CONTROLUL TURAȚIEI UNUI MOTOR DE CURENT CONTINUU CU AJUTORUL PLATFORMEI ARDUINO

3.1. Generalități platformă Arduino

Arduino este o platformă cu microcontroler care poate fi folosită pentru dezvoltarea de aplicații interactive. Practic, informația este preluată de la elementele de intrare (senzori şi comutatoare), se procesează în interiorul microcontrolerului platformei şi este transmisă către elementele de ieşire: leduri, motoare, actuatoare, etc.

Pentru a face o idee despre ceea ce se poate face cu Arduino, iată câteva exemple de proiecte care se pot realiza cu Arduino:

- OpenEnergyMonitor, sistem bazat pe Arduino care monitorizează energia electrică consumată în casă;
- Arduino + senzor temperatură + senzor umiditate + senzor presiune atmosferică + placă de rețea Ethernet care transmite datele de mediu pe Google Docs, la fiecare 10 secunde;
- mână robotică, bazată pe o mânuşă cu senzori de îndoire si servomotoare;
- robot autonom care ocoleşte obstacole;
- robot controlat prin Bluetooth folosind telefonul mobil sau laptop-ul;
- dispozitiv pentru pictat ouă („the EggBot”);
- acces bazat pe cartele RFID + notificari prin Twitter.

Exista mai multe variante de placi de dezvoltare Arduino, cum ar fi: Mega, Diecimila, Duemilanove, Mini, Nano şi chiar Bluetooth Arduino, cele mai noi produse fiind Arduino Uno şi Arduino Mega 2560.

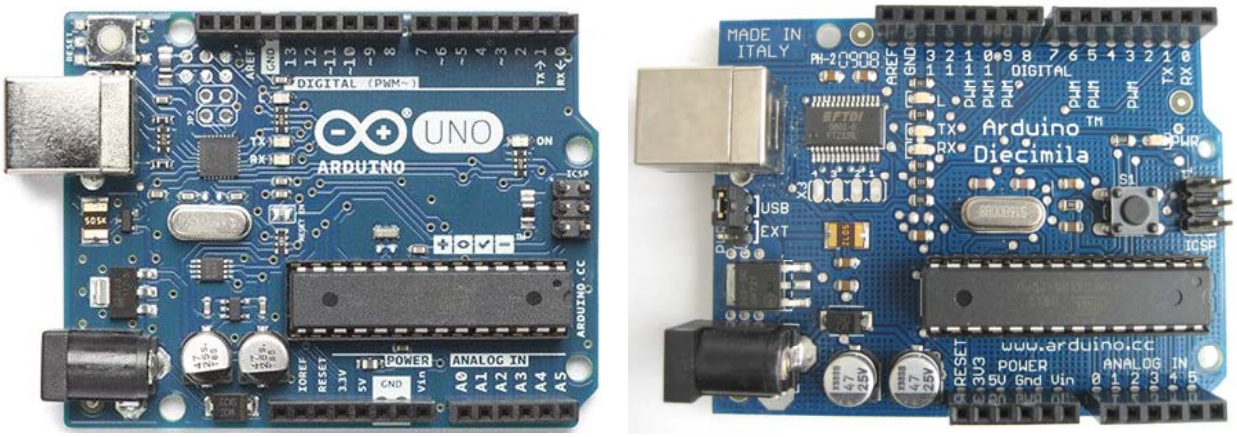

Fig. 3.1. Platforme Arduino

O platformă Arduino este compusă dintr-un microcontroler Atmel AVR de 8, 16 sau 32 biți în special ATmega8, ATmega168, ATmega328, ATmega1280 şi ATmega2560 precum şi componente complementare care facilitează programarea şi încorporarea în alte circuite.

Pentru programare se utilizează software-ul Arduino IDE (Integrated Development Environment) care suportă limbajele de programare C şi C++ folosind reguli speciale de organizare a codului.

3.2. Platforma Arduino Uno

Arduino Uno este o placa de dezvoltare bazată pe microcontrolerul ATmega328P, având 6 intrări analogice, 14 de intrări digitale/pini de ieşire (din care 6 pot fi utilizate ca ieşiri PWM), un oscilator cu quart de 20 MHz, o conexiune USB, o mufa de alimentare, şi un buton de resetare.

