

Variator digital de tensiune continuă

Programul de practică E.P.E. din cadrul Universității Tehnice din Cluj
Napoca – facultatea: Inginerie Electrică, secția: Electronică de Putere

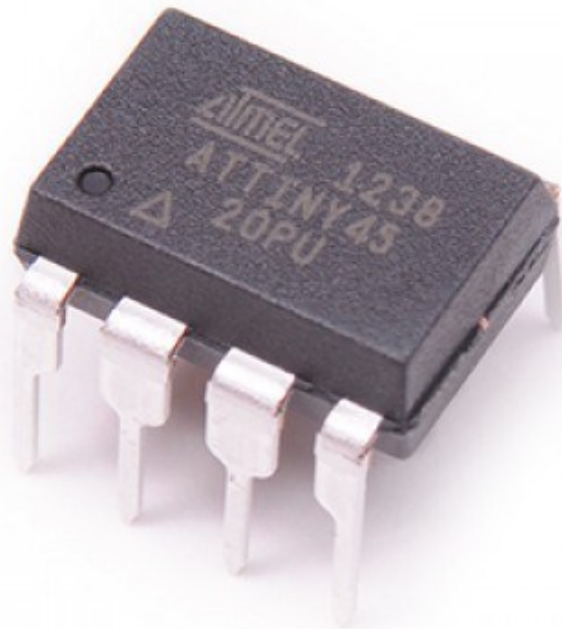
Puncte de reper

- ❖ Prezentarea micro – controllerului integrat ATTiny 45;
- ❖ Familiarizarea cu conceptul de „modulație în lățime a impulsului”;
- ❖ Familiarizarea cu conceptele de „circuit de forță și comandă”;
- ❖ Concretizarea aplicației propuse (implementare);

Puncte de reper

- ❖ **Prezentarea micro – controllerului integrat ATTiny 45;**
- ❖ Familiarizarea cu conceptul de „modulație în lățime a impulsului”;
- ❖ Familiarizarea cu conceptele de „circuit de forță și comandă”;
- ❖ Concretizarea aplicației propuse (implementare);

Micro – controllerul ATTiny 45

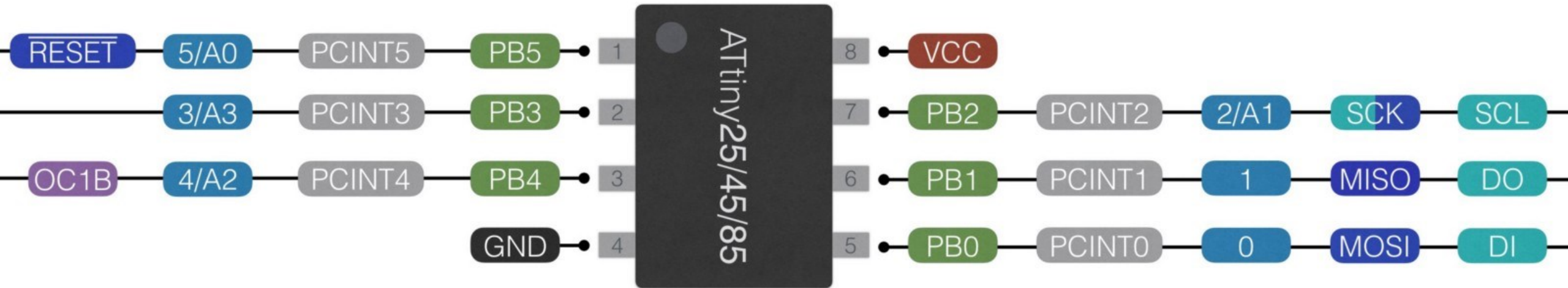


Micro – controllerul ATTiny 45

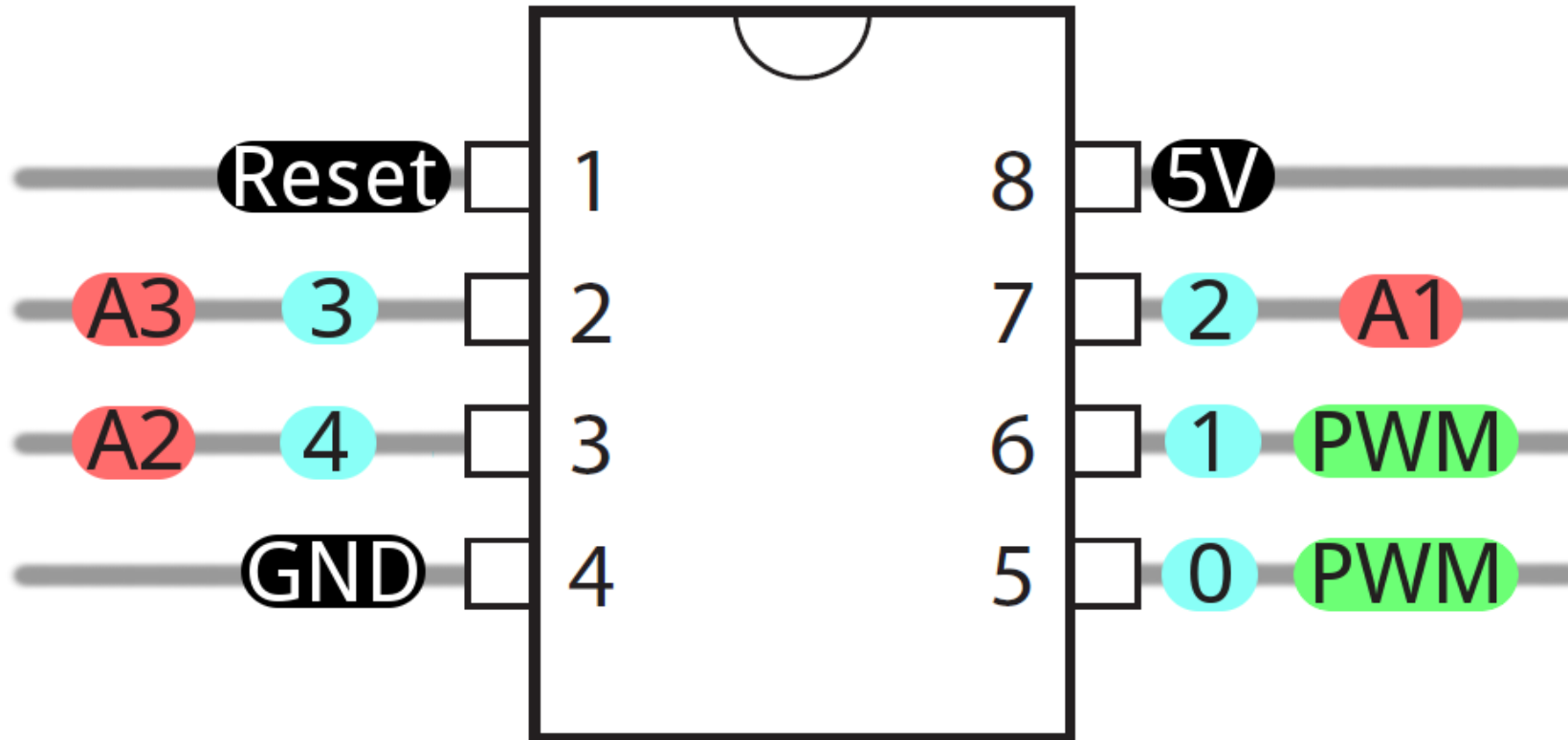
- ✓ Permite implementarea aplicațiilor cu micro – controller la nivel de cablaj imprimat (eng. printed circuit board) sau plăcuțe de test;
- ✓ Capsula „DIP 8” permite instalarea circuitului integrat într-un soclu;
- ✓ Prezintă un consum redus de energie;
- ✓ Logica de funcționare a circuitului este **PROGRAMABILĂ!**

