

# PERIFERIJE I INTERFEJSI

# O PREDMETU

***Studijski programi za koje se organizuje :***

-Specijalističke studije Elektrotehnike, studijski program Elektronika, Telekomunikacije i Računari (ETR), smjer Elektronika.

***Uslovljenost drugim predmetima:*** Nema formalnih uslova. Podrazumijeva se poznavanje C/C++ jezika.

***Ciljevi izučavanja predmeta:*** Interfejsi (međusklopovi) i periferije (spoljni uređaji) povezuju računare sa spoljašnjom sredinom. Cilj ovog predmeta je da studenti ovladaju znanjem i tehnikama, pomoću kojih će moći da uz pomoć računara (automatski) prikupljaju informacije iz spoljašnjeg svijeta i da upravljaju procesima van računara. Osim teoretskog dijela, značajna pažnja se poklanja praktičnom radu.

***Metod nastave i savladanja gradiva:*** Predavanja, računске vježbe i vježbe u računarskoj učionici / laboratoriji. Učenje i samostalna izrada praktičnih zadataka. Konsultacije.

# O PREDMETU

## Sadržaj predmeta:

Pripremna sedmica	Priprema i upis semestra
I sedmica	Uvod; Upoznavanje sa predmetom, ciljevima i načinom rada;
II sedmica	Mikrokontroleri; Razvojne ploče; Arduino Uno;
III sedmica	Izlazni portovi (Sedmosegmentni i LCD displeji);
IV sedmica	Ulazni portovi;
V sedmica	Analogni ulazi i izlazi;
VI sedmica	Koračni motori;
VII sedmica	Optički interfejsi;
VIII sedmica	<i>Provjera znanja</i>
IX sedmica	Servo motori;
X sedmica	Daljinski upravljači;
XI sedmica	Senzori;
XII sedmica	Komunikacioni interfejsi: -paralelni, -serijski; SPI, TWI (I <sup>2</sup> C), UART, ZigBee ,...
XIII sedmica	Wi-Fi, Eternet, povezivanje na internet;
XIV sedmica	Modemi: AT komande, gsm, gprs, LoRa;
XV sedmica	Interfejsi u industriji; Vizuelizacija industrijskih procesa;
XVI sedmica	<i>Završni ispit</i>
Završna sedmica	Ovjera semestra i upis ocjena
XVIII-XXI sedmica	Dopunska nastava i popravni ispitni rok

# O PREDMETU

## ***Opterećenje studenata na predmetu***

### **Sedmično**

**6 kredita x 40/30 = 8 časova**

### **Struktura:**

**3 časa predavanja**

**1 čas računskih vježbi**

**4 časa samostalnog rada,  
uključujući konsultacije**

### **U toku semestra**

**Nastava i završni ispit: (8 časova) x 16 = 128  
časova**

**Neophodne pripreme prije početka semestra  
(administracija, upis, ovjera)**

**2 x (8 časova) = 16 časova**

**Ukupno opterećenje za predmet 6x30 = 180  
časova**

**Dopunski rad** za pripremu ispita u popravnom  
ispitnom roku, uključujući i polaganje popravnog ispita  
**od 0 do 30 časova** (preostalo vrijeme od prve dvije  
stavke do ukupnog opterećenja za predmet 150  
časova)

### **Struktura opterećenja:**

**128 časova (Nastava)+16 časova (Priprema)+36  
časova (Dopunski rad)**

Studenti su obavezni da pohađaju nastavu, rade i predaju sve domaće zadatke, odrade laboratorijske vježbe i obje provjere znanja.

# O PREDMETU

## **Literatura:**

Osnovna i pomoćna literatura u elektronskom obliku na [www.ucg.ac.me/etf](http://www.ucg.ac.me/etf)

Praktični zadaci za laboratorijske vježbe na [www.ucg.ac.me/etf](http://www.ucg.ac.me/etf)

Z.Mijanović i ostali, »Računarski interfejsi i periferije«, Univerzitet Crne Gore

## **Oblici provjere znanja i ocjenjivanje:**

-Bodovi na vježbama 26 poena

-Provjera znanja 24 poena

-Završni ispit 50 poena.

-Prelazna ocjena se dobija ako se kumulativno sakupi najmanje 50 poena.

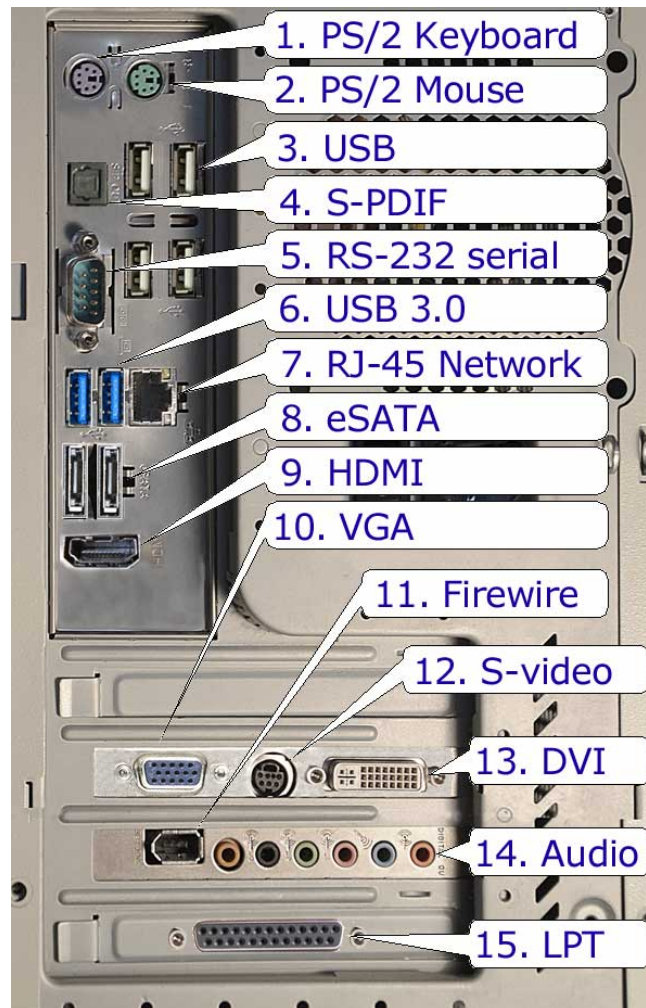
# PERIFERIJE I INTERFEJSI

Interfejsi (međusklopovi) i periferije (spoljni uređaji) povezuju računare sa spoljašnjom sredinom.

Cilj ovog predmeta je da studenti ovladaju znanjem i tehnikama, pomoću kojih će moći:

- da uz pomoć računara (automatski) prikupljaju informacije iz spoljašnjeg svijeta,
- da upravljaju procesima van računara.

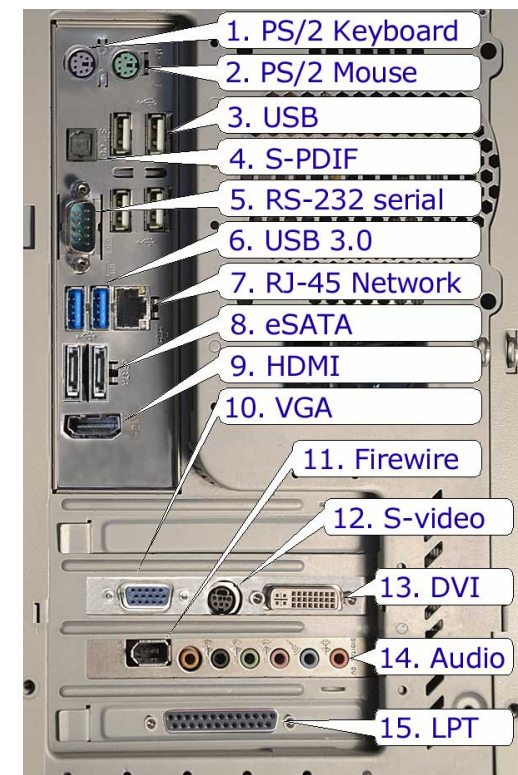
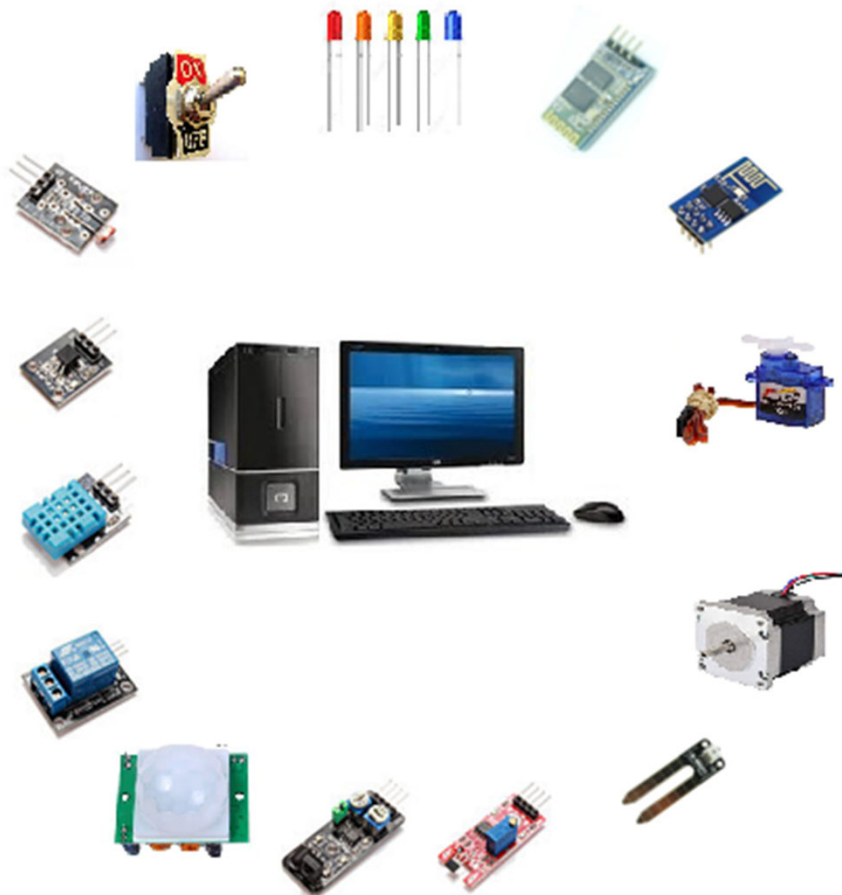
# Računarski portovi





