

Textul si imaginile din acest document sunt licentiate

Attribution-NonCommercial-NoDerivs  
CC BY-NC-ND



Codul sursa din acest document este licentiat

Public-Domain

Esti liber sa distribui acest document prin orice mijloace consideri (email, publicare pe website / blog, printare, sau orice alt mijloc), atat timp cat nu aduci nici un fel de modificari acestuia. Codul sursa din acest document poate fi utilizat in orice fel de scop, de natura comerciala sau nu, fara nici un fel de limitari.

# Realizarea unui sistem de tip Home Automation (Partea I)

## *Arhitectura unui sistem de tip Home Automation*

Prin noțiunea de Home Automation se înțelege implementarea unui sistem de automatizare și control centralizat al echipamentelor electronice de automatizare dintr-o locuință. Automatizarea presupune două aspecte importante: monitorizarea și comanda automată. Monitorizarea se realizează prin intermediul unor echipamente electronice de tip senzori și traductoare ce pot măsura diverși parametrii specifici ambientului din locuință (temperatură, umiditate, luminozitate, calitate aer) sau diverși parametrii de funcționare a echipamentelor electronice (consum electric, stare pornit / oprit, căldură degajată, presiune sau debit apă etc.). Partea de comandă presupune capacitatea sistemului de a reacționa și de a interacționa cu echipamentele electronice și electrocasnice (aparate de climatizare, sisteme de securitate, sisteme multimedia, sisteme de alimentare cu energie electrică) prin intermediul unor elemente de acționare. Conexiunea dintre parametrii mășurați și comenzile automate se face prin intermediul unui sistem de comandă centralizat care poate acționa la comanda manuală a utilizatorului sau poate avea comportamente de comandă predefinite (dacă temperatura în cameră depășește 35 de grade Celsius pornește aerul condiționat).

Astfel, arhitectura unui sistem de tip Home Automation implică următoarele subsisteme:

- Elemente de achiziție de tip senzori (sensors);
- Elemente de acționare (actuators);
- Sistem central de comandă (controller);

Legătura dintre elementele de achiziție, elementele de acționare și sistemul de comandă implică o rețea de comunicație. La momentul actual majoritatea sistemelor utilizează o rețea de comunicație fără fir datorită comodității de instalare și operare. Există mai multe produse comerciale ce oferă soluții complete de Home Automation dar acestea sunt incompatibile între ele (nu se poate folosi un senzor de tipul X cu un sistem de comandă de tipul Y) și au un cost destul de ridicat.

Exemple de soluții comerciale de Home Automation:

Insteon

<http://www.insteon.com/>

Z-Wave Home control | Z-Wave Smart Home

<http://www.z-wave.com/>

Nest

<https://nest.com/>

În cadrul seriei "Realizarea unui sistem de tip Home Automation" vom prezenta o alternativă la soluțiile comerciale utilizând doar elemente open-source (atât hardware cât și software) care asigură un cost mult mai scăzut și o inter-operabilitate mult mai bună cu sisteme similare (open-source sau comerciale).

În cadrul materialului de față vom prezenta o soluție de rețea de interconectare fără fir pentru elementele de achiziție bazată pe platforma MySensors:

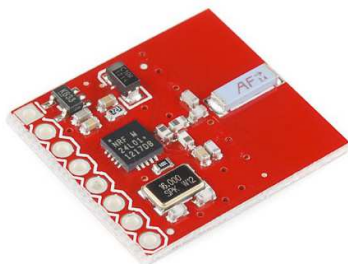
<https://www.mysensors.org/>



## ***Realizarea unei rețele fără fir de elemente de automatizare***

Platforma MySensors este un proiect open-source bazat pe hardware și software compatibil Arduino ce își propune să ofere o soluție completă pentru realizarea de la zero a elementelor de achiziție, a elementelor de acționare și a rețelei radio fără fir între elementele acestea și un element central de comandă. Proiectul nu include partea software de comandă centralizată dar este compatibil cu alte proiecte open-source ce implementează această parte: OpenHab (pe care îl vom prezenta într-un material viitor), MyController.org, MyNodes.NET etc.

Proiectul poate utiliza pentru transmisia radio fără fir două tipuri de module radio (cele două tipuri de module nu pot fi mixate în aceeași rețea):



Module radio bazate pe circuitul nRF24L01+ de la Nordic Semiconductors în bandă ISM de 2.4GHz modulație GFSK – aceste module vor fi utilizate în cadrul implementării de față.