Terminale de acces

ATtiny25/45/85 pinout



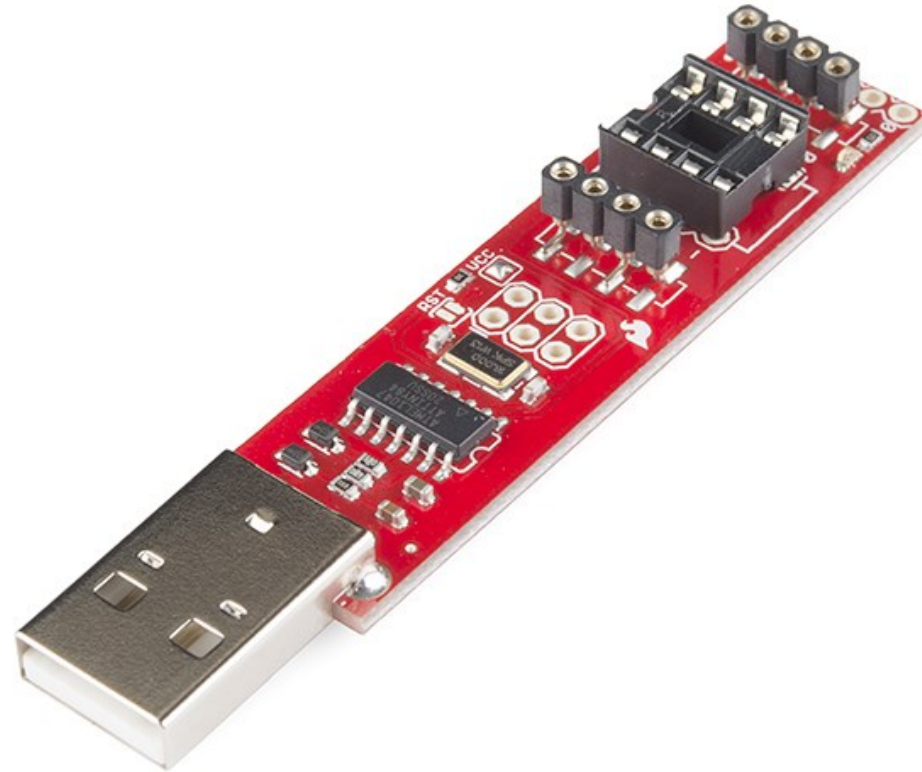
Terminale de acces



Micro – controllerul ATtiny 45

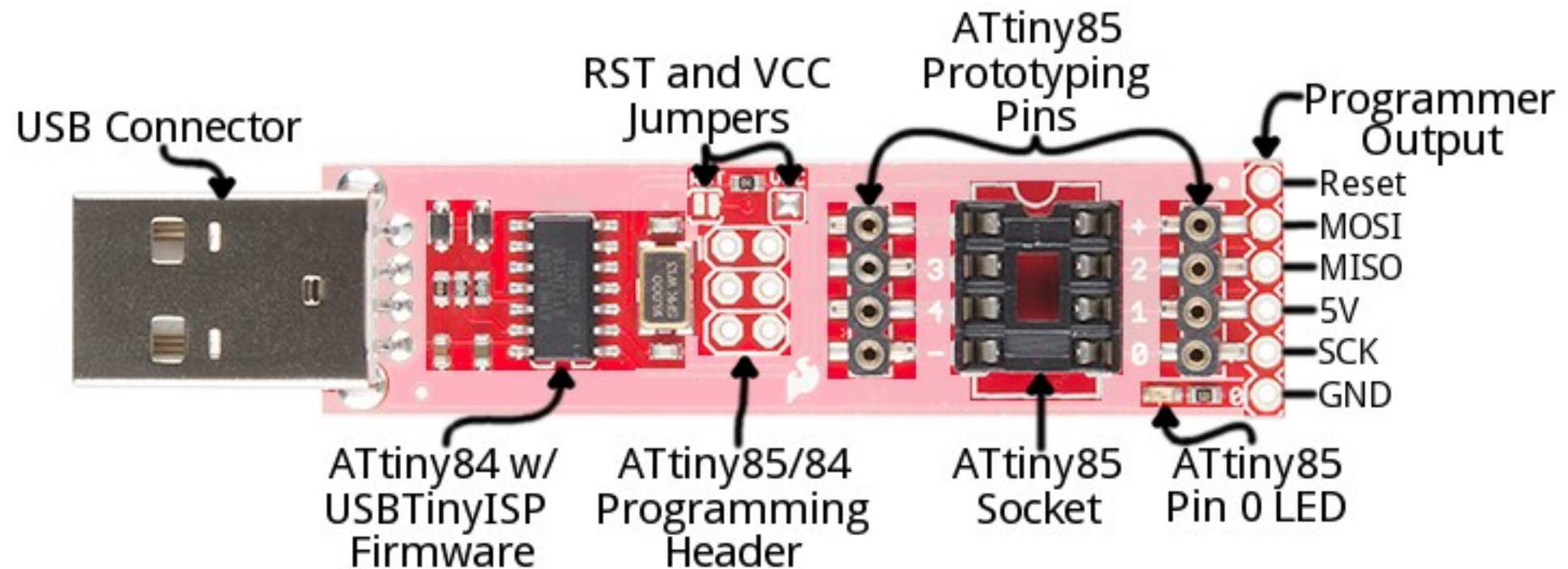
- ✓ Memorie program de tip flash cu capacitatea de: 4 KB;
- ✓ Memorie sRAM: 256 Byte;
- ✓ Memorie de date EEPROM: 256 Byte;
- ✓ 6 intrări și ieșiri digitale de uz general (eng. GPIO);
- ✓ 32 de regiștrii de uz general (eng. GPWR);
- ✓ Un numărător cu reprezentare pe 8 biți;
- ✓ Convertor analog – digital cu reprezentare pe 10 biți;
- ✓ Sistem de întreruperi interne (eng. soft) și externe (eng. hard);