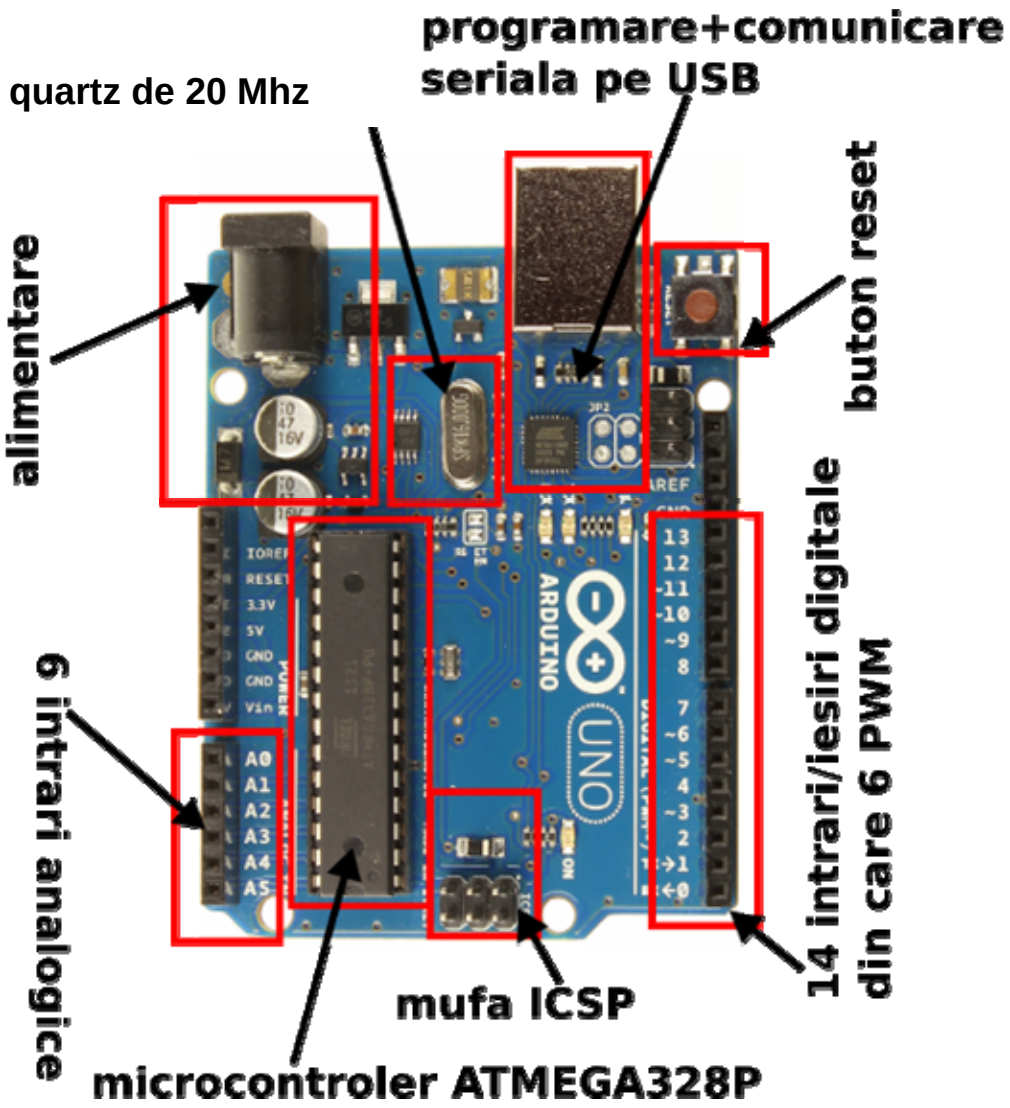

Fig. 3.2. Arduino Uno

3.3. Controlul turației unui motor de curent continuu

Elementele componente ale circuitului ce realizează controlul turației unui motor de curent continuu:

a. microcontrolerul ATmega328P

Caracteristicile microcontrolerului ATmega328P din platforma Arduino Uno sunt prezentate în tabelul 3.1.