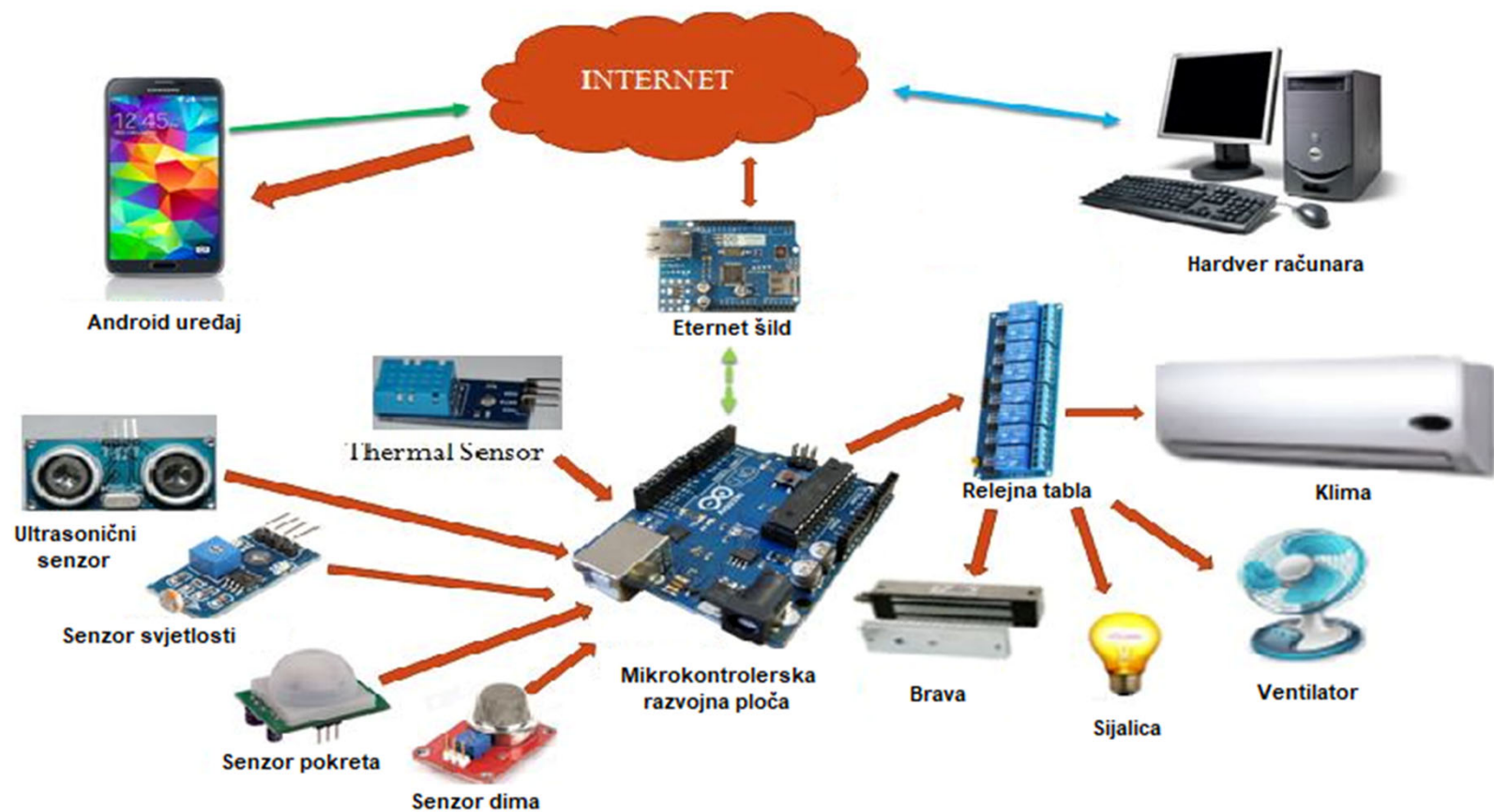


# Kako povezati senzore i aktuatore sa računarom i korisnikom?

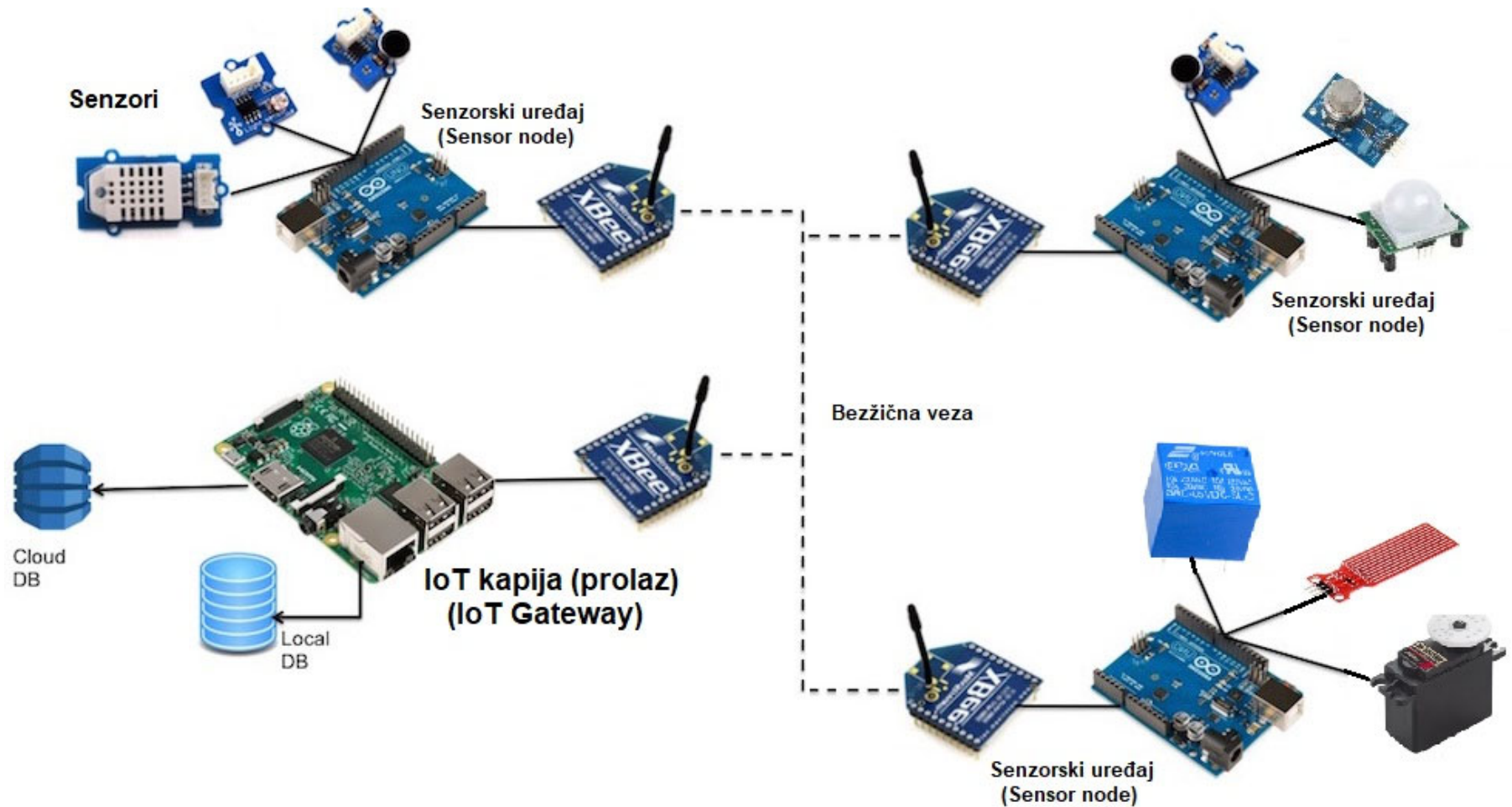




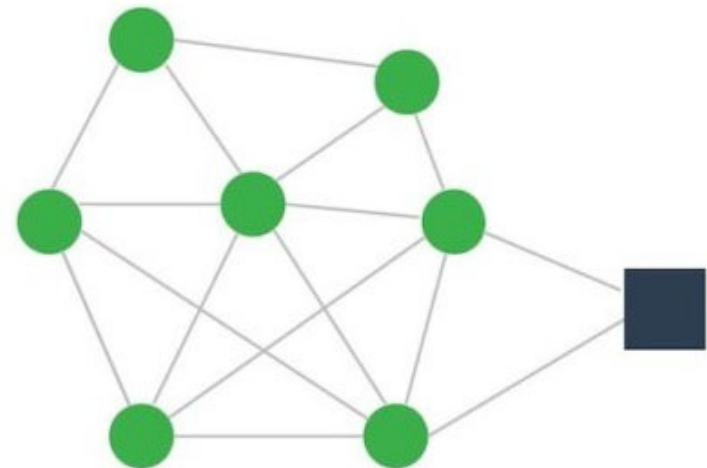
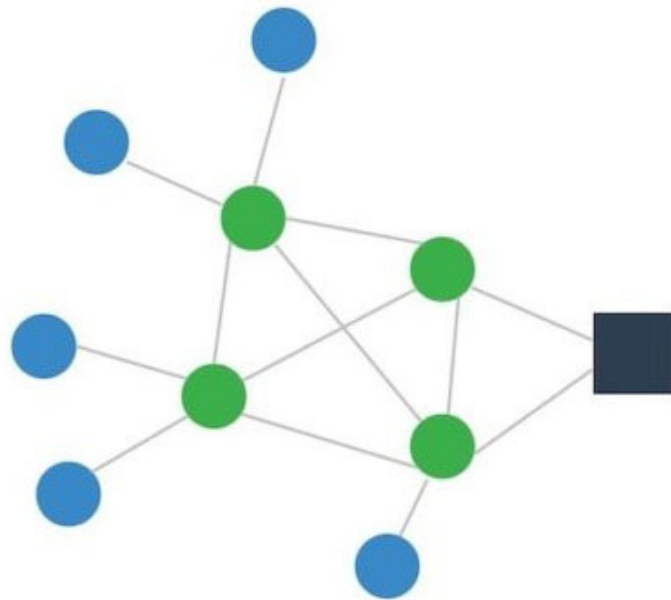
# Povezivanje Arduino uređaja u sistem