Programator SparkFun USBTinyISP - TinyAVR

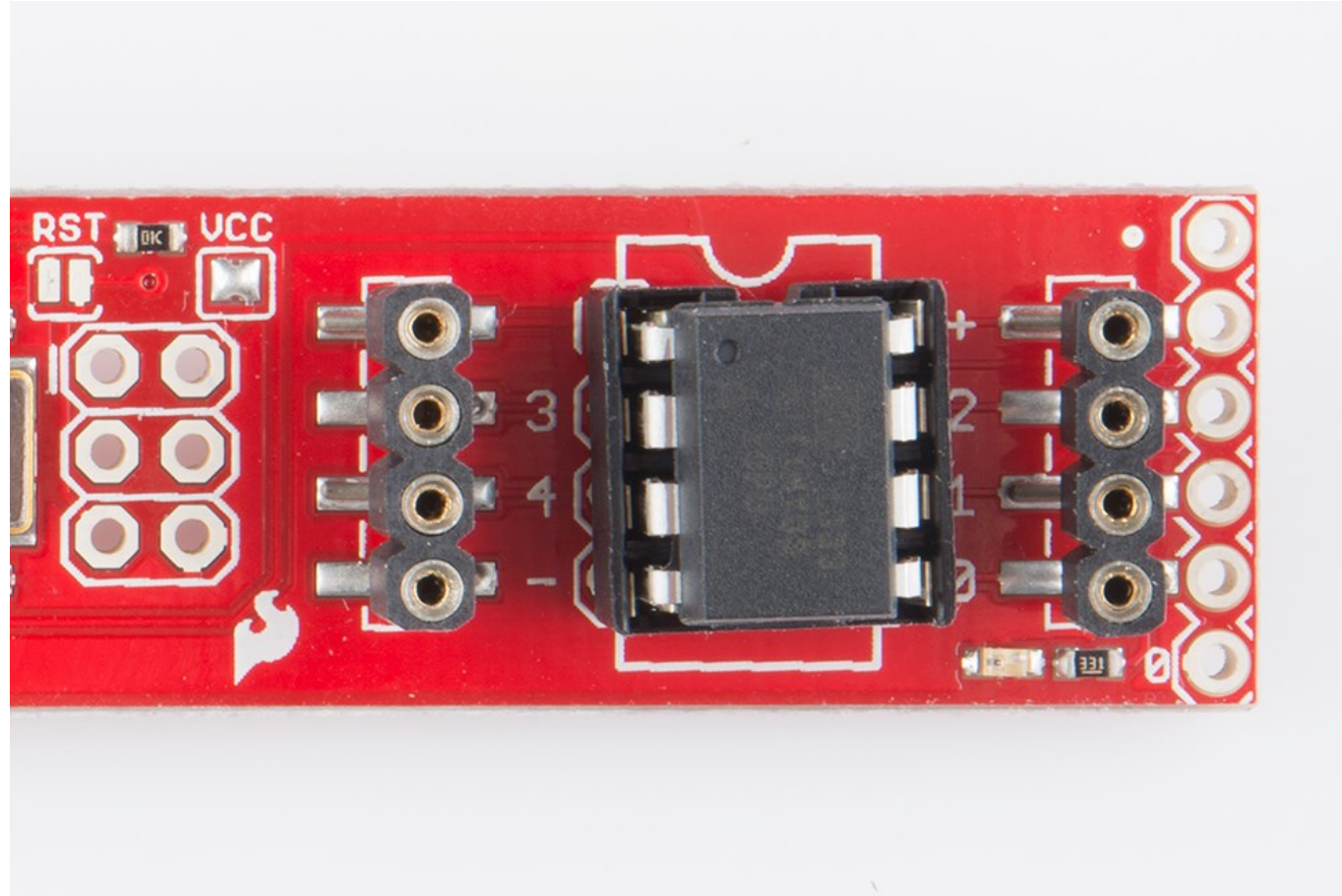


<https://www.sparkfun.com/products/11801>

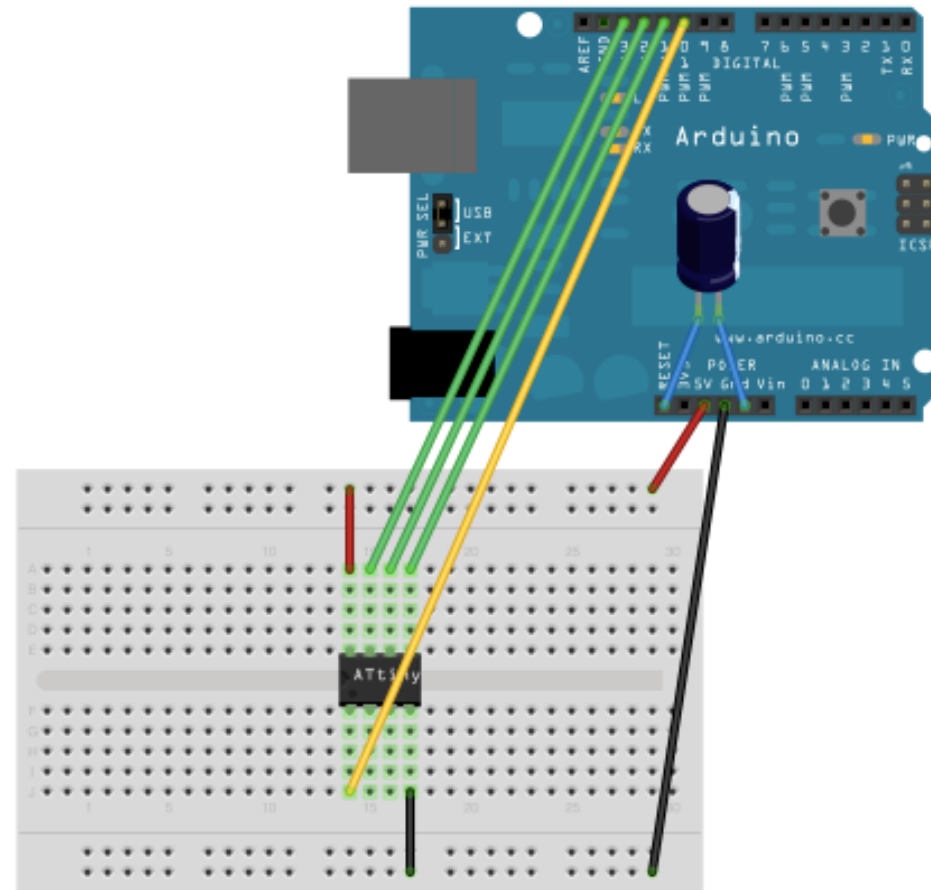
Programator SparkFun USBTinyISP - TinyAVR