Tabel 3.1. Caracteristicile microcontrolerului Atmega328P

Frecvența	**20 MHz**
Tensiunea de alimentare	1,8 – 5,5 V
Număr de pini	**28**
Memorie SRAM	**2 kB**
Memorie EEPROM	**1 kB**
Memorie Flash	**32 kB**
Cicluri scriere/citire	10000 Flash/100000 EEPROM
Număr de pini intrare ieşire	23
Canale conversie analog - digitală	8
Timere pe 8 biți	2
Timer pe 16 biți	1
Consum de energie	Funcționare 0,2 mA Stand-bye 0,1 µA

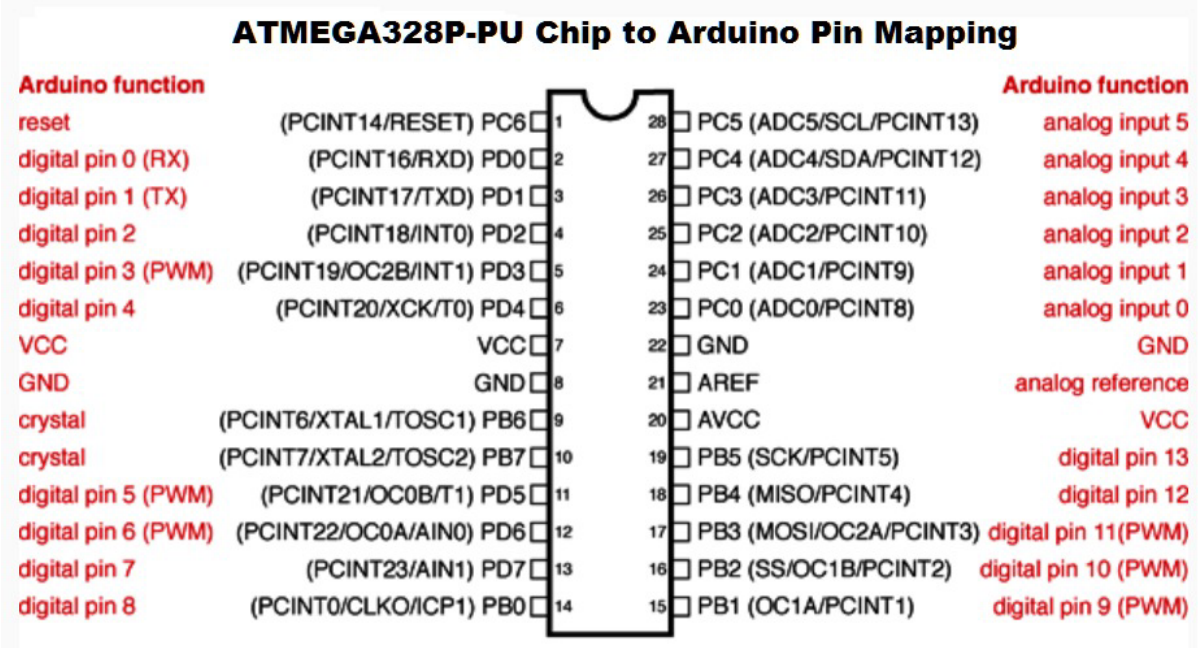

Fig. 3.3. Pinii micrcontrolerului Atmega328P

b. motor pas cu pas (unipolar de curent continuu)

Un **motor electric** (sau **electromotor**) este un dispozitiv electromecanic ce transformă energia electrică în energie mecanică. Indiferent de tipul motorului, acesta este construit din două părți componente: **stator** şi **rotor**.

Motorul pas cu pas este un dispozitiv electromecanic care converteşte impulsurile electrice în mişcări discrete. Motorul pas cu pas este un motor controlat de o serie de bobine electromagnetice. Axul central are o serie de magneți montați, iar bobinele ce înconjoară axul se alimentează în alternanță, realizând astfel mișcarea de rotație a motorului.

La apariția unui semnal de comandă pe unul din polii statorici rotorul se va deplasa până când polii săi se vor alinia în dreptul polilor opuşi statorici. Rotirea acestui tip de rotor se va face practic din pol în pol, de unde şi denumirea sa de motor pas cu pas.

Motorul pas cu pas face parte din categoria motoarelor sincrone, deoarece viteza de deplasare a rotorului exprimată prin numărul de paşi efectuați în unitatea de timp, depinde direct de frecvența impulsurilor de alimentare.

Motorul pas cu pas unipolar are între patru şi cinci fire împreună cu patru bobine. Motoarele pas cu pas se folosesc acolo unde este necesară precizie ridicată (hard disc, copiatoare).

