

# Automatizacija sustava korištenjem ESP32 mikrokontrolera

Branimir Ričko | [branimir.ricko@fer.hr](mailto:branimir.ricko@fer.hr) | Fakultet elektrotehnike i računarstva  
| Automatizacija, ESP32, Mikrokontroleri

# Sadržaj

1. Uvod	1
2. Automatizacija	1
2.1. Definicija	1
2.2. Zašto	1
2.3. Kako	1
2.4. Prednosti	1
2.5. Nedostaci	1
3. Mikrokontroleri	2
3.1. Definicija	2
3.2. Zašto	2
3.3. Kako	2
3.4. Prednosti	2
3.5. Nedostaci	2
3.6. ESP32	2
3.6.1. Zašto	3
3.6.2. Kako	3
3.6.3. Usporedba s konkurencijom	3
3.6.4. Specifičnosti	3
3.6.5. UI	3
3.6.6. Potrošnja	4
3.6.7. API	4
4. Realizacija sustava	4
4.1. Zahtjevi	4
4.2. Plan	4
4.3. Razrada	4
4.4. Usporedba s konkurencijom	4
5. Zaključak	4
6. Literatura	5
7. TL;DR	5

# 1. Uvod

U ovom seminaru ćemo proučavati što je sve potrebno za uspješnu automatizaciju nekog abstraknog sustava korištenjem ESP32 mikrokontrolera, a na kraju ćemo uzet neki realan sustav kao primjer. Postojanje ESP32 mikrokontrolera dosta olakšava automatizaciju bolko kojeg sustava, zbog lakog programiranja, velik broj IO-a. Automatizacija sustava omogućuje lakše i jeftine upravljanje istim sustavom.

## 2. Automatizacija

### 2.1. Definicija

Automatizacija označava tijek prijenosa rada čovjeka na strojeve, obično kroz tehnički napredak. U industrijalizaciji nastavak je mehanizacije.

Npr. Neki sustav koji zahtijeva brojenje nekih predmeta na nekoj mehaniziranoj traci.

### 2.2. Zašto

Sustav je dobro mehanizirati zbog nekoliko glavnih razloga:

- Otklananje ljudskog faktora
- Lakša kompozicija sustava
- Veća preciznost dokumentacije sustava

### 2.3. Kako

Sustav prije nego je automatiziran, mora biti mehaniziran. Mehanizacija sustava podrazumijeva mijenjanje ljudske radne snage strojevima. Nakon mehanizacije, još uvijek mora postojati ljudski faktor koji upravlja strojevima. Automatizacijom otklanjamo i taj ljudski faktor tako što logiku cijelog sustava enkapsuliramo i prilagodimo za izvođenje na nekom mikrokontroleru.

### 2.4. Prednosti

- Otklananje ljudskog faktora
- Lakša kompozicija sustava
- Veća preciznost dokumentacije sustava

### 2.5. Nedostaci

- Ljudi koji rade na sustavu prije automatizacije gube posao
- Velika inicijalna cijena
- Ovisnost o firmi koja je automatizirala sustav

## 3. Mikrokontroleri

### 3.1. Definicija

Mikrokontroler je skup integriarnih krugova koji su ukomponirani na takav način da mogu funkcionirati kao mala računala, najčešće se programira kako bi izvršavao neku specifičnu zadaću.

### 3.2. Zašto

Mikrokontroleri se koriste kad je potrebno izvršavati neku specifičnu zadaću koja je dovoljno kompleksna da je izrada integriranog sklopa pre komplicirana, a opet nije toliko kompleksna da je potrebno računalo.

### 3.3. Kako

### 3.4. Prednosti

- Niska cijena
- Male dimenzije
- Niska potrošnja (1W-2W)
- Relativno velika procesna moć

### 3.5. Nedostaci

- Ponekad je API napisan na nelogičan način
- Relativno mala procesna moć u usporedbi s računalom
- Relativno velika potrošnja ukoliko ne koristimo svi procesnu moć

### 3.6. ESP32



Slika 1. ESP32

ESP32 je mikrokontroler sa integriranim Wi-Fi i bluetooth posdsustavom.

### 3.6.1. Zašto

ESP32 je

### 3.6.2. Kako

### 3.6.3. Usporedba s konkurencijom

	ESP32	ESP8266	Arduino
Napon napajanja	3.3v	3.3v	5.0v
Struja napajanja	5 $\mu$ A - 200 mA	100mA	100mA
IO	I <sup>2</sup> C, Spi, Wi-fi, Bluetooth, Ethernet, GPIO, PWM, UART, Senzor za temperaturu	I <sup>2</sup> C, Spi, Wi-fi, GPIO, PWM, UART	I <sup>2</sup> C, Spi, GPIO, PWM, UART
Procesor	Xtensa dual-core 32-bit LX6 microprocessor (240 MHz)	Xtensa Diamond Standard 106Micro (80 MHz)	Atmel 8-bit AVR (16 MHz)
Memorija	520 KiB SRAM	128Kib	2KiB - 8Kib
Flash	16 MiB	16 MiB	256KiB
SDK	Espressif IoT Development Framework, Espruino, Lua RTOS for ESP32, Mongoose OS, mruby, MicroPython	NodeMCU, ESP-Open-SDK, ESP-Open-RTOS, Mongoose OS, MicroPython	Arduino, OpenWRT

### 3.6.4. Specifičnosti

- Wi-Fi
- Bluetooth
- Ultra low power mode (15  $\mu$ W)
- Hardverski implemetirano AES, SHA-2, RSA, ECC, RNG
- Temperature sensor
- Hall efeket senzor
- Analogno predpojačalo

### 3.6.5. UI

- 12-bit SAR ADC
- 2  $\times$  8-bit DACs
- 10  $\times$  GPIOs

- 4 × SPI
- 2 × I<sup>2</sup>S
- 2 × I<sup>2</sup>C
- 3 × UART
- SD/SDIO/CE-ATA/MMC/eMMC
- SDIO/SPI
- Ethernet
- CAN bus 2.0
- PWM

### 3.6.6. Potrošnja

Wi-Fi slanje	240 mA
Wi-Fi primanje	100 mA
240Mhz	30 mA
2MHz	2 mA
Ultra low power	5 µA - 150 µA

### 3.6.7. API

Postoji nekoliko načina programiranja za ESP32. Originalni API za glavni procesor je napisan u c-u, dok je api za ULP supprocesor napisan u assembleru.

Postoje i implementacije api-a u višim programskim jezicima. Te implementacije mogu biti dosta sporije, no jako olakšavaju programiranje.

## 4. Realizacija sustava

### 4.1. Zahtijevi

### 4.2. Plan

### 4.3. Razrada

### 4.4. Usporedba s konkurencijom

## 5. Zaključak

## 6. Literatura

- [Sva dokumentacija](#)

## 7. TL;DR