# Ne mora svaki Arduino uređaj direktno na internet.



# Različite vrste senzorskih mreža



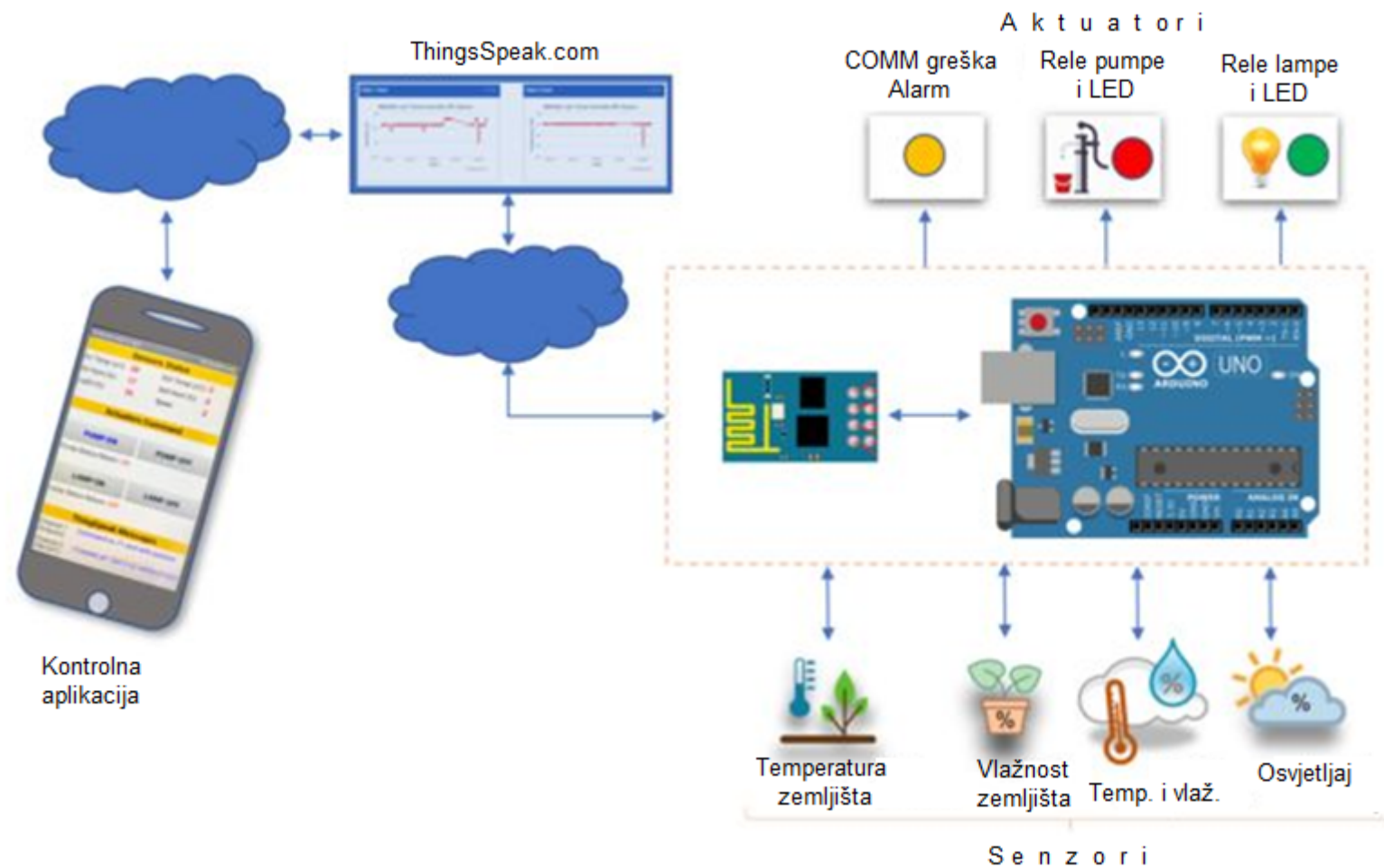
● Senzorski uređaj

● Senzorski uređaj sa funkcijom prenosa / ruter

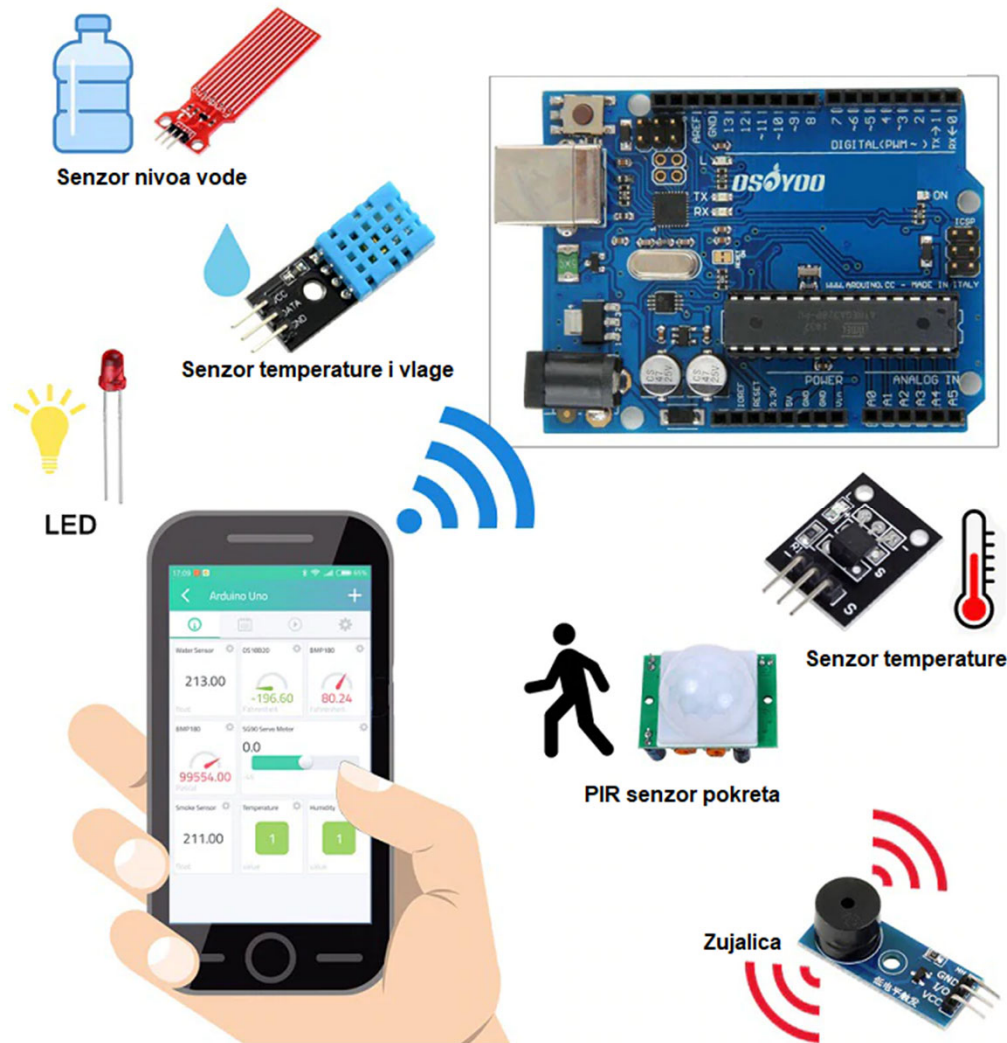
■ Gateway



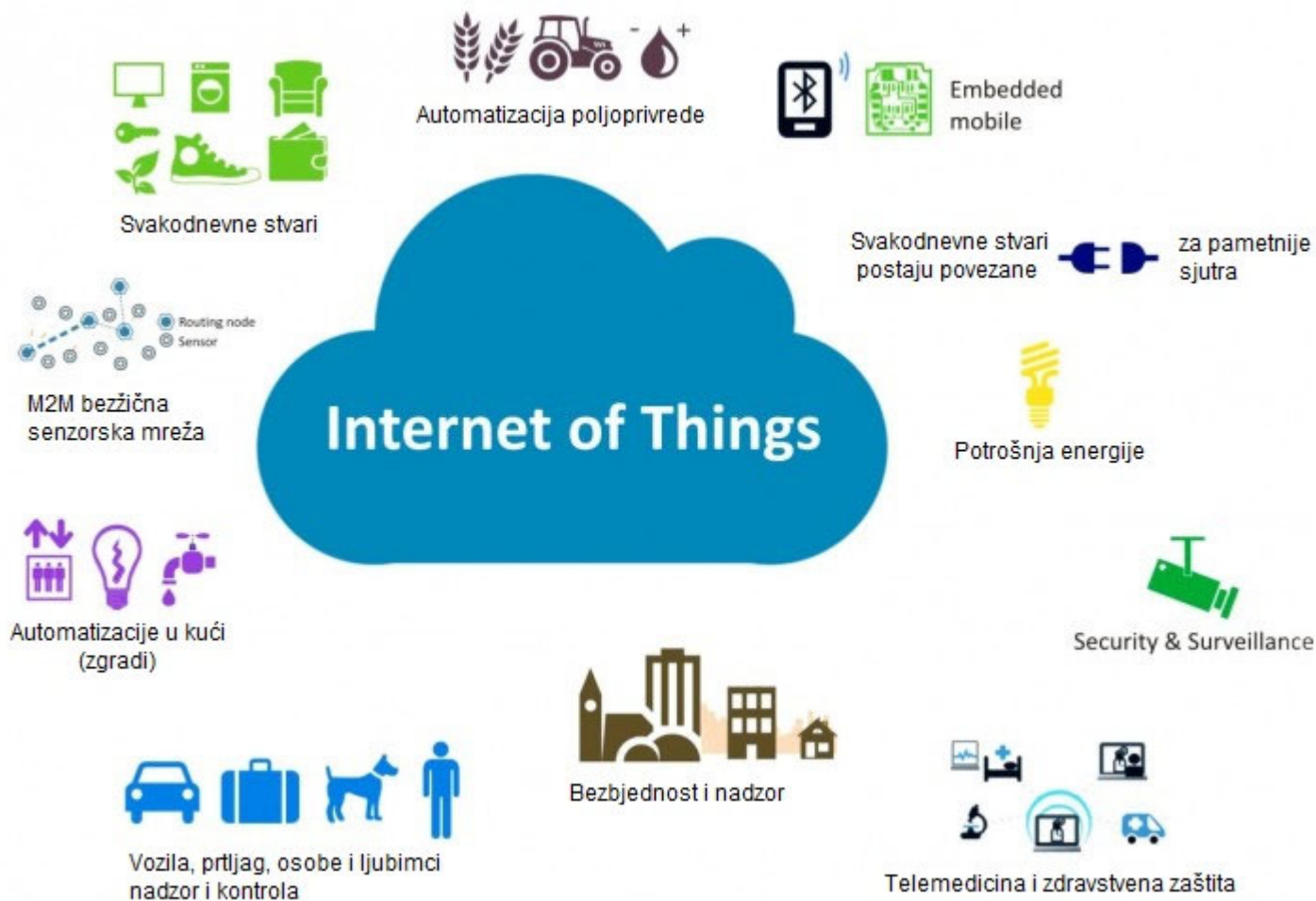
# Povezivanje u oblak



# IoT projekti

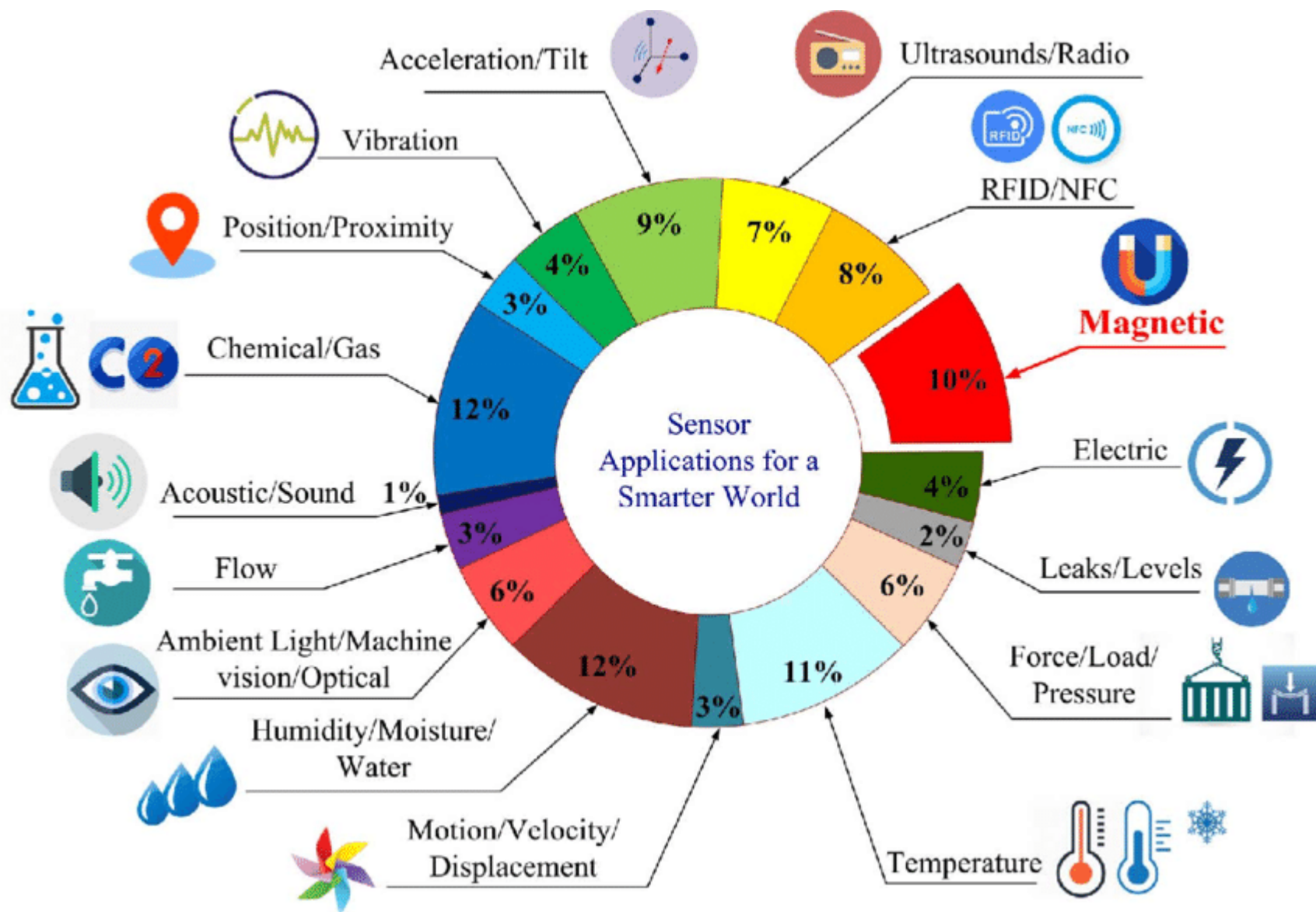


# IoT oblasti primjene

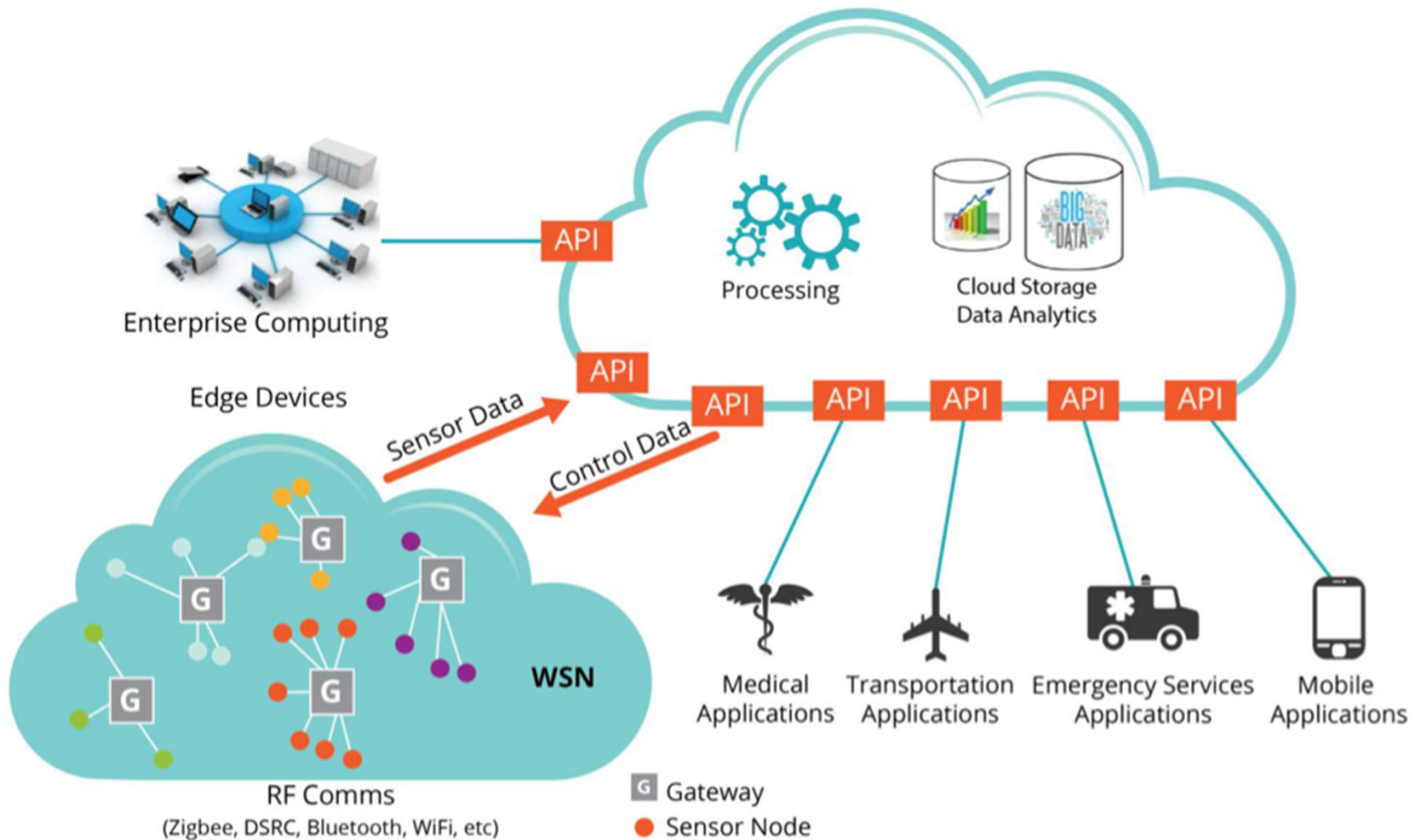




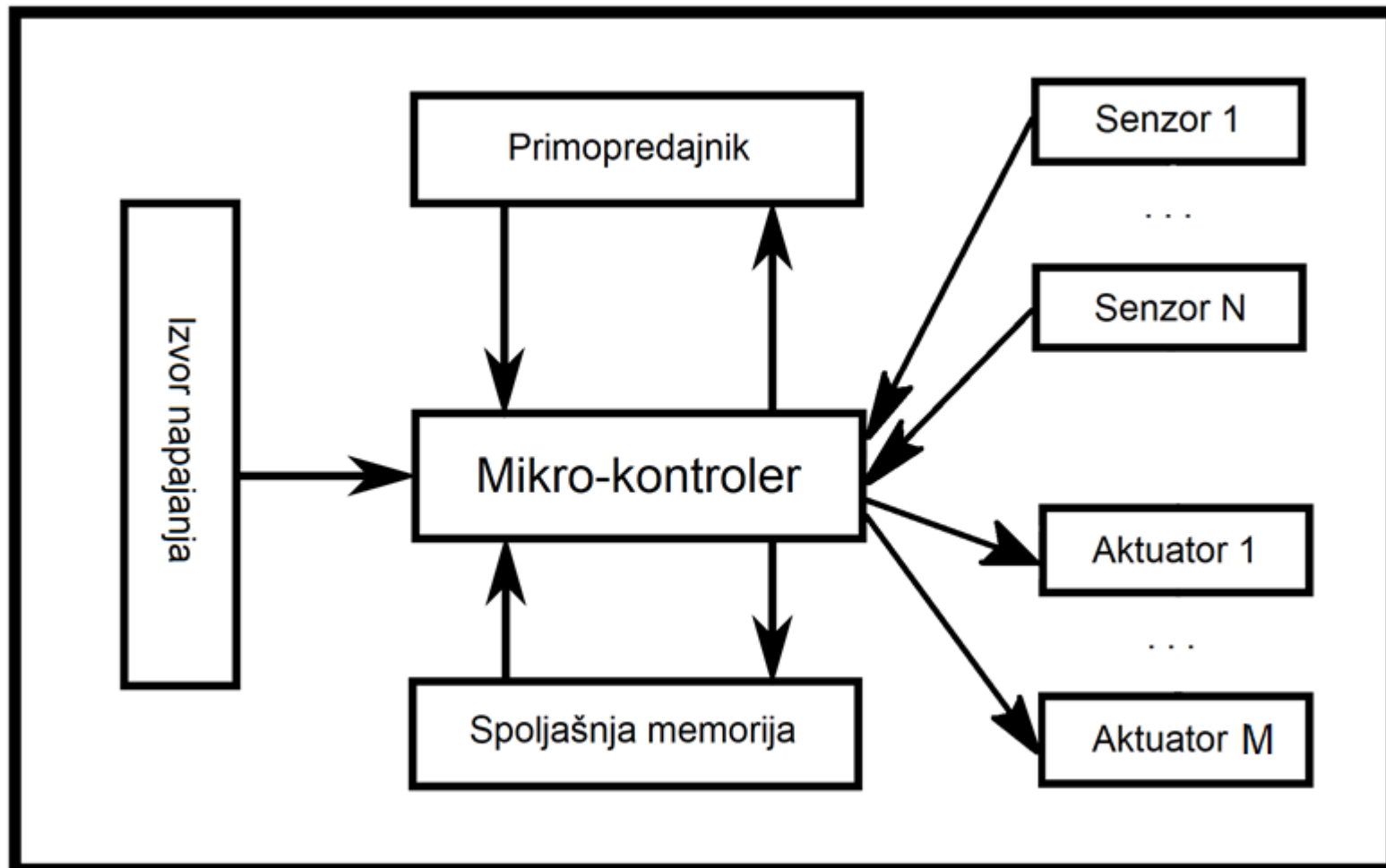
# Senzorske aplikacije za pamatniji svijet



# IoT infrastruktura



# Mikrokontrolerski uređaj (Senzorski čvor – Sensor node, Mote)



# Što je mikrokontroler?

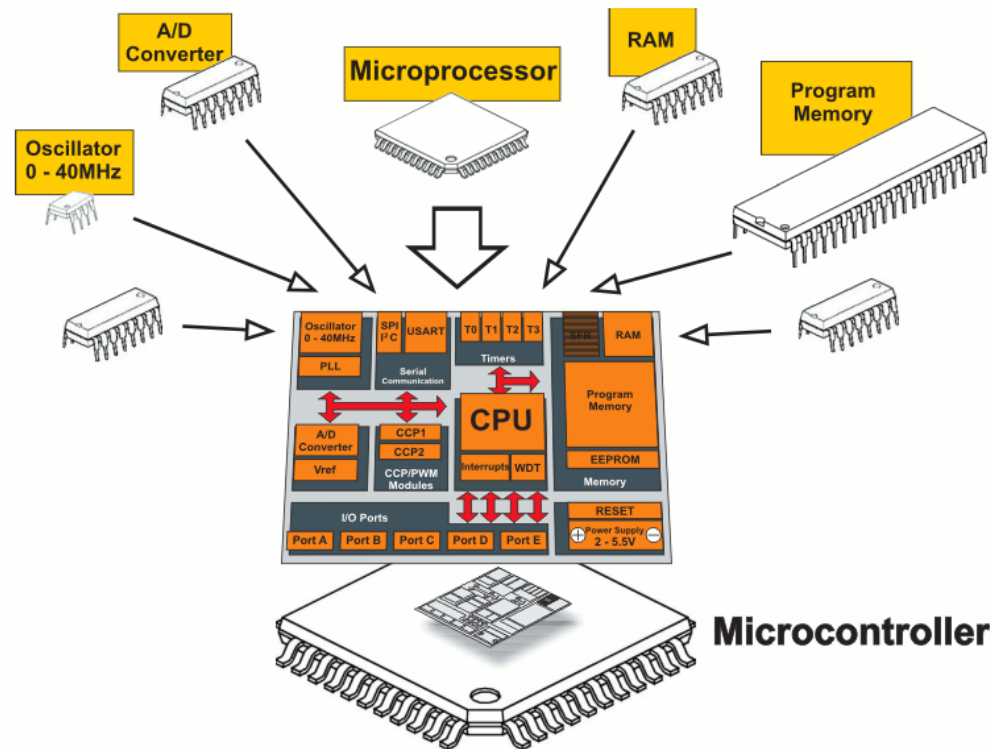
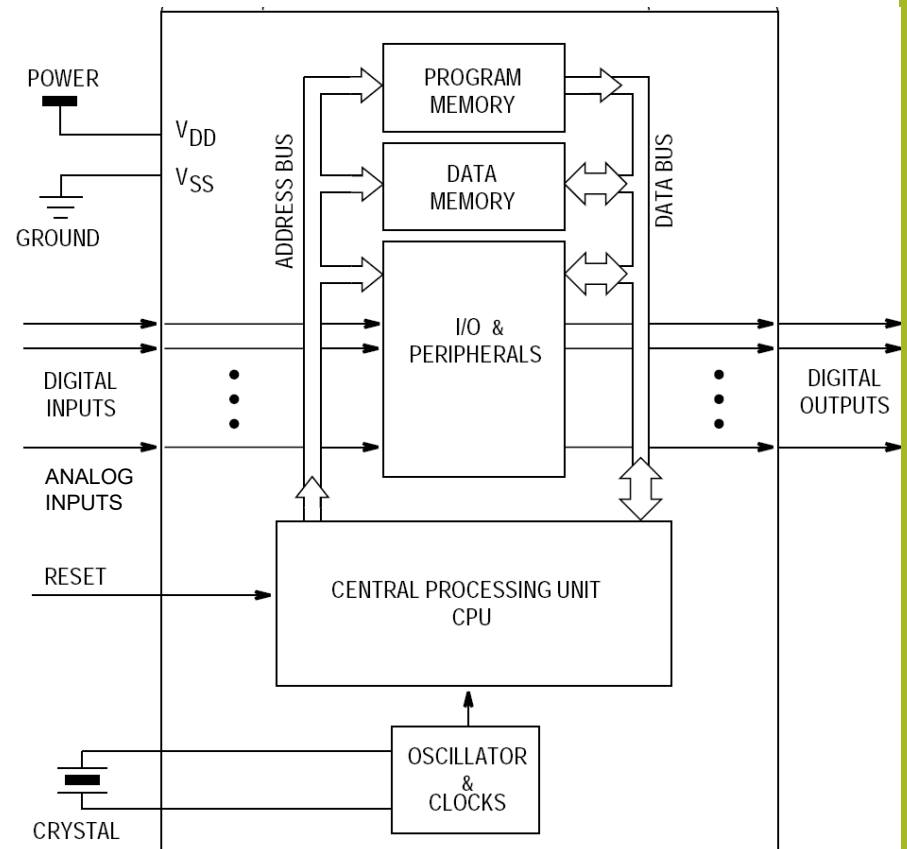
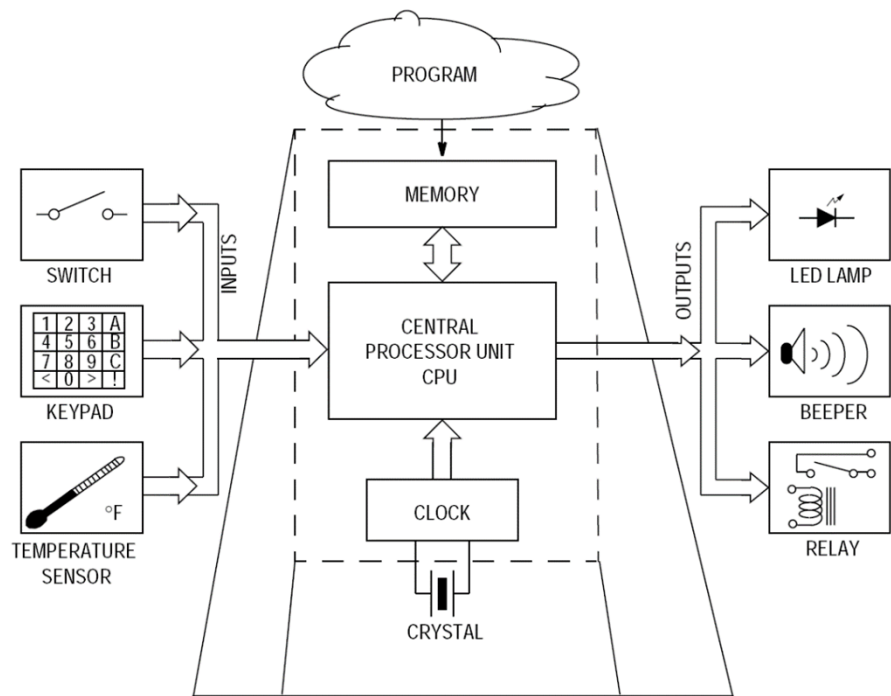


Fig. 0-1 Microcontroller versus Microprocessor

- Mali kompjuter u jednom čipu  
Sadrži procesor, memoriju, i ulaze/izlaze
- Tipično je „**ugrađen**“ unutar nekih uređaja i kontroliše njihov rad
- Mikrokontroler je često mali i jeftin.

