

# Sistem Securitate Usa

## Introducere

*Proiectul* consta in realizarea unui **Sistem Inteligent de Securitate** de tip IoT (Internet of Things), capabil sa monitorizeze in timp real accesul intr-o incapere. Sistemul utilizeaza un senzor de miscare sau de vibratie pentru a detecta intrusii, o camera video pentru verificare vizuala si un sistem de avertizare acustica (buzzer) controlabil de la distanta prin intermediul unui smartphone.

*Scopul principal* este oferirea unei solutii de supraveghere active si accesibile. Spre deosebire de sistemele de alarma clasice care doar suna, acesta permite utilizatorului sa valideze vizual amenintarea prin live stream si sa decida daca activeaza sau nu alarma sonora pentru a speria intrusul.

*Ideea a pornit de la problema alertelor false* generate de senzorii de miscare obisnuiti. Ne-am dorit un sistem care sa ofere "ochi" utilizatorului: primesti alerta, verifici video daca e cineva acolo si abia apoi actionezi buzzer-ul. Este un mod de a avea control total asupra securitatii locuintei de oriunde te-ai afla.

## Descriere generală

Sistemul este construit in jurul microcontrolerului ESP32-WROOM-32U, care gestioneaza colectarea datelor de la senzori, comunicarea wireless cu telefonul utilizatorului si stocarea evenimentelor pe card SD.

### Descrierea Modulelor si a Interactiunii

1. ESP32-WROOM-32U: Microcontrolerul principal. Citeste senzorii, gestioneaza Bluetooth-ul pentru comunicarea cu telefonul utilizatorului, controleaza buzzer-ul si scrie pe SD prin SPI. Comunica cu modulul ESP-01 prin UART pentru obtinerea pozelor de la camera.
2. Senzori (PIR HC-SR501 / MPU-6050): Detecteaza prezenta fizica si miscarea. Senzorul PIR detecteaza caldura corpului in miscare in fata usii (semnal GPIO digital), iar MPU-6050 detecteaza vibratia / deschiderea usii prin accelerometru pe I2C. Ambii sunt monitorizati de ESP32 in firmware.
3. Camera (Telefon Android cu IP Webcam): Solutia adoptata pentru captura video, in locul modulului OV7670. Telefonul ruleaza aplicatia IP Webcam si expune un stream HTTP MJPEG / endpoint /photo.jpg pe reseaua WiFi locala.
4. Modul WiFi ESP-01 (ESP8266): Plăcuta separata dedicata WiFi-ului. ESP32 o controleaza prin comenzi AT trimise pe UART2. ESP-01 face HTTP GET catre camera IP si returneaza JPEG-ul catre ESP32 pe UART, octet cu octet.
5. Buzzer: Elementul de descurajare. Activat exclusiv la comanda explicita a utilizatorului (tasta "B" trimisa prin Bluetooth), oprit cu "S" sau automat dupa 30 secunde (safety timeout). Comanda prin pin GPIO.
6. Cititor Card SD: Stocheaza local un log al evenimentelor (timestamp + tip eveniment + detalii) si

pozele JPEG salvate la fiecare detectie. Conectat prin SPI.

- AMS1117: Regulator de tensiune 3.3V folosit pentru alimentarea stabila a ESP-01 (consum de varf 300 mA in transmisie WiFi care ar putea depasi capacitatea regulatorului intern al placutei ESP32).

**Decizia tehnica: inlocuirea camerei OV7670** Modulul OV7670 din BOM-ul initial nu a putut fi integrat functional cu ESP32-WROOM-32U fara PSRAM. Senzorul transmite pixelii prin interfata paralela in timp real (PCLK la 10-20 MHz), iar fara un buffer FIFO extern (AL422), ESP32 trebuie sa-i capteze direct prin DMA I2S - operatie sensibila la timing. In testele efectuate, frame-urile au sosit constant incomplete (30720 octeti din 38400 asteptati - aproximativ 96 din 120 randuri), indicand o problema fundamentala de sincronizare VSYNC/HREF in absenta unei memorii tampon dedicate.

Solutiile teoretice ar fi fost: (a) modul OV7670 cu cip AL422B FIFO, (b) modul ArduCAM Mini cu OV2640 si FIFO integrat, sau © placuta ESP32-CAM. Variantele (b) si © nu se potriveau cerintei de a folosi componente separate; varianta (a) presupunea o livrare de 2-3 saptamani din afara tarii.

Solutia adoptata - folosirea unui telefon Android second-hand ca sursa video - ofera avantaje practice semnificative: rezolutie superioara (HD/FullHD vs 640x480 OV7670), stream video continuu (nu doar cadre izolate), audio integrat, alimentare prin propria baterie, si mai ales zero probleme de timing. Telefonul comunica cu sistemul prin protocol HTTP standard, ceea ce permite si integrarea facila cu modulul ESP-01.

