

Textul și imaginile din acest document sunt licențiate

Attribution-NonCommercial-NoDerivs

CC BY-NC-ND



Codul sursă din acest document este licențiat

Public-Domain

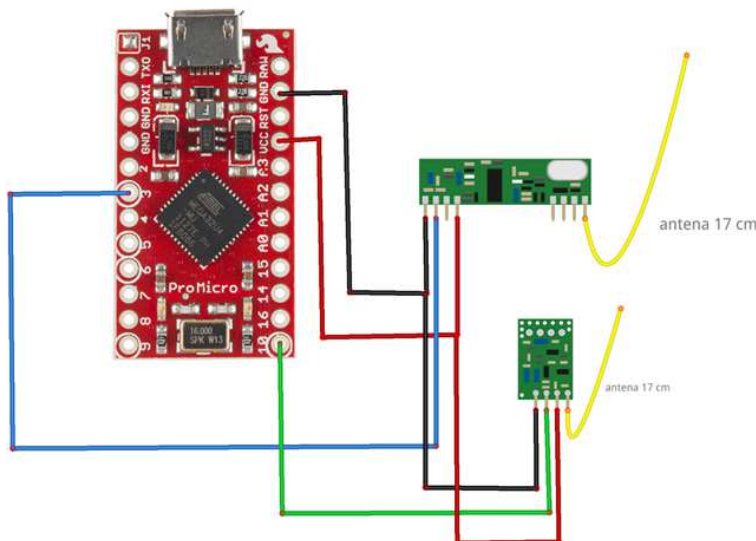
Ești liber să distribui acest document prin orice mijloace consideri (email, publicare pe website / blog, tipărire, sau orice alt mijloc), atât timp cât nu aduci nici un fel de modificări acestuia. Codul sursă din acest document poate fi utilizat în orice fel de scop, de natură comercială sau nu, fără nici un fel de limitări dar autorii nu își asumă nici o răspundere pentru pagubele pricinuite de implementările realizate de utilizatori. Schemele și codul sursă au un rol educativ și nu sunt gândite pentru a fi utilizate în mediu de producție (industrial, casnic sau comercial).

Automatizarea utilizării dispozitivelor radio ASK

Pornind de la sistemul prezentat în lecția anterioară (Convertor ASK USB) putem dezvolta o funcționalitate foarte simplă dar utilă în cazul utilizării mai multor dispozitive radio în bandă ISM (1) de 433MHz modulație ASK (2) – mai ales dacă vorbim de dispozitive de la producători diverși, incompatibile nativ între ele. Această nouă funcționalitate a sistemului ne va permite să automatizăm diverse operații simple ca de exemplu: ”la declanșarea unui senzor de prezență sau de deschidere a ușii să se aprindă lumina într-o anumită cameră,, sau ”la declanșarea unui senzor de gaz să pornească alarma sistemului antiefracție,, etc. Sistemul nostru va face legătura între componente radio ce fac parte din dispozitive complet diferite și va putea fi utilizat indiferent dacă componentele respective mai fac sau nu mai fac parte din dispozitivele inițiale (senzorii și elementele de acționare vor funcționa cu sau fără componenta bază originală).



Vom utiliza aceleași componente ca și în lecția precedentă: placă de dezvoltare Arduino Pro Micro (3) cu un microcontroler ATmega 32U4 la 16MHz / 5V, un transmițător radio (4) și un receptor radio (5) – ambele în bandă de 433MHz modulație ASK. Schema de interconectare este de asemenea la fel (transmițătorul conectat la pinul 10, receptorul la pinul 3):