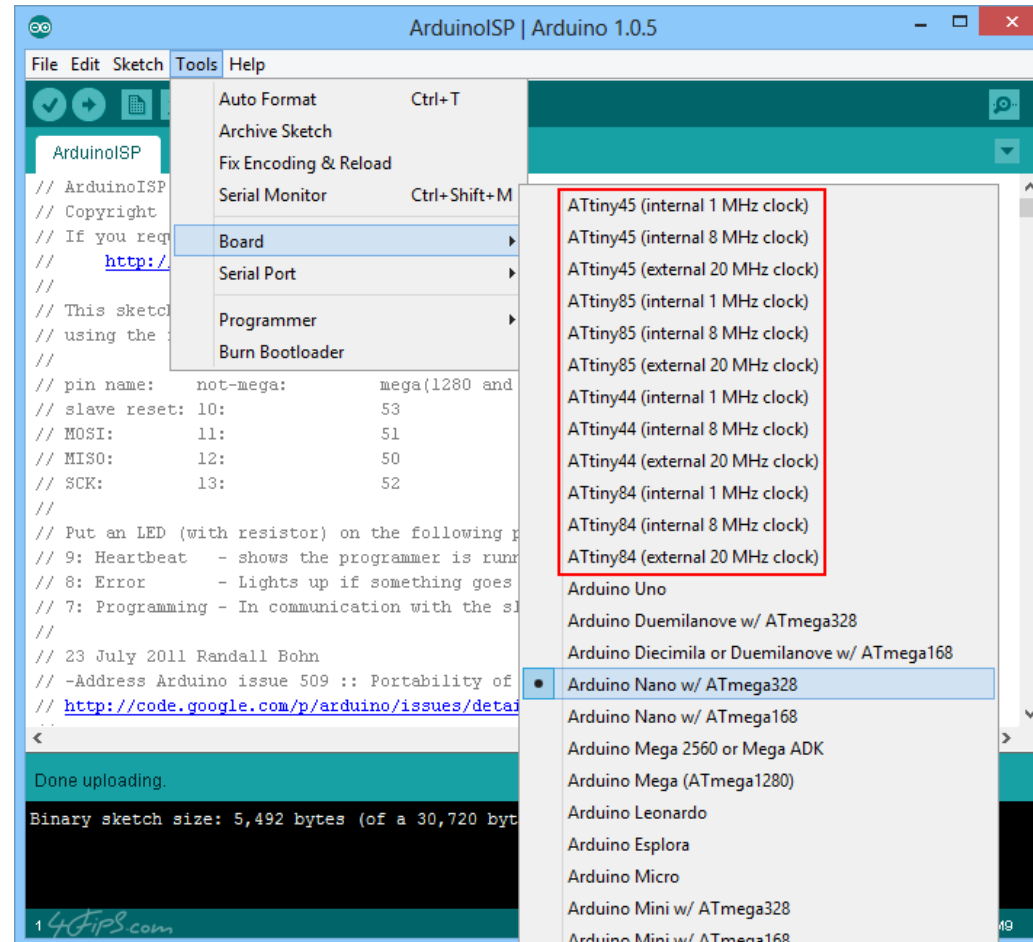
Programator SparkFun USBTinyISP - TinyAVR



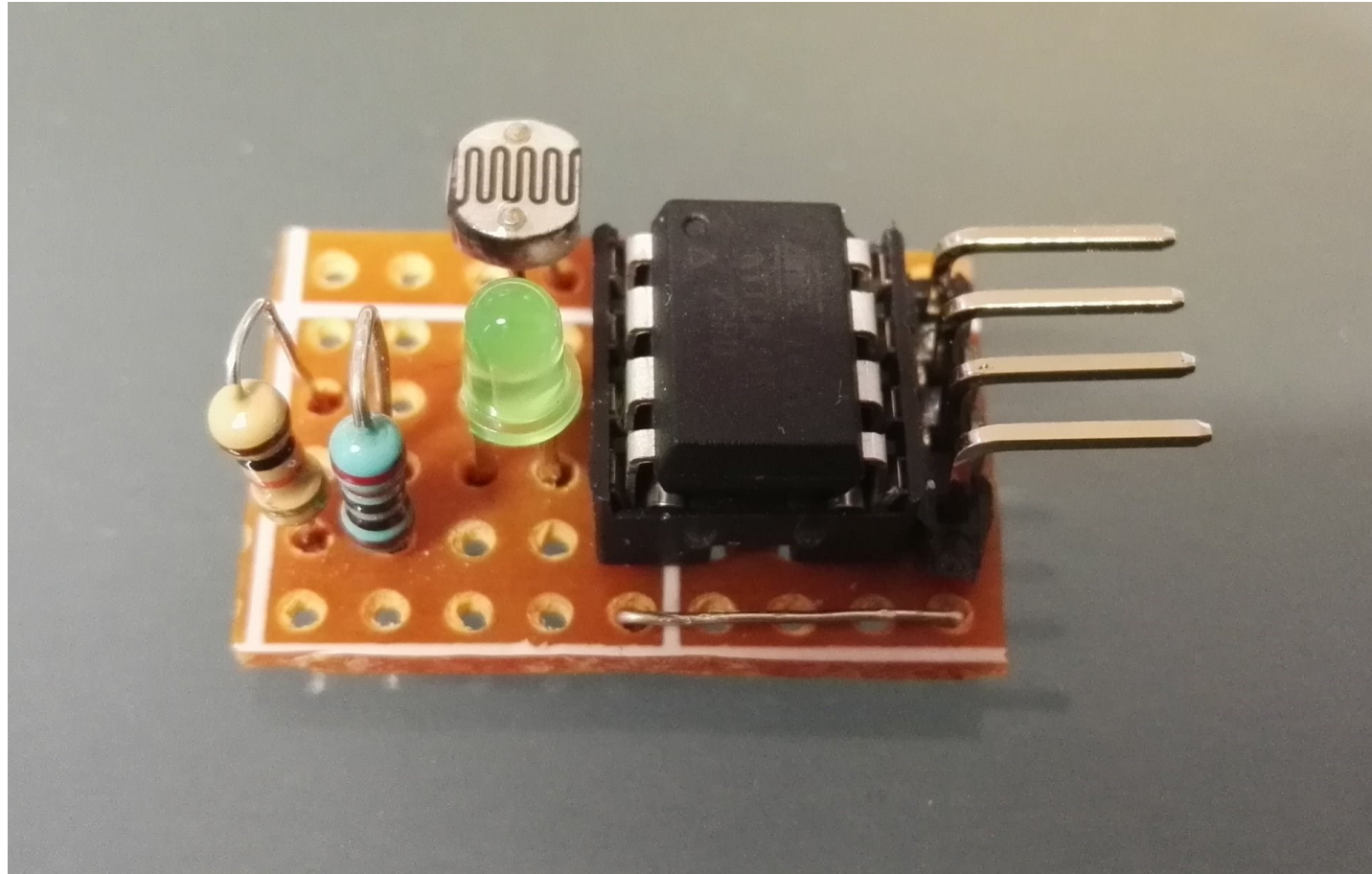
Programarea micro – controllerului ATtiny prin intermediul platformei Arduino



Programarea micro – controllerului ATTiny utilizând Arduino IDE



Exemplu de aplicație – senzor de lumină I2C



<https://quadmeup.com/attiny85-light-sensor-i2c-slave-device/>

Puncte de reper

- ❖ **Prezentarea micro – controllerului integrat ATTiny 45;**
- ❖ Familiarizarea cu conceptul de „modulație în lățime a impulsului”;
- ❖ Familiarizarea cu conceptele de „circuit de forță și comandă”;
- ❖ Concretizarea aplicației propuse (implementare);

