**ARDUNIO BASED IR REMOTE AND MANUAL HOME AUTOMATION**

#include <SPI.h>  
#include <Wire.h>  
  
#include <IRremote.h>  
  
const int relay\_1 = 12;  
const int relay\_2 = 11;  
const int relay\_3 = 10;  
const int relay\_4 = 9;  
  
  
const int mswitch\_1 = 8;  
const int mswitch\_2 = 7;  
const int mswitch\_3 = 6;  
const int mswitch\_4 = 5;  
  
int RECV\_PIN = 3;  
  
IRrecv irrecv(RECV\_PIN);  
decode\_results results;  
  
  
int toggleState\_1 = 0;  
int toggleState\_2 = 0;  
int toggleState\_3 = 0;  
int toggleState\_4 = 0;  
  
void setup() {  
   
  Serial.begin(9600);  
  irrecv.enableIRIn();  
   
  pinMode(relay\_1, OUTPUT);  
  pinMode(relay\_2, OUTPUT);  
  pinMode(relay\_3, OUTPUT);  
  pinMode(relay\_4, OUTPUT);  
  
  pinMode(mswitch\_1, INPUT\_PULLUP);  
  pinMode(mswitch\_2, INPUT\_PULLUP);  
  pinMode(mswitch\_3, INPUT\_PULLUP);  
  pinMode(mswitch\_4, INPUT\_PULLUP);  
}  
  
void relayOnOff(int relay){  
  
    switch(relay){  
      case 1:  
             if(toggleState\_1 == 0){  
              digitalWrite(relay\_1, HIGH);  
              toggleState\_1 = 1;  
              }  
             else{  
              digitalWrite(relay\_1, LOW);  
              toggleState\_1 = 0;  
              }  
             delay(100);  
      break;  
      case 2:  
             if(toggleState\_2 == 0){  
              digitalWrite(relay\_2, HIGH);  
              toggleState\_2 = 1;  
              }  
             else{  
              digitalWrite(relay\_2, LOW);  
              toggleState\_2 = 0;  
              }  
             delay(100);  
      break;  
      case 3:  
             if(toggleState\_3 == 0){  
              digitalWrite(relay\_3, HIGH);  
              toggleState\_3 = 1;  
              }else{  
              digitalWrite(relay\_3, LOW);  
              toggleState\_3 = 0;  
              }  
             delay(100);  
      break;  
      case 4:  
             if(toggleState\_4 == 0){  
              digitalWrite(relay\_4, HIGH);  
              toggleState\_4 = 1;  
              }  
             else{  
              digitalWrite(relay\_4, LOW);  
              toggleState\_4 = 0;  
              }  
             delay(100);  
      break;  
             
      default : break;        
      }  
   
}  
  
  
void loop() {      
       
    if (digitalRead(mswitch\_1) == LOW){  
      delay(200);  
      relayOnOff(1);        
    }  
    else if (digitalRead(mswitch\_2) == LOW){  
      delay(200);  
      relayOnOff(2);  
    }  
    else if (digitalRead(mswitch\_3) == LOW){  
      delay(200);  
      relayOnOff(3);  
    }  
    else if (digitalRead(mswitch\_4) == LOW){  
      delay(200);  
      relayOnOff(4);  
    }  
  
      if (irrecv.decode(&results)) {  
        switch(results.value){  
          case 0xFD08F7:  
                    relayOnOff(1);  
          break;  
          case 0xFD8877:  
                    relayOnOff(2);  
          break;  
          case 0xFD48B7:  
                    relayOnOff(3);  
          break;  
          case 0xFD28D7:  
                    relayOnOff(4);  
          break;  
          default : break;        
          }      
    irrecv.resume();  
    }  
  }

**SIMULATION:**[https://www.tinkercad.com/things/eVej1FRAdxn](https://www.tinkercad.com/things/eVej1FRAdxn" \t "https://mail.google.com/mail/u/0/" \l "inbox/_blank)