Programul, dezvoltat și testat utilizând Arduino IDE 1.8.1, extensia Arduino AVR Boards 1.6.19 și bibliotecă rc-switch 2.6.2, este puțin diferit:

```
#include <RCSwitch.h>

RCSwitch mySwitch = RCSwitch();

void setup() {
    while (!Serial);
    Serial.begin(115200);
    mySwitch.enableTransmit(10);
    mySwitch.setRepeatTransmit(3);
    mySwitch.enableReceive(0);
}

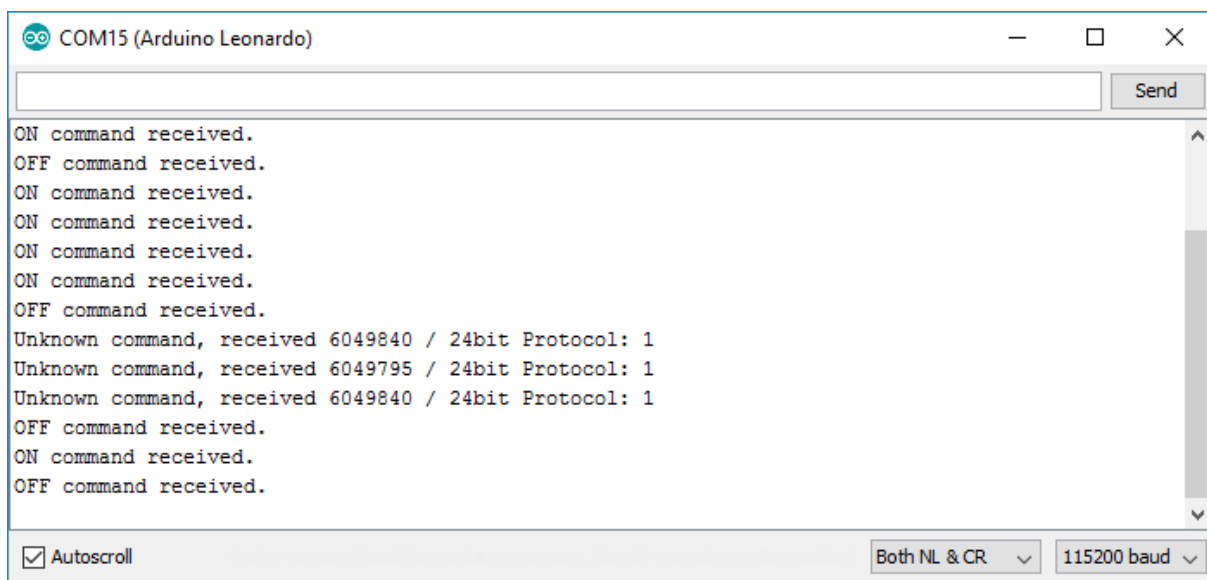
void loop() {
    if (mySwitch.available()) {
        int value = mySwitch.getReceivedValue();
        if (value != 0) {
            switch (mySwitch.getReceivedValue()) {
                case 6083408:
                    Serial.println("ON command received.");
                    mySwitch.setProtocol(2);
                    mySwitch.send(2474994176, 32);
                    break;
                case 6049984:
                    Serial.println("OFF command received.");
                    mySwitch.setProtocol(2);
                    mySwitch.send(2743429632, 32);
                    break;
                default:
                    Serial.print("Unknown command, received ");
```

```

        Serial.print( mySwitch.getReceivedValue() );
        Serial.print(" / ");
        Serial.print(    mySwitch.getReceivedBitlength()
);
        Serial.print("bit ");
        Serial.print("Protocol: ");
        Serial.println(    mySwitch.getReceivedProtocol()
);
        break;
    }
}
mySwitch.resetAvailable();
}
}

```

Se poate observa că în cadrul secțiunii *loop()* se execută o buclă infinită care așteaptă primirea unui cod radio. În funcție de codul radio primit (instrucțiunea *switch(mySwitch.getReceivedValue())*) se va trimite un alt cod radio ca efect al acțiunii dorite.



În cazul exemplului dat s-a utilizat un senzor de deschidere a ușii și o telecomandă radio (ambele făcând parte dintr-un sistem de securitate a locuinței) ca intrări și o priză

Conrad RSL ca element de acționare. Pentru mai multe informații legate de diverse dispozitive radio de securitate se poate parcurge și proiectul "HomeWatch" din cartea "10(zece) proiecte cu Arduino" (6) și pentru mai multe informații legate de dispozitivele din gama Conrad RSL se pot parcurge și materialele "Cloud's Lights" (7) și "Local Area Power Plugs" (8).