**Decizia tehnica: introducerea modulului ESP-01** La integrarea WiFi pe ESP32 alaturi de Bluetooth Classic (necesar pentru comunicarea cu telefonul utilizatorului) a aparut o limitare hardware: stiva Bluetooth Classic ocupa circa 110 KB RAM, iar stiva WiFi + lwIP suplimentar 60-70 KB. Cu un total de 320 KB RAM disponibil in ESP32-WROOM-32U (fara PSRAM), suma necesarului depaseste capacitatea, iar initializarea WiFi esueaza ("Failed to allocate ... bytes").

Solutia adoptata distribuie sarcina pe doua microcontrolere care comunica prin UART: ESP32 ramane dedicat senzorial + Bluetooth, iar ESP-01 (ESP8266) preia complet stiva WiFi/HTTP. ESP32 trimite comenzi AT pe UART2 catre ESP-01, care face cererea HTTP catre telefonul-camera si returneaza JPEG-ul octet cu octet. ESP32 scrie streaming pe SD, fara sa pastreze intregul JPEG in RAM.

### Modul de functionare pe scurt

- La pornire, ESP32 initializeaza senzorii, SD-ul, Bluetooth-ul si stabileste conexiunea cu ESP-01 prin UART; ESP-01 se autentifica pe reseaua WiFi locala.
- Senzorul PIR sau MPU-6050 detecteaza miscare la usa.
- ESP32 trimite o alerta catre telefonul utilizatorului prin Bluetooth, impreuna cu URL-ul stream-ului video.
- In paralel, ESP32 instruieste ESP-01 sa descarce o poza instantanee de la camera IP; poza e salvata pe SD ca /snap\_NNN.jpg.
- Utilizatorul deschide URL-ul primit si vede stream-ul live din fata usii.
- Daca identifica un intrus, apasa "B" in aplicatia Bluetooth Terminal de pe telefon, iar ESP32 activeaza buzzer-ul ca metoda de descurajare. Trimite "S" pentru oprire, sau buzzer-ul se opreste automat dupa 30 secunde.
- Toate evenimentele (detectii, comenzi, schimbari de stare) sunt logate pe SD cu data si ora exacta, pentru audit ulterior.

### Diagrama bloc inainte de schimbarea modulului OV7670



## Diagrama bloc dupa schimbarea modului OV7670



## Hardware Design

Aici puneți tot ce ține de hardware design:

- listă de piese
- scheme electrice (se pot lua și de pe Internet și din datasheet-uri, e.g. <http://www.captain.at/electronic-atmega16-mmc-schematic.png>)
- diagrame de semnal
- rezultatele simulării

## Hardware

### Lista componente folosite:

1. ESP32-WROOM-32U (microcontroller principal)
  - Dual-core Xtensa LX6 @ 240 MHz, 320 KB RAM, 4 MB Flash
  - Bluetooth Classic + BLE, WiFi 802.11 b/g/n (conector IPEX pentru antena externa)
  - 34 GPIO programabile, ADC/DAC, I2C, SPI, UART, PWM (LEDC)Datasheet:

[https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32\\_datasheet\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32_datasheet_en.pdf)

2. PIR HC-SR501 (senzor miscare prin infrarosu pasiv)
  - Tensiune: 5V DC, consum < 65 mA
  - Distanta detectie: 3-7 m (reglabila)
  - Timp HIGH la detectie: 5-200 s (reglabil)
  - Output: digital 3.3V compatibil cu GPIO ESP32Datasheet:  
<https://www.mpja.com/download/31227sc.pdf>  
Referinta tehnica:  
<https://components101.com/sensors/hc-sr501-pir-sensor>

3. MPU-6050 (modul GY-521, accelerometru + giroscop)
  - 3 axe accelerometru (+/- 2g, 4g, 8g, 16g)
  - 3 axe giroscop (+/- 250, 500, 1000, 2000 dps)

- Interfata I2C, adresa 0x68 (sau 0x69 daca AD0 = HIGH)
  - Tensiune: 3.3V - 5V (regulator on-board)
  - Senzor temperatura integrat
- Datasheet senzor:

<https://invensense.tdk.com/wp-content/uploads/2015/02/MPU-6000-Datasheet1.pdf>

Register map (referinta pentru firmware):

<https://invensense.tdk.com/wp-content/uploads/2015/02/MPU-6000-Register-Map1.pdf>

4. ESP8266 ESP-01 (modul WiFi separat)
- SoC ESP8266EX cu Tensilica L106 @ 80 MHz
  - 802.11 b/g/n, TCP/IP integrat, securitate WPA/WPA2
  - 1 MB Flash, firmware AT pentru control prin UART
  - Tensiune: 3.3V, consum varf ~300 mA
  - UART la 115200 bps (default)
- Datasheet ESP8266EX:

[https://www.espressif.com/sites/default/files/documentation/0a-esp8266ex\\_datasheet\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf)

Documentatie AT commands:

[https://www.espressif.com/sites/default/files/documentation/4a-esp8266\\_at\\_instruction\\_set\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/4a-esp8266_at_instruction_set_en.pdf)

5. AMS1117-3.3 (regulator de tensiune LDO 3.3V)
- Intrare: 4.3V - 12V
  - Iesire: 3.3V fix
  - Curent maxim: 1 A
  - Drop-out tipic: 1.2V
  - Capacitor decuplare obligatoriu pe iesire ( $\geq 10 \mu\text{F}$ )
- Datasheet:
- <https://www.advanced-monolithic.com/pdf/ds1117.pdf>

6. Cititor microSD SPI (Elektroweb 3.3V)
- Interfata SPI (MISO, MOSI, SCK, CS)
  - Suport carduri microSD pana la 32 GB (FAT32) - in practica merge si cu 128 GB
  - Tensiune: 3.3V (regulator LDO pe modul)
  - Convertor de nivel logic 5V -> 3.3V integrat
- Datasheet specificatie SD over SPI:

[https://www.sdcard.org/downloads/pls/pdf/index.php?p=Part1\\_Physical\\_Layer\\_Simplified\\_Specification\\_Ver8.00.jpg&f=Part1\\_Physical\\_Layer\\_Simplified\\_Specification\\_Ver8.00.pdf&e=EN\\_SS1\\_8](https://www.sdcard.org/downloads/pls/pdf/index.php?p=Part1_Physical_Layer_Simplified_Specification_Ver8.00.jpg&f=Part1_Physical_Layer_Simplified_Specification_Ver8.00.pdf&e=EN_SS1_8)

7. Buzzer activ 3-24V (generator de ton intermitent intern)

- Activare prin tensiune continua aplicata pe pini
  - Ton intermitent generat de oscilator intern (~85 dB la 30 cm)
  - Necesita tranzistor de comanda (BC547 / 2N2222) pentru alimentare de pe GPIO ESP32 - curentul depaseste capacitatea pinilor (40 mA max)
- Specificatii model:  
<https://www.emag.ro/buzzer-cu-generator-de-ton-intermitent-3-24v-75db-...>

#### 8. Baterie 9V (alcalina standard)

- Sursa primara de alimentare a sistemului
- Conectata prin AMS1117 -> 3.3V pentru ESP-01 si periferice

### Componente din BOM-ul initial neutilizate in versiunea finala:

1. Camera OV7670 (CMOS VGA 640x480, fara FIFO)

Inlocuita cu telefon Android + IP Webcam din motivele explicate in

descrierea generala (timing inadecvat pe ESP32-WROOM fara PSRAM).  
Datasheet OV7670 (pentru referinta):  
<https://www.voti.nl/docs/OV7670.pdf>

### Telefon Android second-hand (camera IP)

1. Functie: server video HTTP cu stream MJPEG si endpoint /photo.jpg
2. App: IP Webcam (Pavel Khlebovich, gratuit pe Google Play)
3. Comunica cu ESP-01 prin WiFi 2.4 GHz / HTTP
4. Necesita o retea WiFi locala comuna cu modulul ESP-01

## Software Design

Descrierea codului aplicatiei (firmware):


- mediu de dezvoltare (if any) (e.g. AVR Studio, CodeVisionAVR)
- librării și surse 3rd-party (e.g. Procyon AVRlib)
- algoritmi și structuri pe care plănuieți să le implementați
- (etapa 3) surse și funcții implementate

## Rezultate Obținute

Care au fost rezultatele obținute în urma realizării proiectului vostru.

## Concluzii

## Download

O arhivă (sau mai multe dacă este cazul) cu fișierele obținute în urma realizării proiectului: surse, scheme, etc. Un fișier README, un ChangeLog, un script de compilare și copiere automată pe uC crează întotdeauna o impresie bună .

Fișierele se încarcă pe wiki folosind facilitatea **Add Images or other files**. Namespace-ul în care se încarcă fișierele este de tipul **:pm:prj20??:c?** sau **:pm:prj20??:c?:nume\_student** (dacă este cazul). **Exemplu:** Dumitru Alin, 331CC → **:pm:prj2009:cc:dumitru\_alin**.

## Jurnal

Puteți avea și o secțiune de jurnal în care să poată urmări asistentul de proiect progresul proiectului.

## Bibliografie/Resurse

Listă cu documente, datasheet-uri, resurse Internet folosite, eventual grupate pe **Resurse Software** și **Resurse Hardware**.

[Export to PDF](#)

From:

<http://ocw.cs.pub.ro/courses/> - **CS Open CourseWare**

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2026/bianca.popa1106/victoria.plamadeala> 

Last update: **2026/05/18 16:20**