Puncte de reper

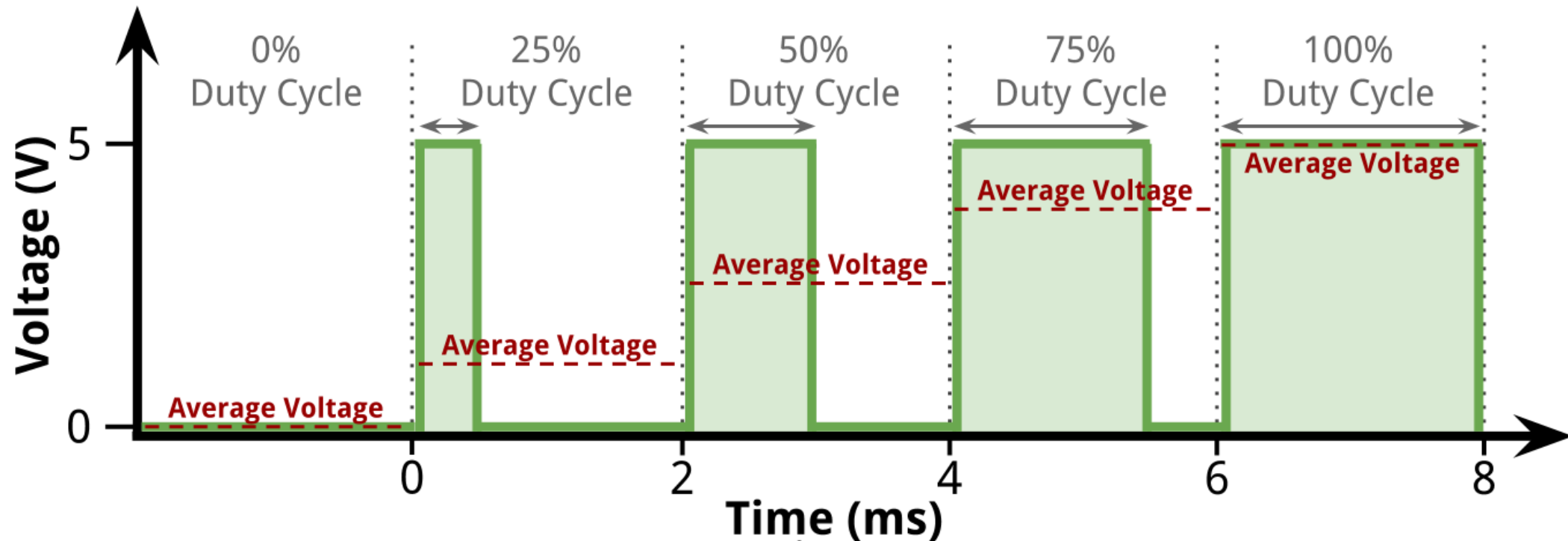
- ✓ **Prezentarea micro – controllerului integrat ATTiny 45;**
- ❖ **Familiarizarea cu conceptul de „modulație în lățime a impulsului”;**
- ❖ Familiarizarea cu conceptele de „circuit de forță și comandă”;
- ❖ Concretizarea aplicației propuse (implementare);

Modularea în lățime a impulsurilor

- Presupune generarea unui tren de impulsuri dreptunghiulare;
- **Frecvența și amplitudinea** semnalului este **CONSTANTĂ**;
- Durata impulsului (sau lățimea) se modifică;
- Frecvența fiind destul de mare, nu provoacă senzația discontinuității;
- Tensiunea medie variază proporțional cu factorul de umplere;
- Efectul constă în variația mărimii reglate (ex. intensitate luminoasă);
- Astfel se poate realiza un reglaj fin în mod **DIGITAL / NUMERIC**.