[https://www.robofun.ro/wireless/wireless-2-4/transceiver\\_nRF24L01\\_antena](https://www.robofun.ro/wireless/wireless-2-4/transceiver_nRF24L01_antena)

<https://www.robofun.ro/forum/>



Module radio bazate pe circuitul RFM69HCW de la HopeRF Electronics în bandă ISM de 433MHz sau 868MHz modulație FSK/GFSK/MSK/GMSK/OOK.

<https://www.robofun.ro/wireless/wireless-433/sparkfun-rfm69-breakout-434mhz->

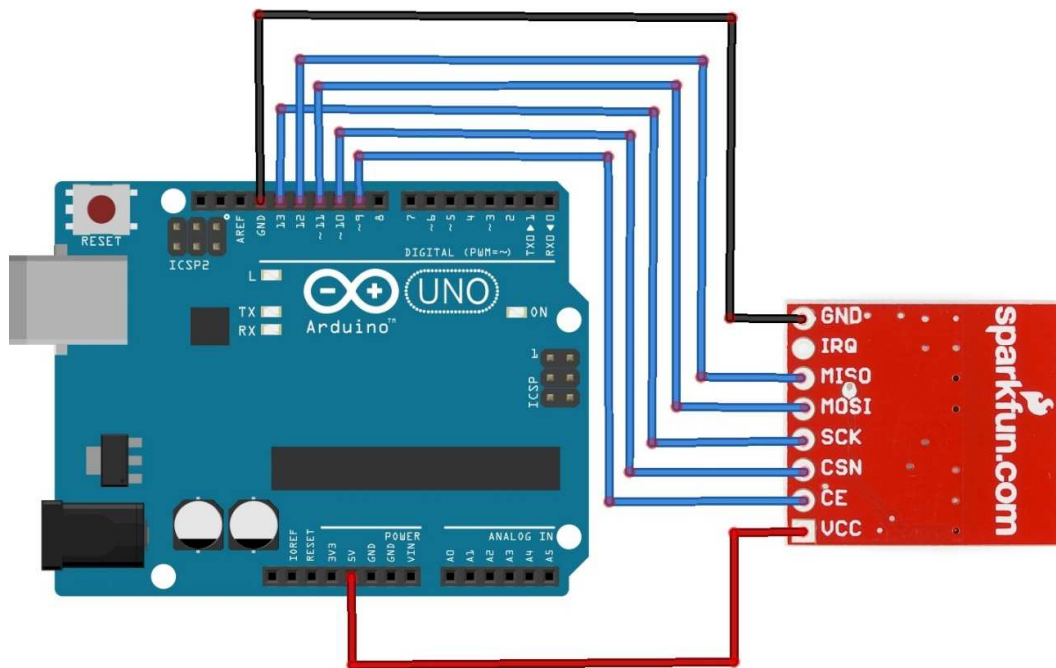
Pentru implementarea soluție de rețea fără fir vom avea nevoie de două plăci Arduino Uno sau similare și două module radio nRF24L01+. Ca și plăci de dezvoltare putem utiliza orice variantă de placă Arduino Uno:

[https://www.robofun.ro/arduino/arduino\\_uno\\_v3](https://www.robofun.ro/arduino/arduino_uno_v3)

<https://www.robofun.ro/arduino/redboard>

[https://www.robofun.ro/platforme/arduino\\_dev/arduino-industrial](https://www.robofun.ro/platforme/arduino_dev/arduino-industrial)

Conexiunea dintre modulul radio și placa de dezvoltare se realizează prin magistrala SPI și presupune următoarele conexiuni: Vcc – 5V, CE – D9, CSN – D10, SCK – D13, MOSI – D11, MISO – D12, GND – GND.