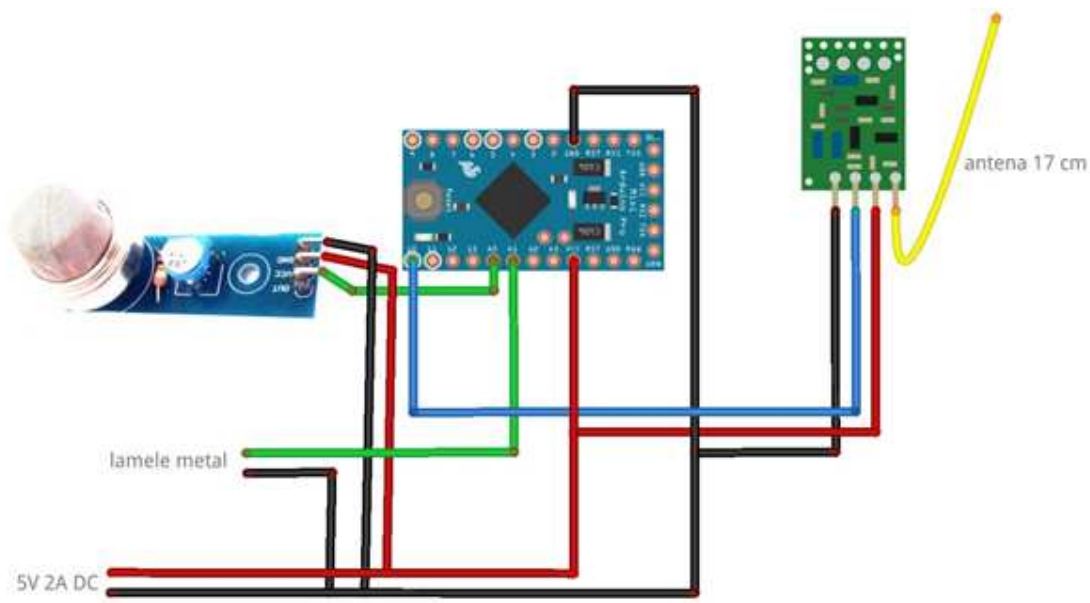
Codul 6083408 este transmis de senzorul de deschidere a ușii și declanșează deschiderea prizei – cod radio transmis 2474994176. Putem considera că priza alimentează o veioză, o sonerie sau o cameră video de securitate.



Codul 6049984 este transmis de butonul de armare al telecomenzii radio și va declanșa transmiterea codului radio 2743429632 de închidere a prizei. Orice alt cod radio, necunoscut, va avea ca efect afișarea în consola serială a mesajului *Unknown command* urmat de codul numeric propriu-zis, acest lucru permite aflarea codurilor ce dorim a fi introduse în program.

Funcționarea sistemului nu este limitată la dispozitive radio prefabricate. Putem cu ușurință să dezvoltăm sisteme simple de supraveghere a parametrilor de mediu sau sisteme de acționare pentru diverse sarcini casnice. Un bun exemplu este prezentat în cadrul proiectului "HomeWatch" din cartea "10(zece) proiecte cu Arduino" (6) – este vorba de un senzor de bucătărie ce permite avertizarea în caz de inundație sau în cazul unor scurgeri de gaze. Senzorul se bazează pe o placă de dezvoltare Arduino Pro Mini (9), un modul senzor de gaz metan MQ4 (10) și un transmițător radio 433MHz ASK (4).

Senzorul de bucătărie poate detecta două evenimente neplăcute ce pot apărea într-o locuință în zona bucătăriei: scurgeri de gaz, senzorul va trimite codul radio 3333310 la apariția acestui eveniment, și scurgeri de apă la nivelul podelei, senzorul va trimite codul radio 3333311 la apariția acestui eveniment. Acest sistem de tip senzor poate fi utilizat cu ușurință împreună cu sistemul de automatizare prezentat putând declanșa o alarmă sonoră sau, de ce nu, un sistem de deschidere a ferestrelor. Schema și codul necesar sunt prezentate în cele ce urmează.