# Što je mikrokontroler?

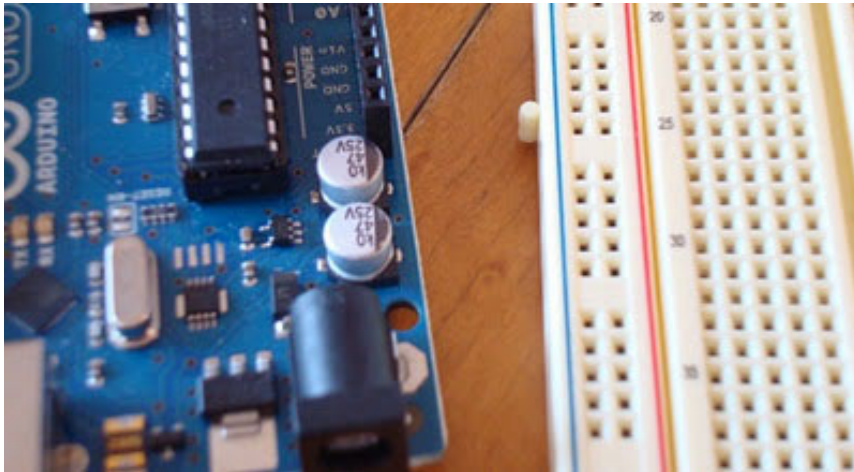


Što je razlika između 'Digital Input' i 'Analog Input'?

# Mikrokontroleri – definicija

- Programeri rade u virtuelnom svijetu.
- Uređaji rade u fizičkom svijetu.
- Kako povezati vituelni i fizički svijet?
- Uvedite mikrokontroler.
- Mikrokontroler je u osnovi mali računar koji posjeduje programabilne ulaze i izlaze opšte namjene.
- Ulazi mogu biti upravljani od strane fizičkog okruženja dok izlazi mogu upravljati fizičkim okruženjem.

# Što je razvojna ploča?



- Štampana matična ploča dizajnirana da olakša rad sa mikrokontrolerom
- Razvojna ploča tipično uključuje:
  - napojno kolo;
  - programerski interfejs;
  - Lako dostupne ulazno/izlazne pinove.

# Arduino – Zvanična definicija

- Uzeto sa zvaničnog web sajta (arduino.cc):

Arduino je open-source elektronska prototipna platforma zasnovana na fleksibilnom, jednostavnom za upotrebu, hardveru i softveru.

Namijenjen je dizajnerima, hobistima, i svima drugima koji su zainteresovani za kreiranje interaktivnih objekata i okruženja.