Modularea în lățime a impulsurilor

Pulse Width Modulation Duty Cycles



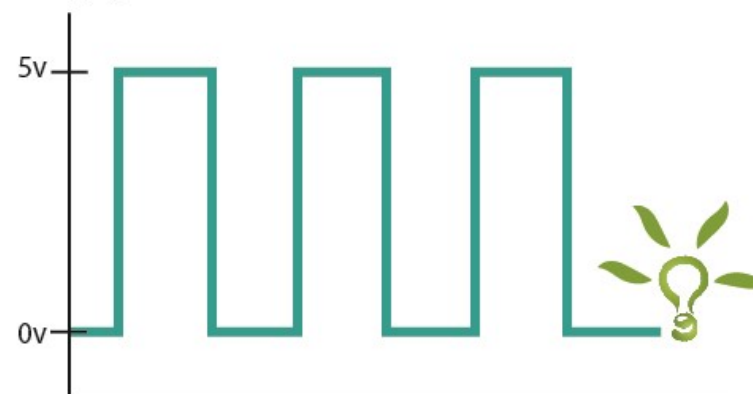
Note: 1 cycle = 2ms @ 500 Hz

Modularea în lățime a impulsurilor

20% Duty cycle



50% Duty cycle



80% Duty cycle

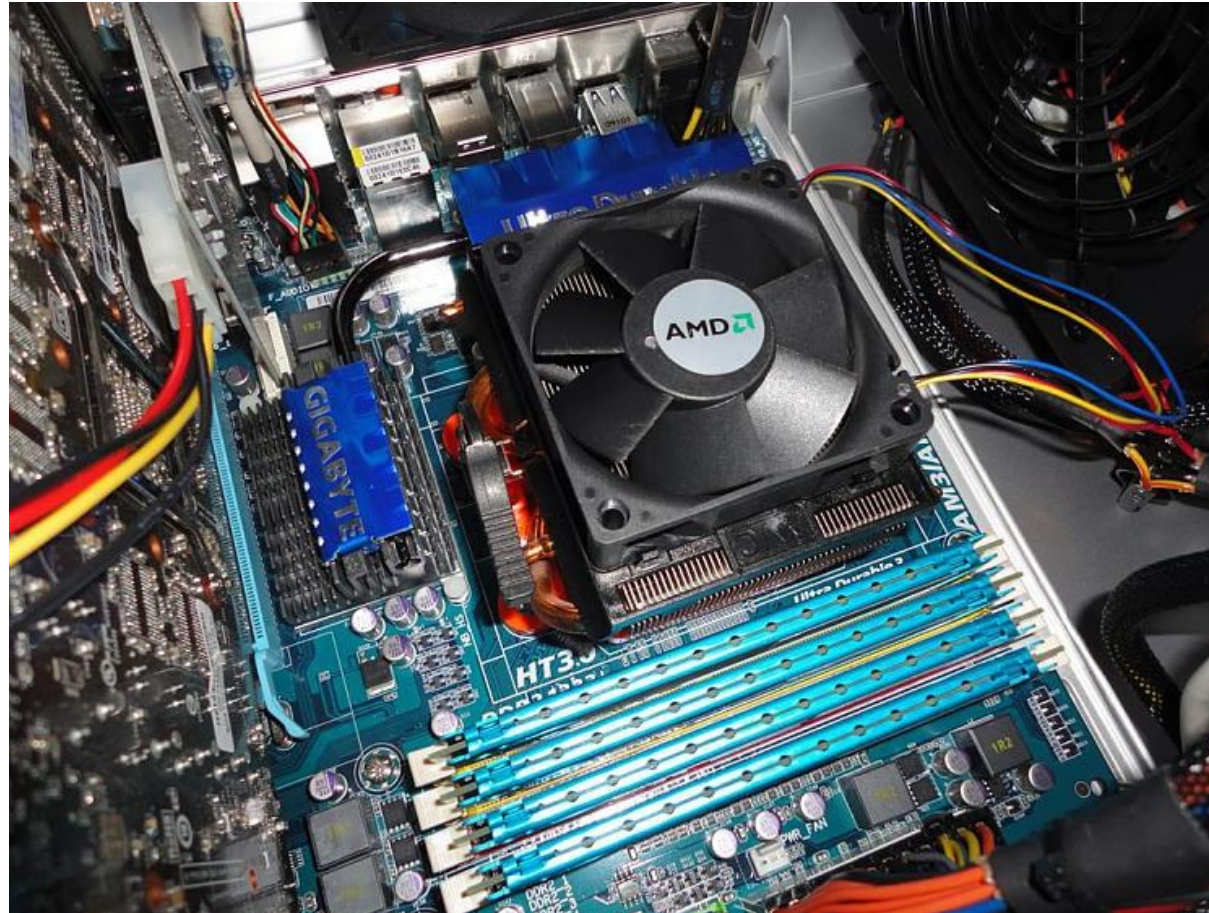


Exemplu de aplicație – lampa ecranului



<http://www.laptoprepair101.com/troubleshooting-laptop-with-backlight-failure/>

Exemplu de aplicație – turația ventilatorului



Puncte de reper

- ✓ **Prezentarea micro – controllerului integrat ATtiny 45;**
- ❖ **Familiarizarea cu conceptul de „modulație în lățime a impulsului”;**
- ❖ Familiarizarea cu conceptele de „circuit de forță și comandă”;
- ❖ Concretizarea aplicației propuse (implementare);

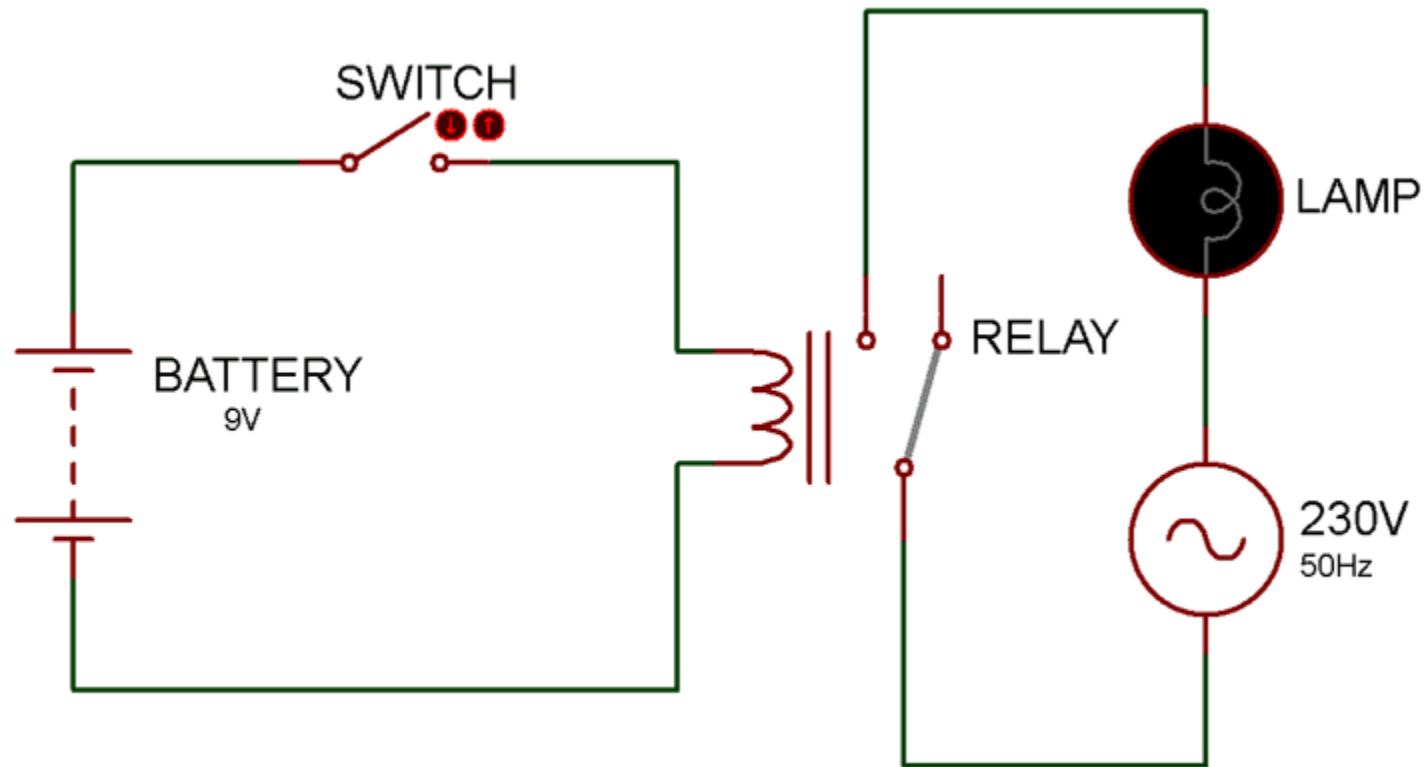
Puncte de reper

- ✓ **Prezentarea micro – controllerului integrat ATTiny 45;**
- ✓ **Familiarizarea cu conceptul de „modulație în lățime a impulsului”;**
- ❖ **Familiarizarea cu conecptele de „circuit de forță și comandă”;**
- ❖ Concretizarea aplicației propuse (implementare);