Programul a fost testat utilizând Arduino IDE 1.8.1, extensia Arduino AVR Boards 1.6.19 și bibliotecă rc-switch 2.6.2.

```
#include <RCSwitch.h>

RCSwitch mySwitch = RCSwitch();

volatile unsigned long alarma_apa = 0;
volatile unsigned long alarma_gaz = 0;

const unsigned long interval = 60000;

const unsigned int pin_rf = 10;
const unsigned int pin_gaz = A0;
const unsigned int pin_apa = A1;

const unsigned long cod_gaz = 3333310;
const unsigned long cod_apa = 3333311;

void setup() {
```

```

mySwitch.enableTransmit (pin_rf);
mySwitch.setRepeatTransmit (10);
pinMode (pin_apa, INPUT_PULLUP);
}

void loop() {
  if (analogRead(pin_gaz)>500)
  {
    if (alarma_gaz == 0)
    {
      mySwitch.send(cod_gaz,24);
      delay(3000);
      alarma_gaz = millis();
    }
    else
      if ((millis() - alarma_gaz) > interval)
      {
        mySwitch.send(cod_gaz,24);
        delay(2000);
        alarma_gaz = millis();
      }
  }
  else
    if ((alarma_gaz != 0) && ((millis() - alarma_gaz) >
interval) )
    {
      alarma_gaz = 0;
    }
  if (analogRead(pin_apa)<500)
  {
    if (alarma_apa == 0)
    {

```

```
        mySwitch.send(cod_apa,24);  
        delay(3000);  
        alarma_apa = millis();  
    }  
    else  
        if ((millis() - alarma_apa) > interval)  
        {  
            mySwitch.send(cod_apa,24);  
            delay(2000);  
            alarma_apa = millis();  
        }  
    }  
    else  
        if ((alarma_apa != 0) && ((millis() - alarma_apa) >  
interval) )  
        {  
            alarma_apa = 0;  
        }  
    }  
}
```


Referințe on-line

(1) ISM band

https://en.wikipedia.org/wiki/ISM_band

(2) Amplitude-shift keying

https://en.wikipedia.org/wiki/Amplitude-shift_keying

(3) Pro Micro 5V/16MHz - ATmega 32U4

https://www.robofun.ro/pro_micro_5v_16_mhz?utm_source=newsletter&utm_medium=email&utm_content=productLink&utm_campaign=CURS_EMAIL

(4) Transmisor radio 434 Mhz

https://www.robofun.ro/wireless/wireless-433/transmisor_radio?utm_source=newsletter&utm_medium=email&utm_content=productLink&utm_campaign=CURS_EMAIL

(5) Receptor radio 434 MHz

https://www.robofun.ro/wireless/wireless-433/receptor_radio?utm_source=newsletter&utm_medium=email&utm_content=productLink&utm_campaign=CURS_EMAIL

(6) 10(zece) proiecte cu Arduino

<https://play.google.com/store/books/details?id=aMB8BgAAQBAJ>

(7) Cloud's Lights

<https://blog.robofun.ro/2017/03/28/proiect-clouds-lights/>

(8) Local Area Power Plugs

<https://blog.robofun.ro/2017/03/21/proiect-local-area-power-plugs/>

(9) Arduino Pro Mini 328 - 5V/16MHz

https://www.robofun.ro/arduino/arduino_pro_mini_328_16mhz?utm_source=newsletter&utm_medium=email&utm_content=productLink&utm_campaign=CURS_EMAIL

(10) Kit Senzor Metan MQ4

https://www.robofun.ro/senzori/biometric/kit-senzor-metan-mq4?utm_source=newsletter&utm_medium=email&utm_content=productLink&utm_campaign=CURS_EMAIL