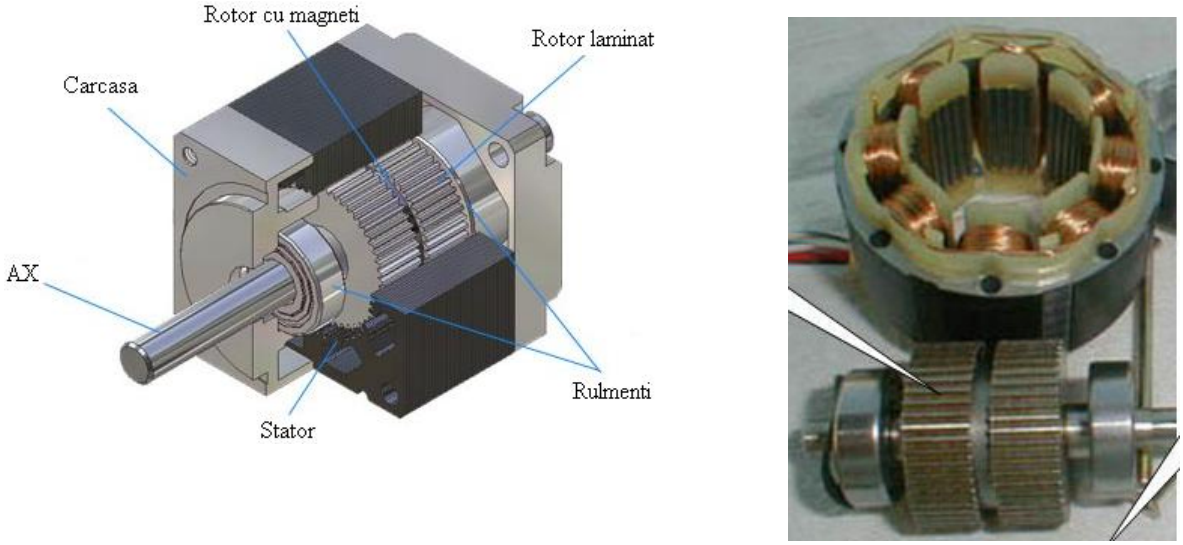

Fig. 3.4. Componentele motorului pas cu pas

c. matrice Darlington U2004 – permite alimentarea motorului pas cu pas şi comanda acestuia cu ajutorul platformei Arduino.

d. potențiometru

Este un rezistor cu trei terminale cu un contact culisant sau rotativ, ce formează un divizor de tensiune reglabilă. În cazul folosirii numai a două terminale, acesta acționează ca un rezistor variabil sau reostat.

e. alimentare 9V

f. breadboard

g. fire de legătură

În figura 3.5. se prezintă schema de principiu a circuitului Arduino pentru controlul turației unui motor de curent continuu.