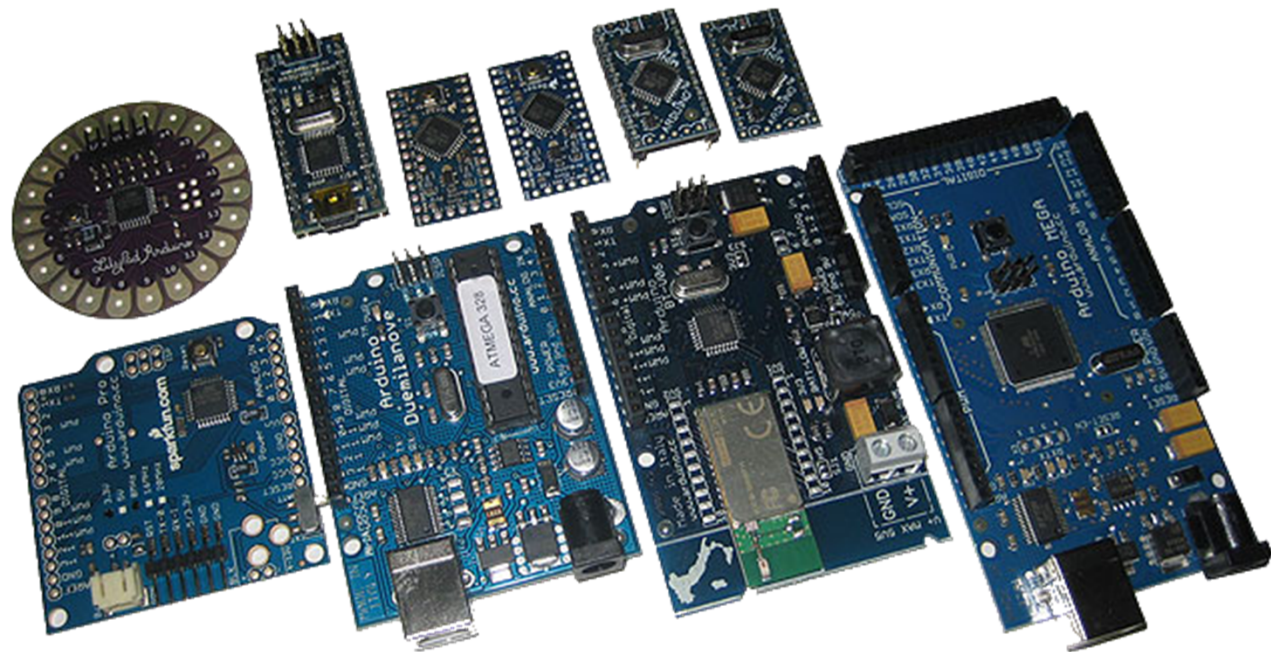


# Zašto Arduino?

- Bez obzira na razlog, Arduino platforma je postala defakto standard.
  - Postoji puno realizovanih, dostupnih, projekta koji koriste arduino platformu.
- Teži ravnoteži između jednostavnosti upotrebe i korisnosti.
  - Programski jezici se uglavnom vide kao glavna poteškoća.
  - Arduino C je značajno uproštena verzija C++.
- Nije skup.

# Tipovi Arduino-a

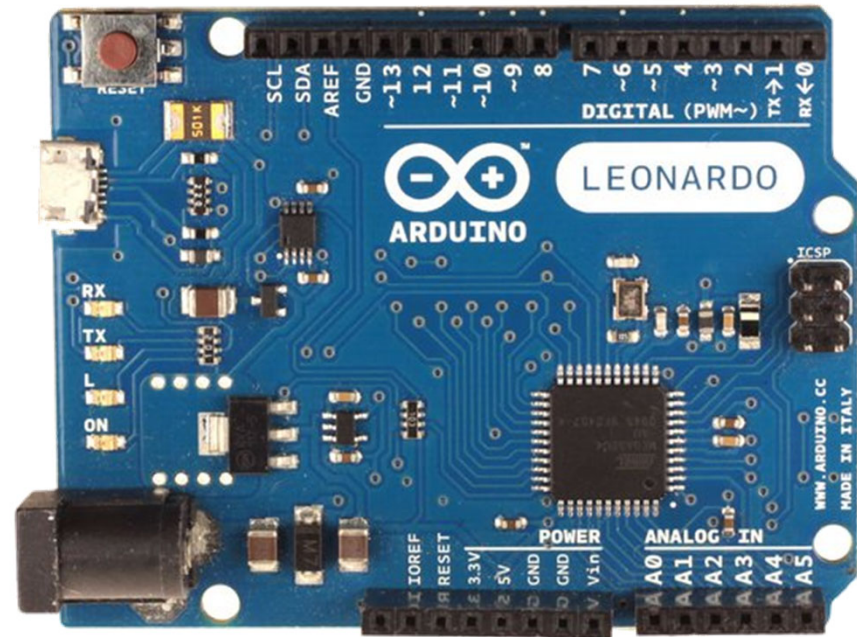
- Više različitih verzija
  - Broj ulaznih/izlaznih kanala
  - Oblik (gabariti)
  - Procesorska snaga
- Leonardo
- Due
- Micro
- LilyPad
- Esplora
- Uno



# Leonardo

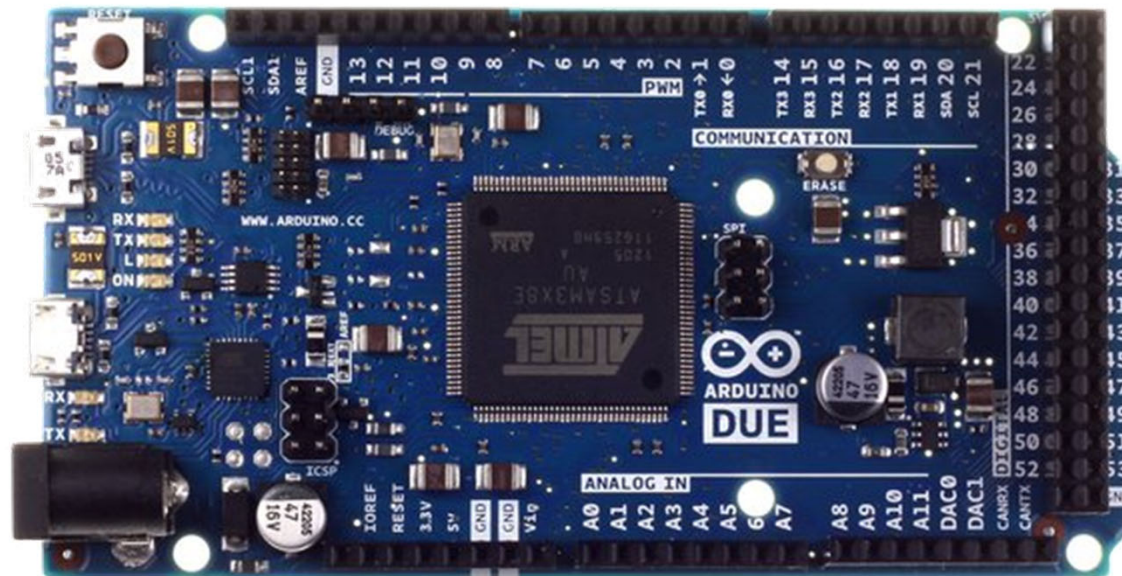
- U poređenju sa Uno, malo unaprijedjen.
- Koristi ATmega32u4 mikrokontroler koji ima ugrađenu USB komunikaciju

- Nema potrebe za dodatnim mikrokontrolerom
- Može se prikazati PC-u kao miš ili tastatura



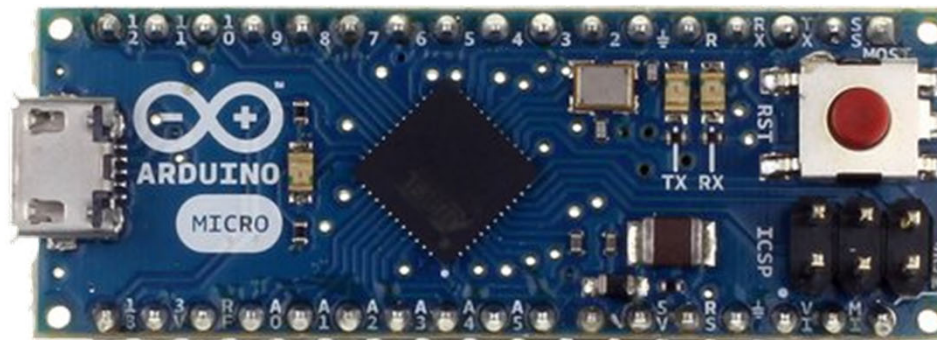
# Due

- Mnogo brži procesor, mnogo više pinova
- Radi na 3.3 volta
- Izgledom sličan Mega



# Micro

- Kad je veličina važna: Micro, Nano, Mini
- Uključuju sve funkcionalnosti Leonardo-a





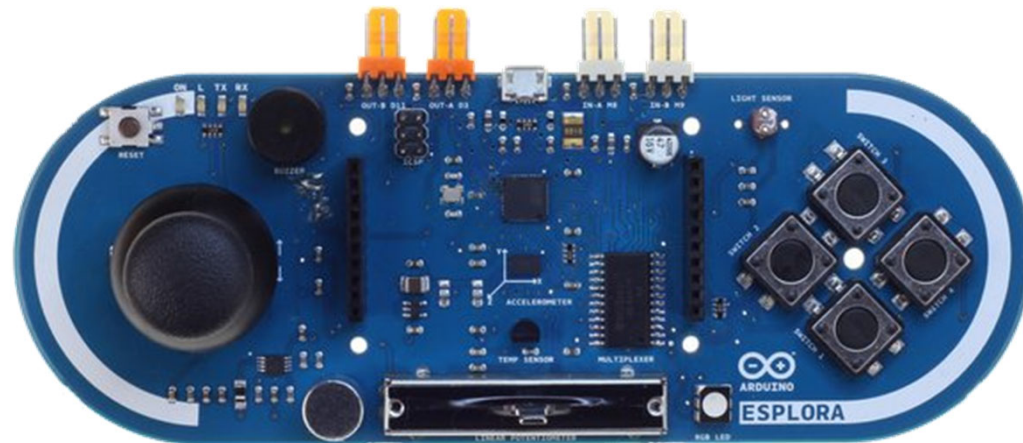
# LilyPad

- LilyPad je pogodan za primjenu na odjeći.



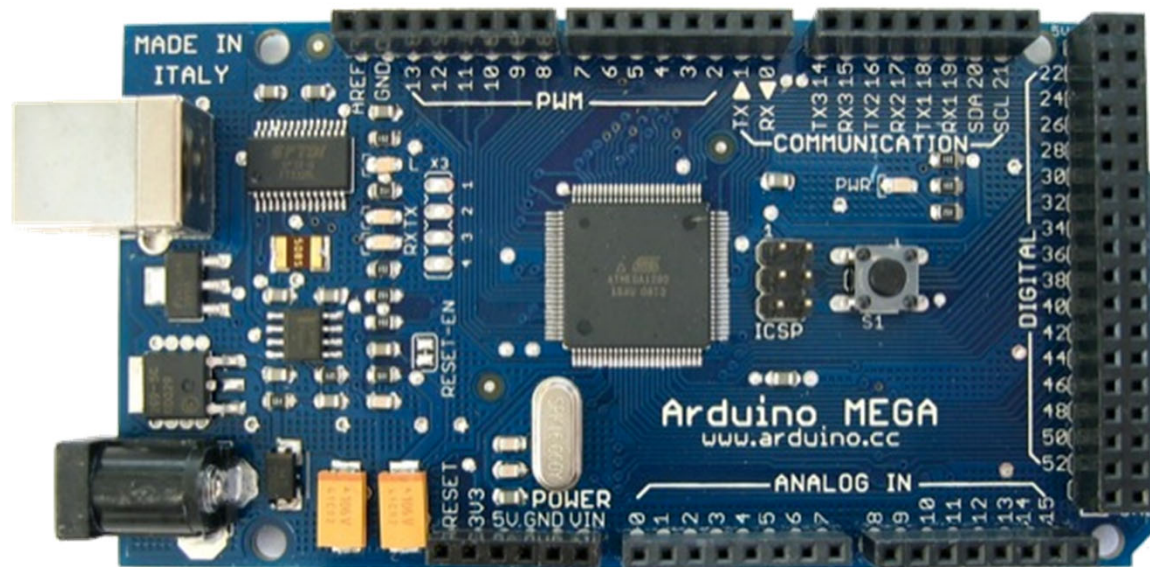
# Esplora

- Game controller
- Sadrži džojstik, četiri tastera, linearni potencijometar (klizač), mikrofon, svjetlosni senzor, senzor temperature, tro-osni akceleromatar.
- Nema standardi set IO pinova.