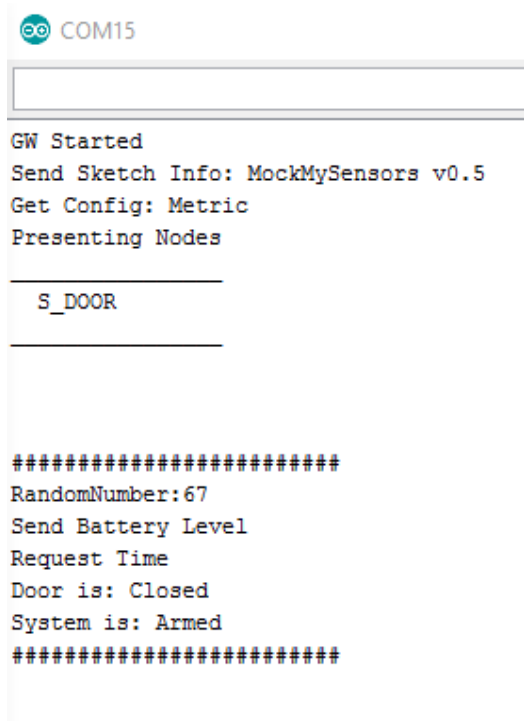
<https://www.robofun.ro/forum/>

După realizarea celor două montaje vom trece la programarea acestora. Vom folosi Arduino IDE 1.6.9 împreună cu biblioteca software MySensors 2.0.0 (ultima versiune apărută pe 9 iulie 2016).

<https://github.com/mysensors/MySensors/tree/master>

Din cadrul librăriei vom utiliza două exemple pe care le vom încărca pe cele două montaje realizate: MockMySensors și GatewaySerial.

**MockMySensors** – este un exemplu ce simulează toate tipurile de senzori și elemente de acționare definite în cadrul bibliotecii; îl vom utiliza pentru testarea comunicației radio și pentru înțelegerea modului de funcționare a instrumentelor puse la dispoziție de aceasta. Acest exemplu este un bun punct de plecare în implementarea oricărui tip de element de măsurare sau de acționare. Execuția sa (cu directiva *#define MY\_DEBUG* comentată) va avea următorul efect în consola serială:



```
COM15

GW Started
Send Sketch Info: MockMySensors v0.5
Get Config: Metric
Presenting Nodes


S_DOOR

#####
RandomNumber:67
Send Battery Level
Request Time
Door is: Closed
System is: Armed
#####
```

Comportamentul implicit al exemplului este mixt: senzor de ușă – S\_DOOR (open/closed) și element de acționare încuietore – S\_ARMED (armed/disarmed). Operațiile efectuate de sistem sunt următoarele: **INIȚIALIZARE COMUNICAȚIE**: trimite numele și versiunea de sistem (MockMySensors v0.5) către sistemul gateway, solicită configurația metrologică: metrică sau imperială, prezintă către sistemul gateway facilitățile în nodul de comunicație (S\_DOOR, S\_ARMED); **TRANSMITERE/RECEPȚIONARE PERIODICĂ INFORMAȚII**: trimite nivelul bateriei (generat aleatoriu), solicită timpul de la sistemul gateway, trimite starea ușii și a sistemului de închidere.

**GatewaySerial** – este un exemplu ce permite transformarea mesajelor radio în mesaje

pe interfața serială (USB); îl vom utiliza pentru a testa comunicarea radio și pentru conectarea ulterioară la un sistem de comandă centralizat. Execuția sa (cu directiva `#define MY_DEBUG` comentată) va avea următorul efect în consola serială:

 COM3 (Arduino/Genuino Uno)

```
0;255;3;0;14;Gateway startup complete.
0;255;0;0;18;2.0.0
254;255;3;0;6;0
254;255;3;0;11;MockMySensors
254;255;3;0;12;v0.5
254;1;0;0;0;Outside Door
254;255;3;0;0;67
254;255;3;0;1;
254;1;1;0;16;0
254;1;1;0;15;1
254;255;3;0;0;67
```

Informațiile afișate în consola serială a sistemului gateway respectă următorul format de mesaj:

```
adresă nod; adresă senzor; tip mesaj; confirmare; subtip mesaj; mesaj \n
```

Astfel primul mesaj:

```
0;255;3;0;14;Gateway startup complete.
```

provine de la sistemul gateway (adresă 0), subsistem broadcast (255) – mesaj către toată rețeaua; este un mesaj de tip intern (3) subtip I\_GATEWAY\_READY (14); nu se solicită confirmare de primire(0); mesaj: Gateway startup complete.

Un mesaj provenit de la sistemul MockMySensor este de exemplu:

```
254;1;1;0;15;1
```

unde adresa sistemului este 254, senzor intern 1 – ușă; mesaj de informare asupra valorii unui sensor (1) subtip V\_ARMED (15), fără confirmare de primire (0), mesaj: 1 – adică încuietore închisă.