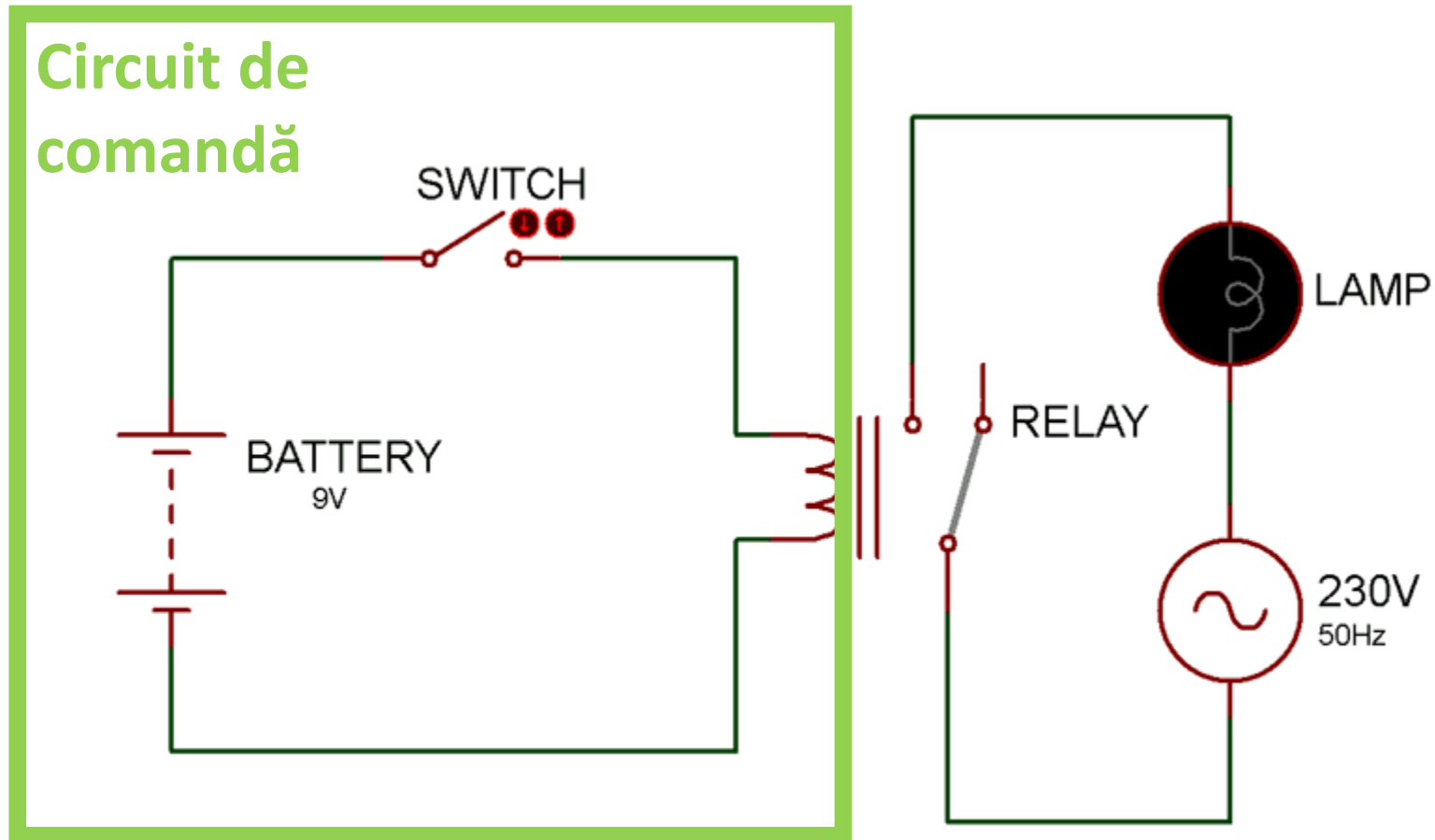
Circuit de forță și circuit de comandă

- Circuitul de forță, reprezintă un etaj dintr-o topologie complexă, în care se vehiculează semnale de putere mare (ex. cureți, tensiuni);
- Circuitul de comandă, reprezintă un etaj dintr-o topologie complexă, care are ca scop, furnizarea / vehicularea unor semnale de curent sau tensiune de valori semnificativ reduse. Rolul semnalelor vehiculate sau produse de / în aceste circuite este de a asigura „Legea de variație” a pentru parametrii semnalului de putere mare / vehiculat în etajul de forță.

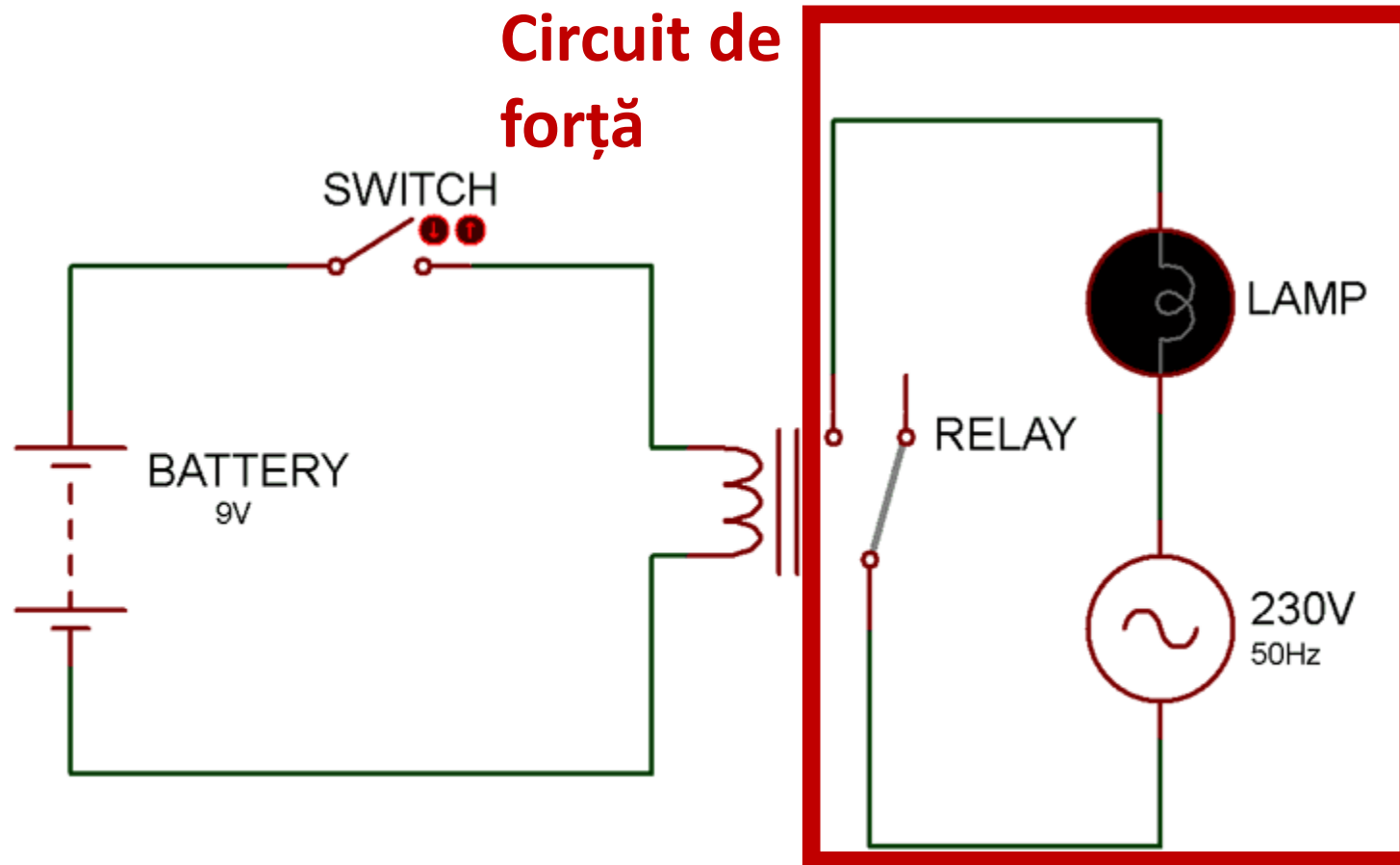
Circuit de forță și circuit de comandă - circuite cu releu -