# Mega

- U poređenju sa Uno, Mega:
  - Mnogo više komunikacionih pinova
  - Više memorije



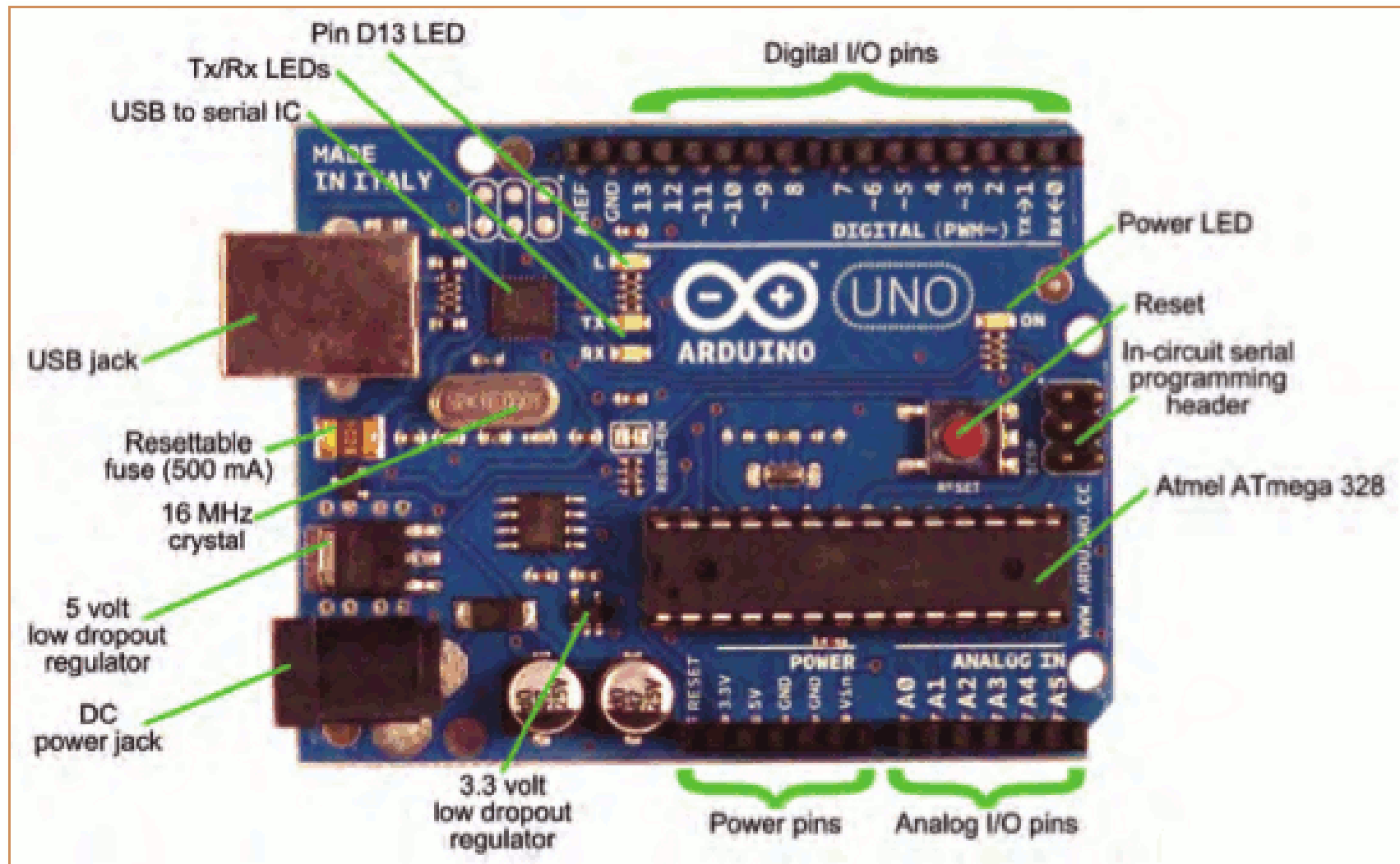


# Arduino Uno

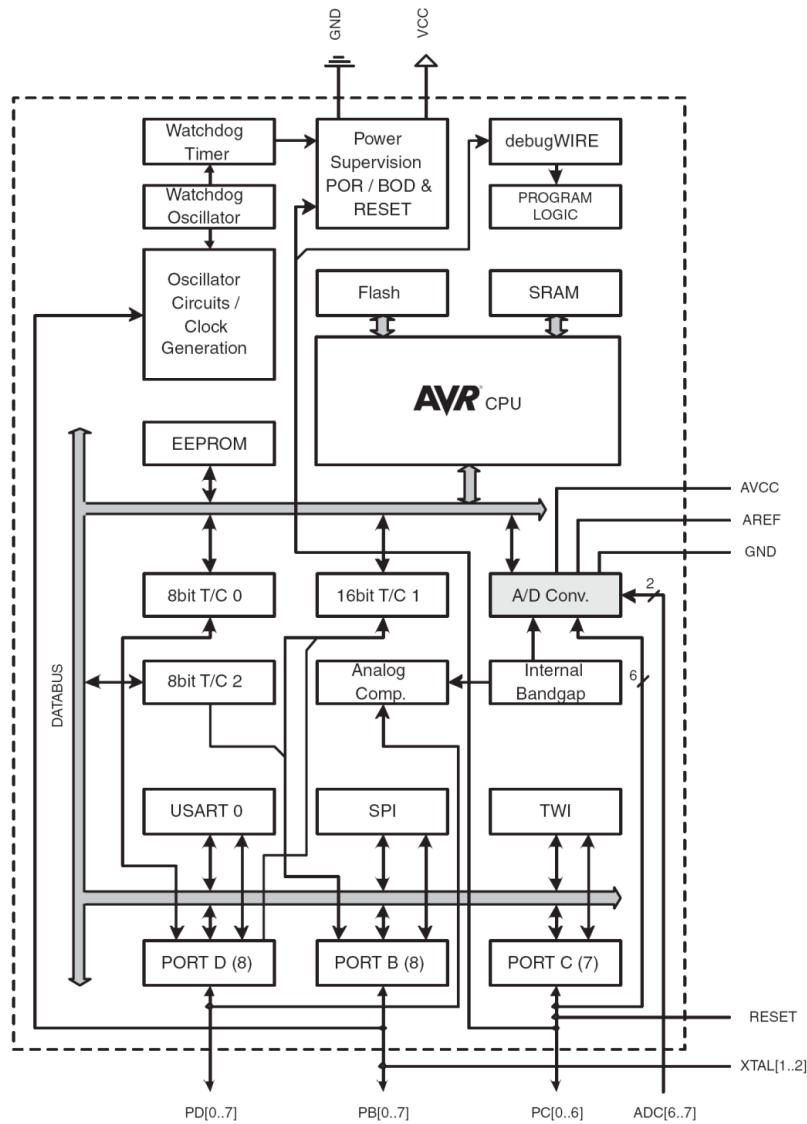
- Pinovi su grupisani u 3 grupe:
  - 14 digitalnih pinova
  - 6 analognih pinova
  - Napajanje
  - Pojavio se 2010



# Arduino Uno razvojna ploča

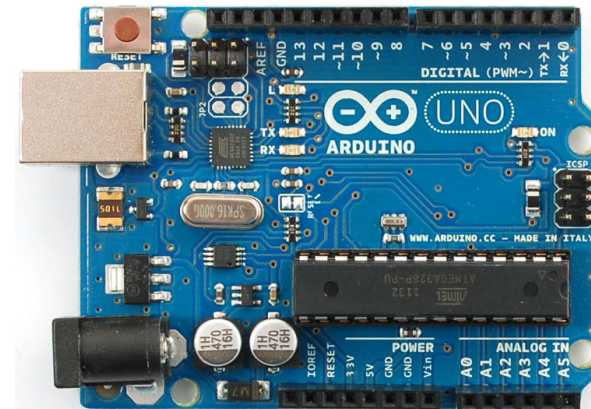


# ATmega328 unutrašnja architektura



(PCINT14/RESET) PC6	1	28	PC5 (ADC5/SCL/PCINT13)
(PCINT16/RXD) PD0	2	27	PC4 (ADC4/SDA/PCINT12)
(PCINT17/TXD) PD1	3	26	PC3 (ADC3/PCINT11)
(PCINT18/INT0) PD2	4	25	PC2 (ADC2/PCINT10)
(PCINT19/OC2B/INT1) PD3	5	24	PC1 (ADC1/PCINT9)
(PCINT20/XCK/T0) PD4	6	23	PC0 (ADC0/PCINT8)
		22	GND
		21	AREF
(PCINT6/XTAL1/TOSC1) PB6	9	20	AVCC
(PCINT7/XTAL2/TOSC2) PB7	10	19	PB5 (SCK/PCINT5)
(PCINT21/OC0B/T1) PD5	11	18	PB4 (MISO/PCINT4)
(PCINT22/OC0A/AIN0) PD6	12	17	PB3 (MOSI/OC2A/PCINT3)
(PCINT23/AIN1) PD7	13	16	PB2 (SS/OC1B/PCINT2)
(PCINT0/CLKO/ICP1) PB0	14	15	PB1 (OC1A/PCINT1)

ATmega328 data sheet pp. 2, 5



# ATmega328 karakteristike

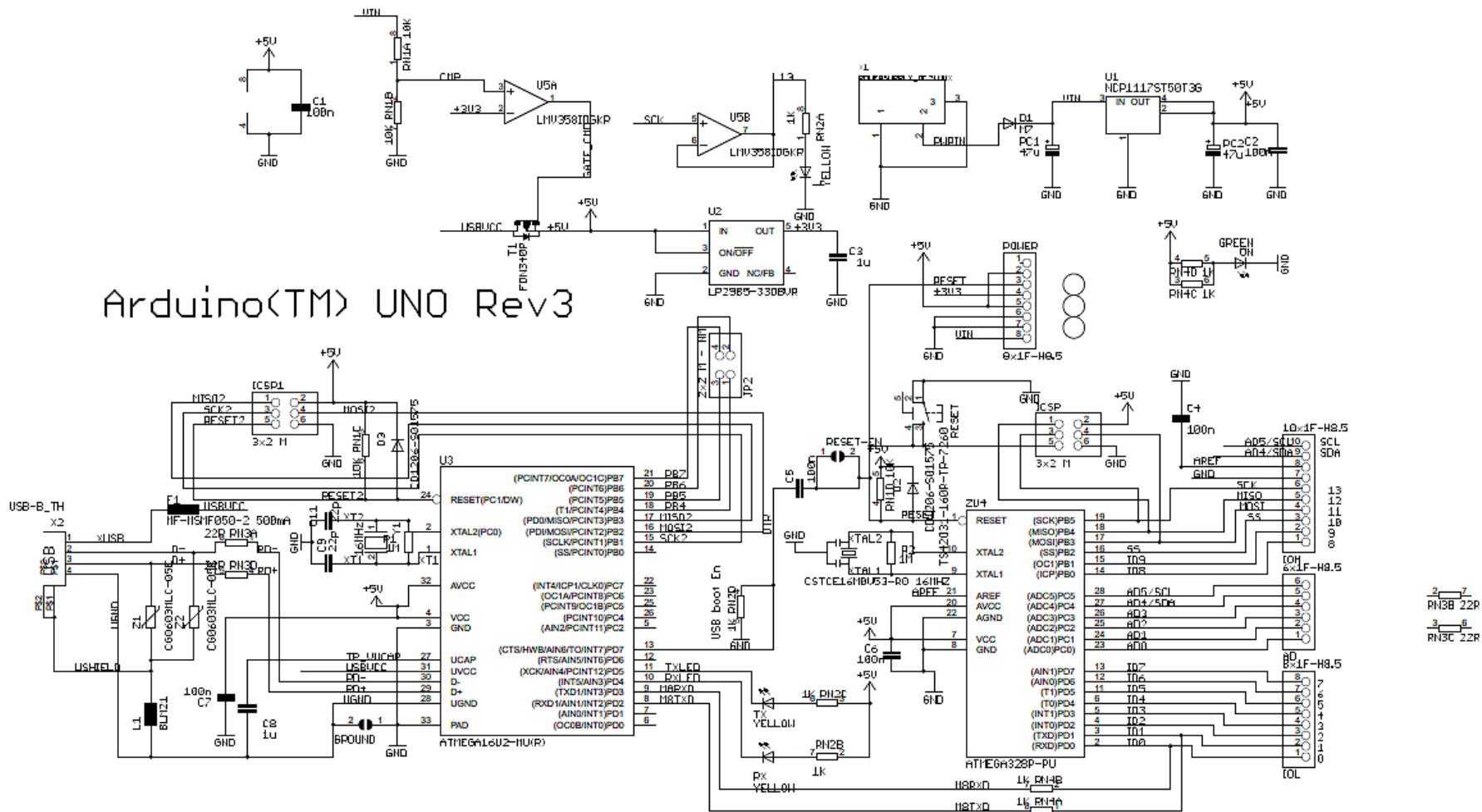
## Features

- High Performance, Low Power AVR® 8-Bit Microcontroller
- Advanced RISC Architecture
  - 131 Powerful Instructions – Most Single Clock Cycle Execution
  - 32 x 8 General Purpose Working Registers
  - Fully Static Operation
  - Up to 20 MIPS Throughput at 20 MHz
  - On-chip 2-cycle Multiplier
- High Endurance Non-volatile Memory Segments
  - 4/8/16/32K Bytes of In-System Self-Programmable Flash program memory
  - 256/512/512/1K Bytes EEPROM
  - 512/1K/1K/2K Bytes Internal SRAM
  - Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
  - Data retention: 20 years at 85°C/100 years at 25°C<sup>(1)</sup>
  - Optional Boot Code Section with Independent Lock Bits  
In-System Programming by On-chip Boot Program  
True Read-While-Write Operation
  - Programming Lock for Software Security
- Peripheral Features
  - Two 8-bit Timer/Counters with Separate Prescaler and Compare Mode
  - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
  - Real Time Counter with Separate Oscillator
  - Six PWM Channels
  - 8-channel 10-bit ADC in TQFP and QFN/MLF package  
Temperature Measurement
  - 6-channel 10-bit ADC in PDIP Package  
Temperature Measurement
  - Programmable Serial USART
  - Master/Slave SPI Serial Interface
  - Byte-oriented 2-wire Serial Interface (Philips I<sup>2</sup>C compatible)
  - Programmable Watchdog Timer with Separate On-chip Oscillator
  - On-chip Analog Comparator
  - Interrupt and Wake-up on Pin Change
- Special Microcontroller Features
  - Power-on Reset and Programmable Brown-out Detection
  - Internal Calibrated Oscillator
  - External and Internal Interrupt Sources
  - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby, and Extended Standby
- I/O and Packages
  - 23 Programmable I/O Lines
  - 28-pin PDIP, 32-lead TQFP, 28-pad QFN/MLF and 32-pad QFN/MLF
- Operating Voltage:
  - 1.8 - 5.5V
- Temperature Range:
  - -40°C to 85°C
- Speed Grade:
  - 0 - 4 MHz@1.8 - 5.5V, 0 - 10 MHz@2.7 - 5.5.V, 0 - 20 MHz @ 4.5 - 5.5V
- Power Consumption at 1 MHz, 1.8V, 25°C
  - Active Mode: 0.2 mA
  - Power-down Mode: 0.1 µA
  - Power-save Mode: 0.75 µA (Including 32 kHz RTC)