Pentru mai multe informații legate de formatul mesajelor puteți consulta:

<https://www.robofun.ro/forum/>

## MySensors Serial Protocol (2.0)

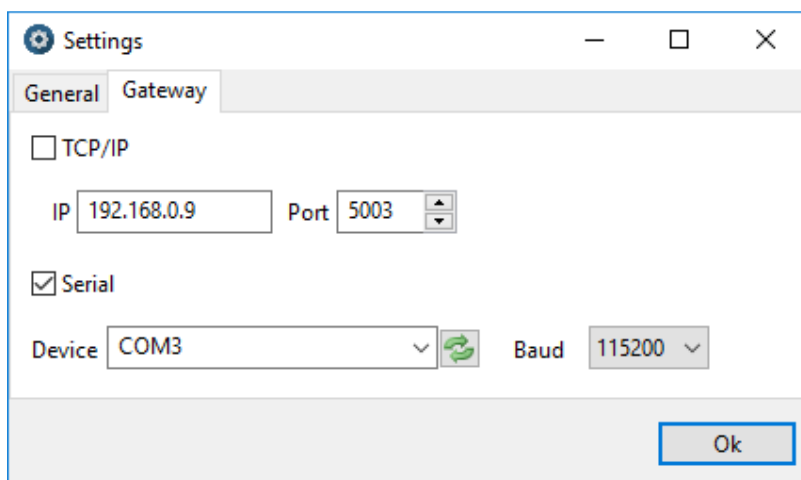
[https://www.mysensors.org/download/serial\\_api\\_20](https://www.mysensors.org/download/serial_api_20)

Bineînțeles, aceste mesaje nu sunt concepute pentru a fi citite și interpretate de utilizator. Cea mai simplă modalitate de interpretare, în lipsa unui sistem de comandă centralizată, este utilizarea software-ului MYSController:

### Windows GUI/Controller for MySensors

<https://forum.mysensors.org/topic/838/windows-gui-controller-for-mysensors/2>

Acesta este gratuit, rulează sub sisteme de operare Microsoft Windows și nu necesită instalare. Primul pas în utilizarea acestui program este identificarea legăturii cu sistemul gateway. Din meniul *Settings / Gateway* se va selecta *Serial* și numărul portului serial pe care este conectată placa Arduino ce rulează exemplul GatewaySerial.



După configurare se va da comanda de *Connect* și în interfața aplicației se vor putea observa mesajele radio primite / transmise de sistemul gateway într-o formă mult mai ușor de înțeles și urmărit. Captura de ecran din pagina următoare prezintă aceiași succesiune de mesaje ca și captura serială a sistemului gateway din pagina anterioară. Se poate observa că pe lângă zona de "decodificare" a mesajelor, interfața aplicației mai oferă și o listă a tuturor nodurilor (sistemelor) din rețea precum și o topologie a rețelei (indică dacă conexiunea este directă între nodul de achiziție și nodul gateway sau mesajul a fost retransmis prin intermediul altor noduri). Aplicația mai permite și transmiterea de mesaje către noduri de achiziție / acționare (partea de jos a interfeței) – în cazul exemplului nostru se poate selecta elementul de acționare *1 – Outside Door* și completa ca *Subtype: V\_ARMED* și ca *Payload: 0* (Unarmed). După comanda de *Send* se va putea observa o succesiune de 3 mesaje: mesajul de comandă către nodul de acționare (Mode: TX, Node: 254 – MockMySensor) și 2 mesaje ce raportează starea actuală a celor două subsisteme ale nodului (Mode: RX, Node: 254 – MockMySensor,



Subtype: V\_TRIPPED – status ușă și V\_ARMED: status încuietore):

79	7/18/2016 19:28:45	TX	254 - MockMySensors	1 - Outside Door	C_SET	NO	V_ARMED	0
80	7/18/2016 19:28:46	RX	254 - MockMySensors	1 - Outside Door	C_SET	NO	V_TRIPPED	0
81	7/18/2016 19:28:46	RX	254 - MockMySensors	1 - Outside Door	C_SET	NO	V_ARMED	0

Pe baza exemplului MockMySensors se pot experimenta și alte tipuri de senzori și elemente de acționare specifice bibliotecii MySensors. Acest lucru se poate realiza comentând în sursa exemplului cele două funcționalități implicite (S\_ARMED și S\_DOOR) și decommentând alte funcționalități (S\_MOTION, S\_SMOKE, S\_LIGHT etc.).

```
#define ID_S_ARMED 0
#define ID_S_DOOR 1
//#define ID_S_MOTION 2
//#define ID_S_SMOKE 3
//#define ID_S_LIGHT 4
...
```

În materialul următor vom prezenta construirea și programarea unui sistem personalizat de achiziție a temperaturii cu mai multe zone de măsurare compatibil cu platforma MySensors.