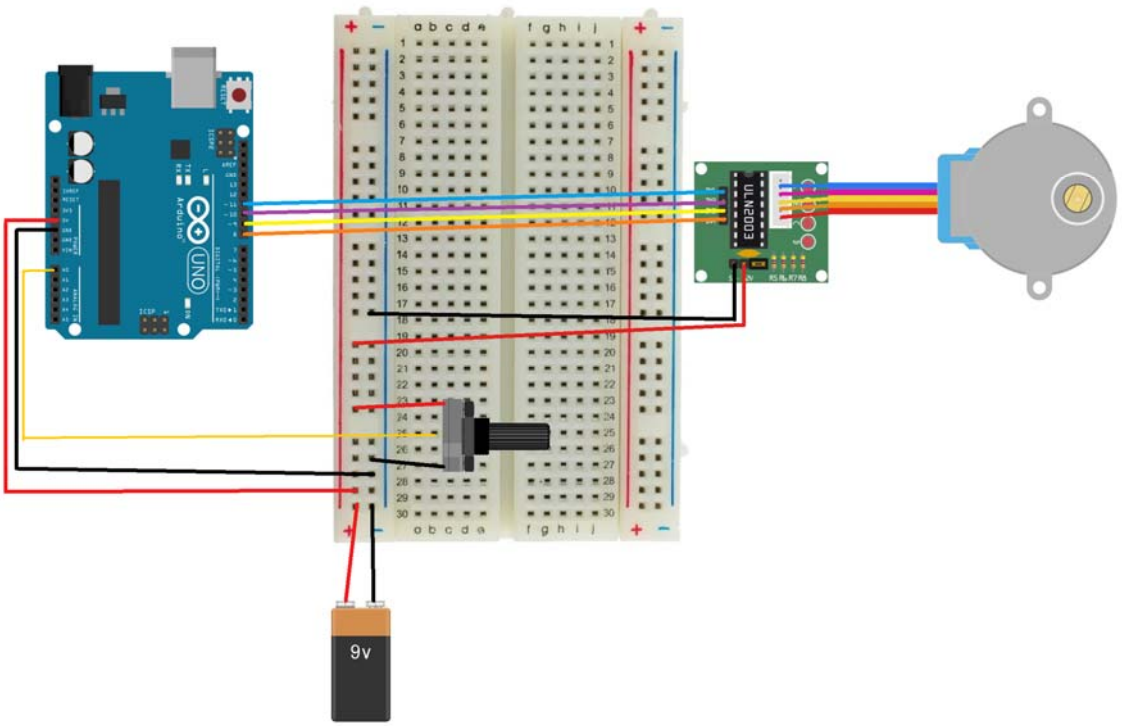

Fig. 3.5. Schema de principiu a circuitului

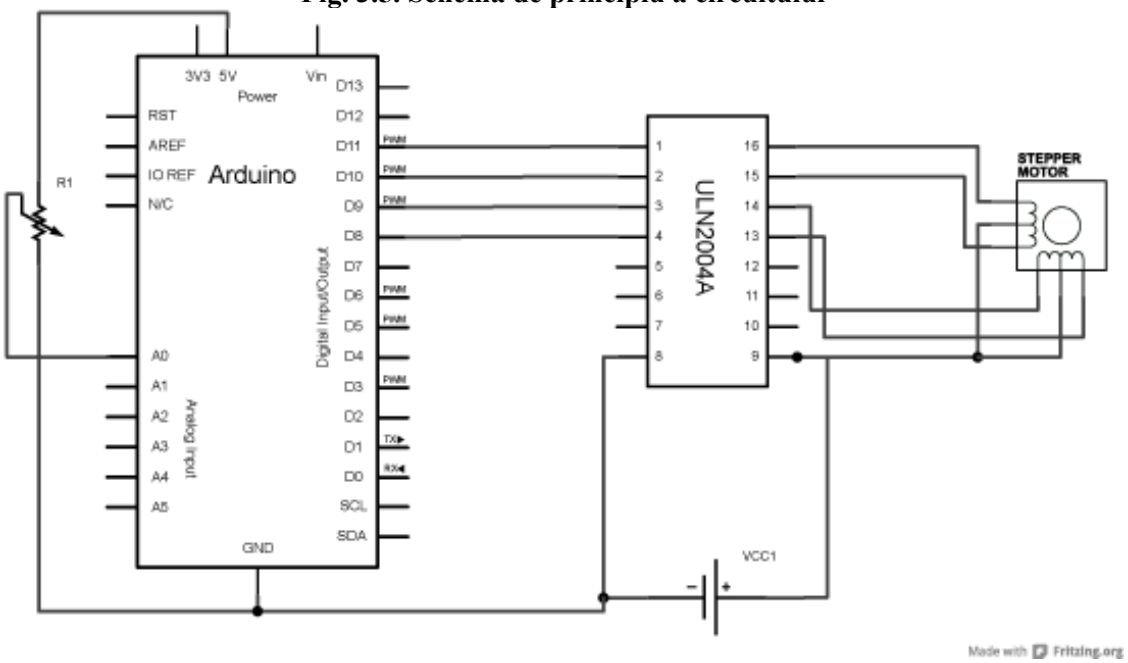

Fig. 3.6. Conectarea motorului pas cu pas la platforma Arduino

3.4. Programul pentru controlul turației

Motorul este ataşat la pinii digitali 8 – 11 al microcontrolerului Arduino. Un potențiometru este conectat la intrarea analogica 0.

Motorul se va roti în sensul acelor de ceasornic. Creşterea valorii potențiometrului este direct proporțională cu viteza motorului.

Din moment ce comanda „setSpeed()" setează pauza dintre trepte, se observă că motorul este mai puțin sensibil la viteze mici.

```
#include <Stepper.h>
const int stepsPerRevolution = 200; //numărul de paşi pentru o rotație
Stepper myStepper(stepsPerRevolution, 8, 9, 10, 11); //pinii 8, 9, 10 şi 11 sunt definiți ca pini pentru alimentarea motorului
int stepCount = 0;  //se inițializează cu 0 numărul de paşi ai motorului
void setup()
{
}
void loop()
{
int sensorReading = analogRead(A0); //se citeşte poziția potențiometrului
int motorSpeed = map(sensorReading, 0, 1023, 0, 100);
//se setează viteza motorului
if (motorSpeed > 0)
{
myStepper.setSpeed(motorSpeed);
//pasul este 1/100 pentru o rotație
myStepper.step(stepsPerRevolution / 100);
}
}
```