ATmega328 data sheet p. 1

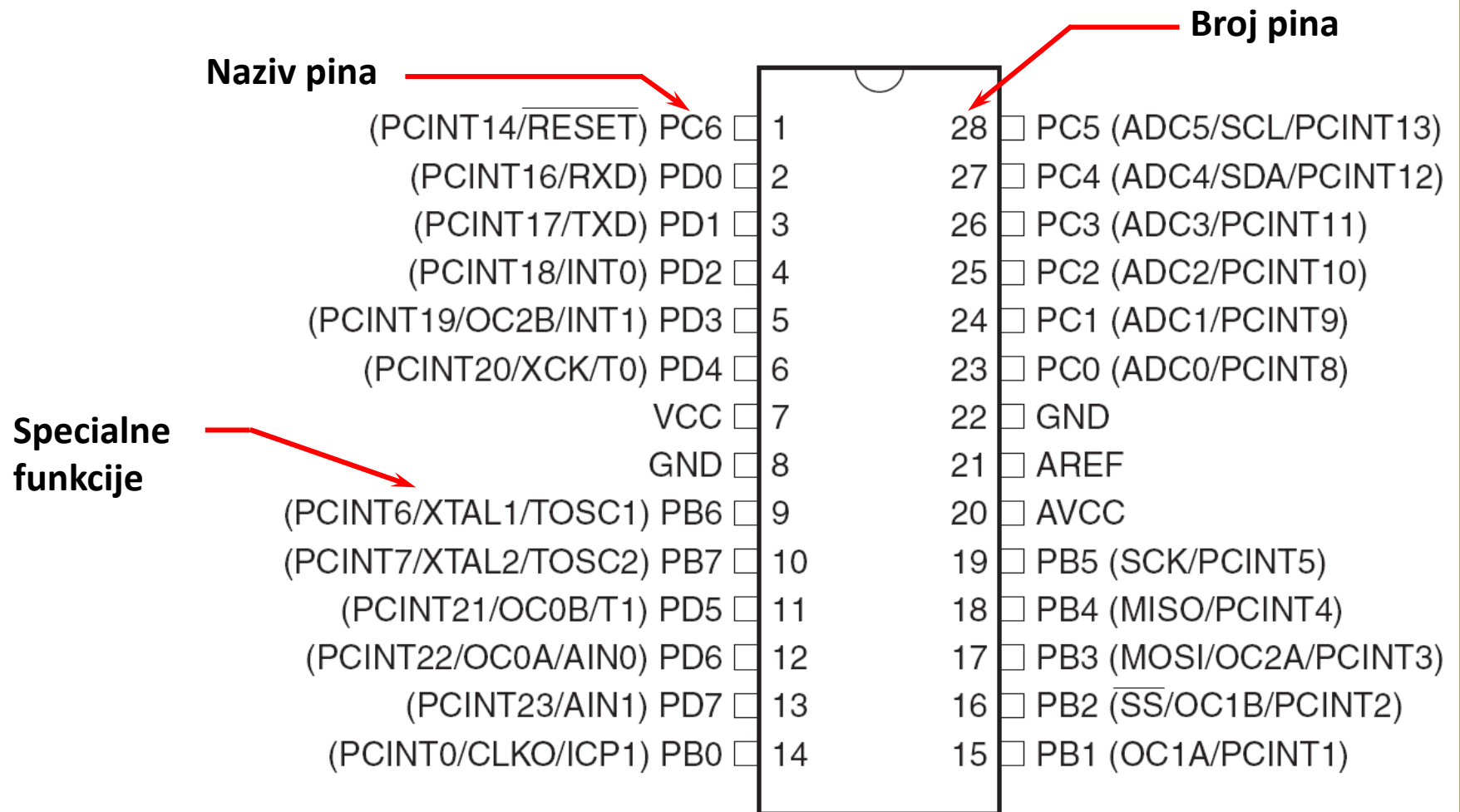
<http://ww1.microchip.com/downloads/en/devicedoc/doc2467.pdf>

# Arduino Uno – električna šema



Arduino(TM) UNO Rev3

# ATmega328 Microcontroller



# Absolutni maximumi

## 28.1 Absolute Maximum Ratings\*

\*NOTICE:

Stresses beyond those listed under “Absolute Maximum Ratings” may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Operating Temperature .....	-55°C to +125°C
Storage Temperature .....	-65°C to +150°C
Voltage on any Pin except $\overline{\text{RESET}}$ with respect to Ground .....	-0.5V to $V_{CC}+0.5V$
Voltage on $\overline{\text{RESET}}$ with respect to Ground.....	-0.5V to +13.0V
Maximum Operating Voltage .....	6.0V
DC Current per I/O Pin .....	40.0 mA
DC Current $V_{CC}$ and GND Pins .....	200.0 mA

ATmega328 data sheet p. 316

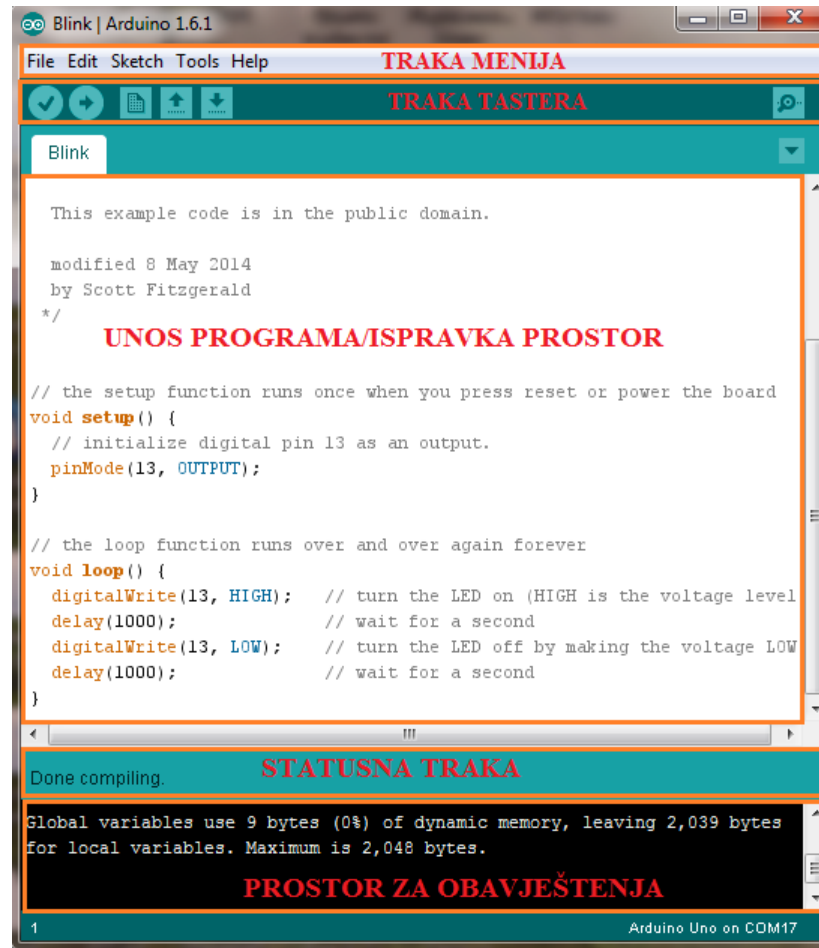


# Kako startovati?

- Posjetite: <http://arduino.cc/en/Guide/HomePage>
  1. Preuzmite & instalirajte Arduino environment (IDE)
  2. Povežite ploču sa računarom pomoću USB kabla
  3. Ako je potrebno, instalirajte dodatne drajvere
  4. Pokrenite Arduino IDE
  5. Selektujte razvojnu ploču
  6. Selektujte serijski port
  7. Otvorite blink primjer
  8. Upišite program u razvojnu ploču
  - ...
  9. Pisanje vlastitog programa
  10. Nerviranje/Debugiranje/Primoravanje da radi
  11. Oduševljenje i neposredno započinjanje novog projekta
  12. (spavanje samo kad se mora)



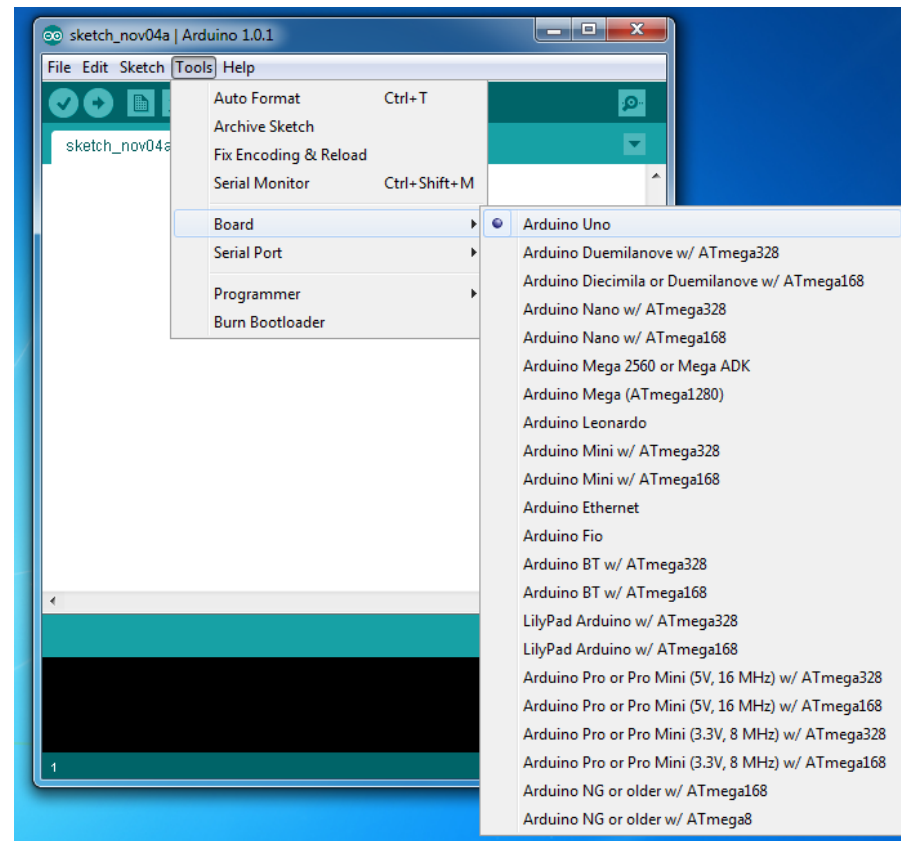
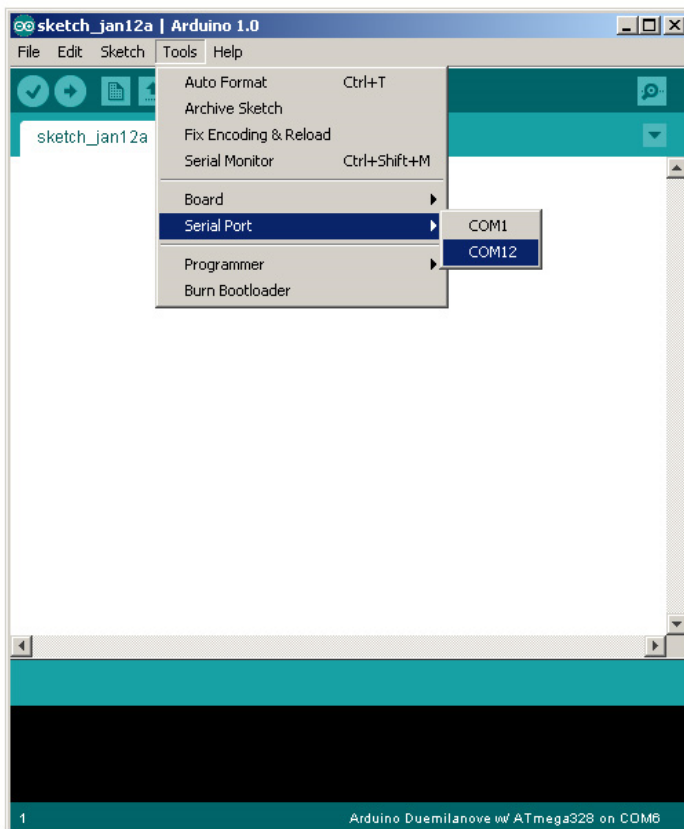
# Arduino IDE



```
Arduino 1.6.1
File Edit Sketch Tools Help
TRAKA MENIJA
TRAKA TASTERA
Blink
This example code is in the public domain.
modified 8 May 2014
by Scott Fitzgerald
*/
UNOS PROGRAMA/ISPRAVKA PROSTOR
// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin 13 as an output.
  pinMode(13, OUTPUT);
}
// the loop function runs over and over again forever
void loop() {
  digitalWrite(13, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000); // wait for a second
  digitalWrite(13, LOW); // turn the LED off by making the voltage LOW
  delay(1000); // wait for a second
}
Done compiling.
STATUSNA TRAKA
Global variables use 9 bytes (0%) of dynamic memory, leaving 2,039 bytes
for local variables. Maximum is 2,048 bytes.
PROSTOR ZA OBAVJEŠTENJA
1 Arduino Uno on COM17
```

Pogledajte: <http://arduino.cc/en/Guide/Environment>  
za više informacija

# Odaberite serijski port i ploču



# Razvoj Arduino programa

- Zasnovan na C++ bez 80% komandi.
- Pregršt novih komandi.
- Programi se nazivaju 'sketches' (skečevi, skice) .
- Skečevi obavezno sadrže dvije funkcije:
  - void setup( )
  - void loop( )
- setup( ) se pokreće prvi i samo jedanput.
- loop( ) se pokreće neprestano, dok se ne isključi napajanje ili se ne učita novi skeč.

