
          c2=1;
        }
        break;
      case 'h':
        if (c2==1)

```

```
    {  
      digitalWrite(8, LOW); // if on, turn it off  
      c2=0;  
    }  
    break;  
  }  
}
```