# Arduino C

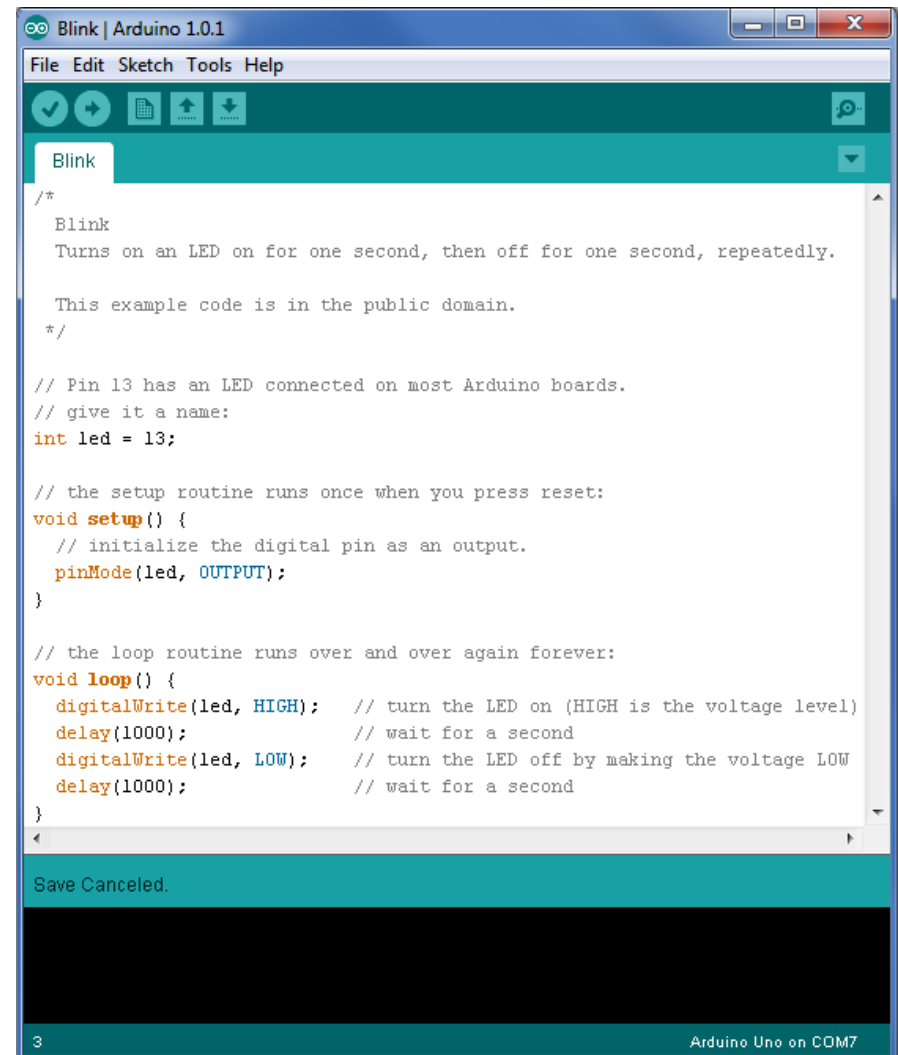
- Arduino skečevi uglavnom upravljaju pinovima na arduino ploči.
- Arduino skečevi su uvijek petlja.
  - void loop( ) {} je isto što i while(1) { }

# Arduino tajming

- `delay(ms)`  
Pauza nekoliko milisekundi
- `delayMicroseconds(us)`  
Pauza nekoliko mikrosekundi
- Više komandi: [arduino.cc/en/Reference/HomePage](https://arduino.cc/en/Reference/HomePage)

# Osobine kompajlera

- Brojni jednostavni skečevi su uključeni u kompajler
- Nalaze se pod opcijom File, Examples
- Kada je skeč napisan, može se upisati u programsku memoriju mikrokontrolera na Arduino štampanoj ploči kroz opcije File, Upload, ili pritiskom na <Ctrl> U



The screenshot shows the Arduino IDE interface with the 'Blink' sketch loaded. The code is as follows:

```
Arduino 1.0.1
File Edit Sketch Tools Help
Blink
/*
 * Blink
 * Turns on an LED on for one second, then off for one second, repeatedly.
 *
 * This example code is in the public domain.
 */
// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
int led = 13;

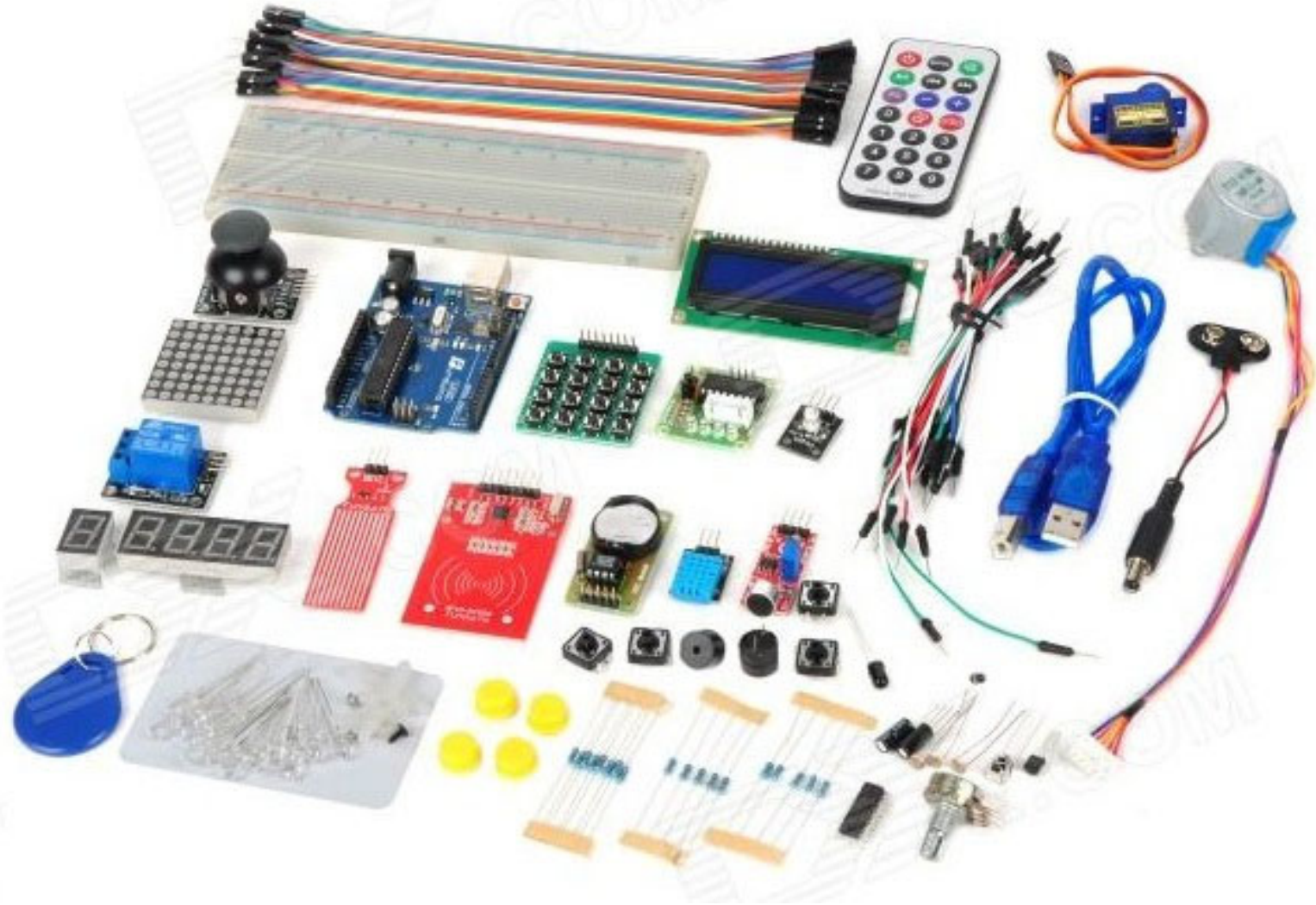
// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);             // wait for a second
  digitalWrite(led, LOW);  // turn the LED off by making the voltage LOW
  delay(1000);             // wait for a second
}
Save Canceled.
3 Arduino Uno on COM7
```



# Arduino oprema

## Keys RFID Learning Module Set for Arduino



# Konkurencija Arduinu

- Alternative Arduino platformi
  - Pinguino – PIC kontroler
  - MSP430 – Texas Instruments;
  - Drugi: korisnički, Teensy, itd.
- Netduino
- Računari
  - Raspberry Pi
  - BeagleBones – TI; ima računar i kontroler

# Netduino

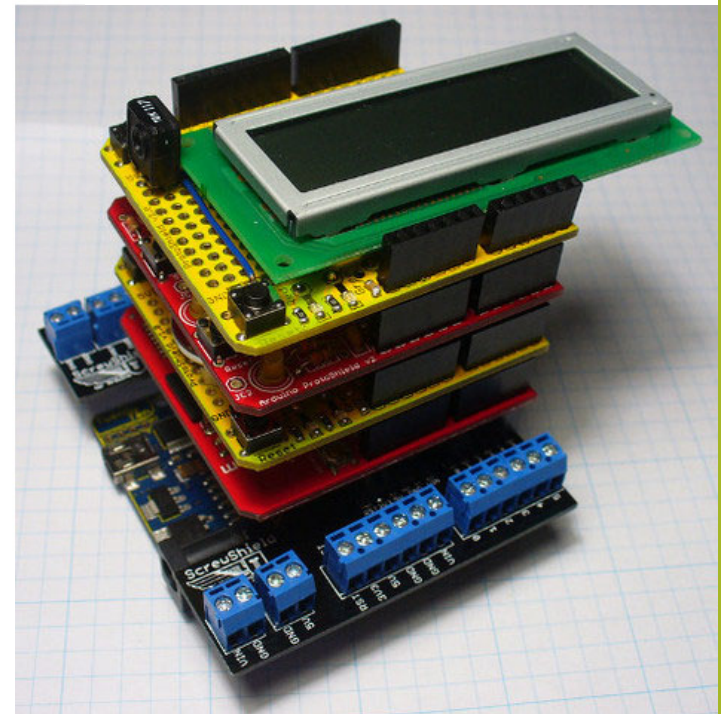
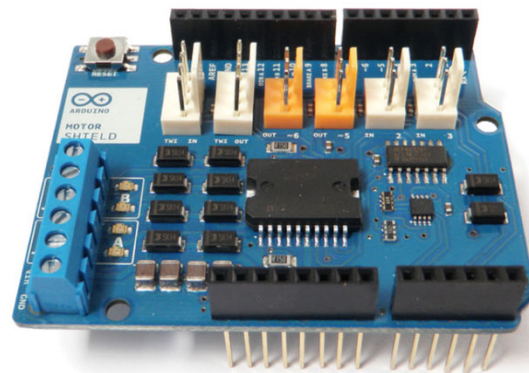
- Mikrokontroler i razvojno okruženje kreirano od strane Microsoft-a za rad sa .NET Micro Framework.
- DALEKO moćnije razvojno okruženje.
- Razlike
  - Pinovi na Netduino su 3.3V (ne 5V).
  - Netduino ima mnogo brži procesor
  - 60K RAM-a (naspram Uno-vih 2K).
- U velikoj mjeri kompatibilan sa Arduino-om, ali nije potpuno izmjenjiv.

# Raspberry Pi

- Pojednostavljeni računar, ne kontroler
- Koristi Debian Linux
  - Arch Linux ARM, Fedora, FreeBSD, Slackware...
- Programiran sa Python-om
  - BBC BASIC, C, Perl
- Kako je u pitanju računar (ime operativni sistem), ima drugačiju ulogu u projektima.
- Hijerarhija: računar upravlja kontrolerima, kontroleri upravljaju hardverom.

# Šildovi (Dodaci)

- Šildovi su ploče koje se dodaju na Arduino ploču.
- Oni proširuju mogućnosti Arduina.
- Primjeri:
  - Ethernet
  - GPS
  - Motor
  - Prototip
- [shieldlist.org](http://shieldlist.org)



# Zaključak

- Arduino platforma predstavlja jeftin način da se uđe u svijet robotike.
- Arduino ima:
  - Brojne korisnike
  - Bogatu online biblioteku kodova i projekata



Kraj