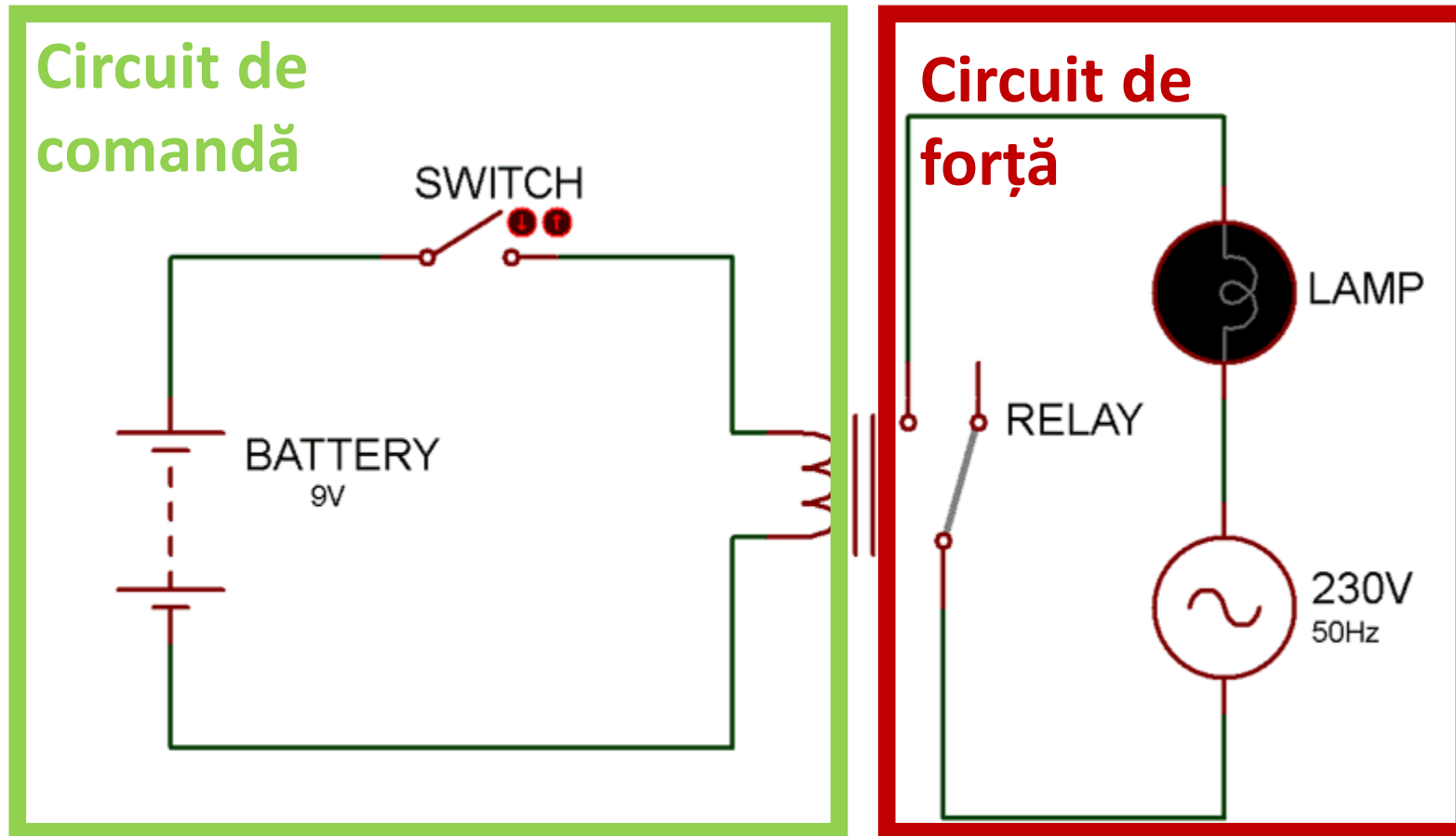
Circuit de forță și circuit de comandă - circuite cu releu -



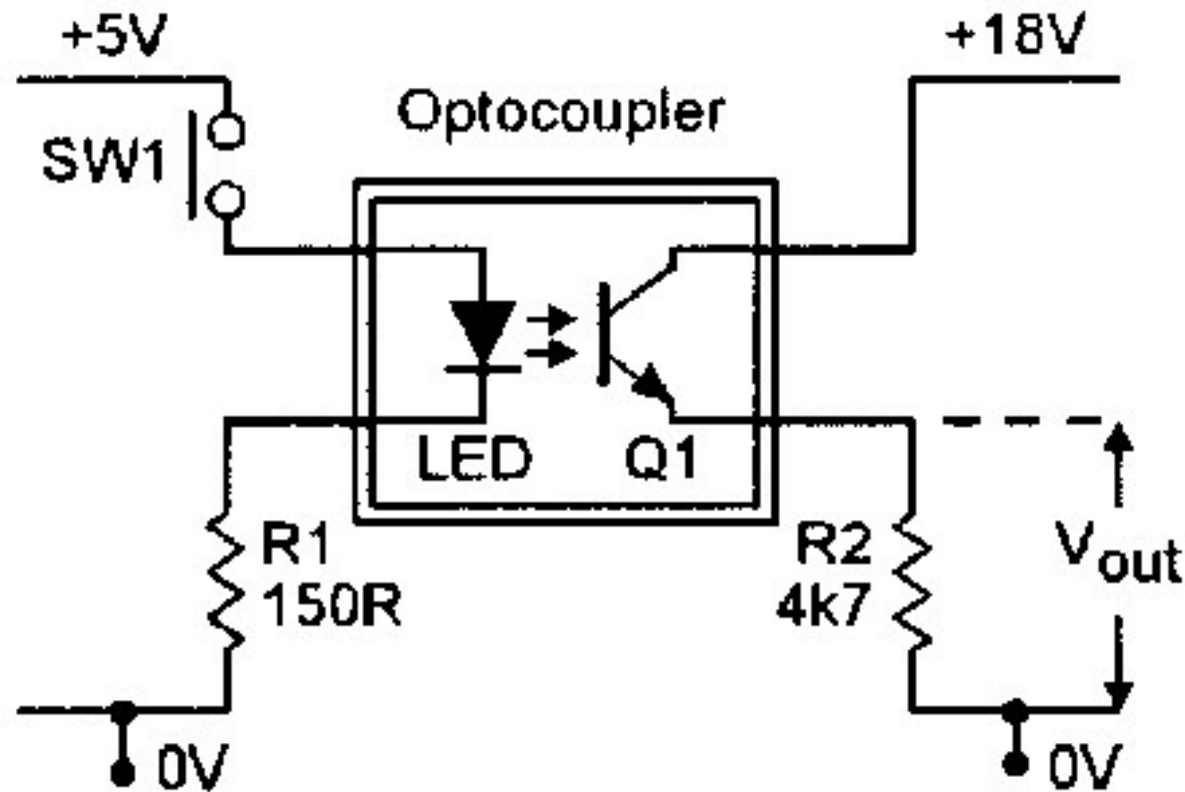
Circuit de forță și circuit de comandă - circuite cu releu -



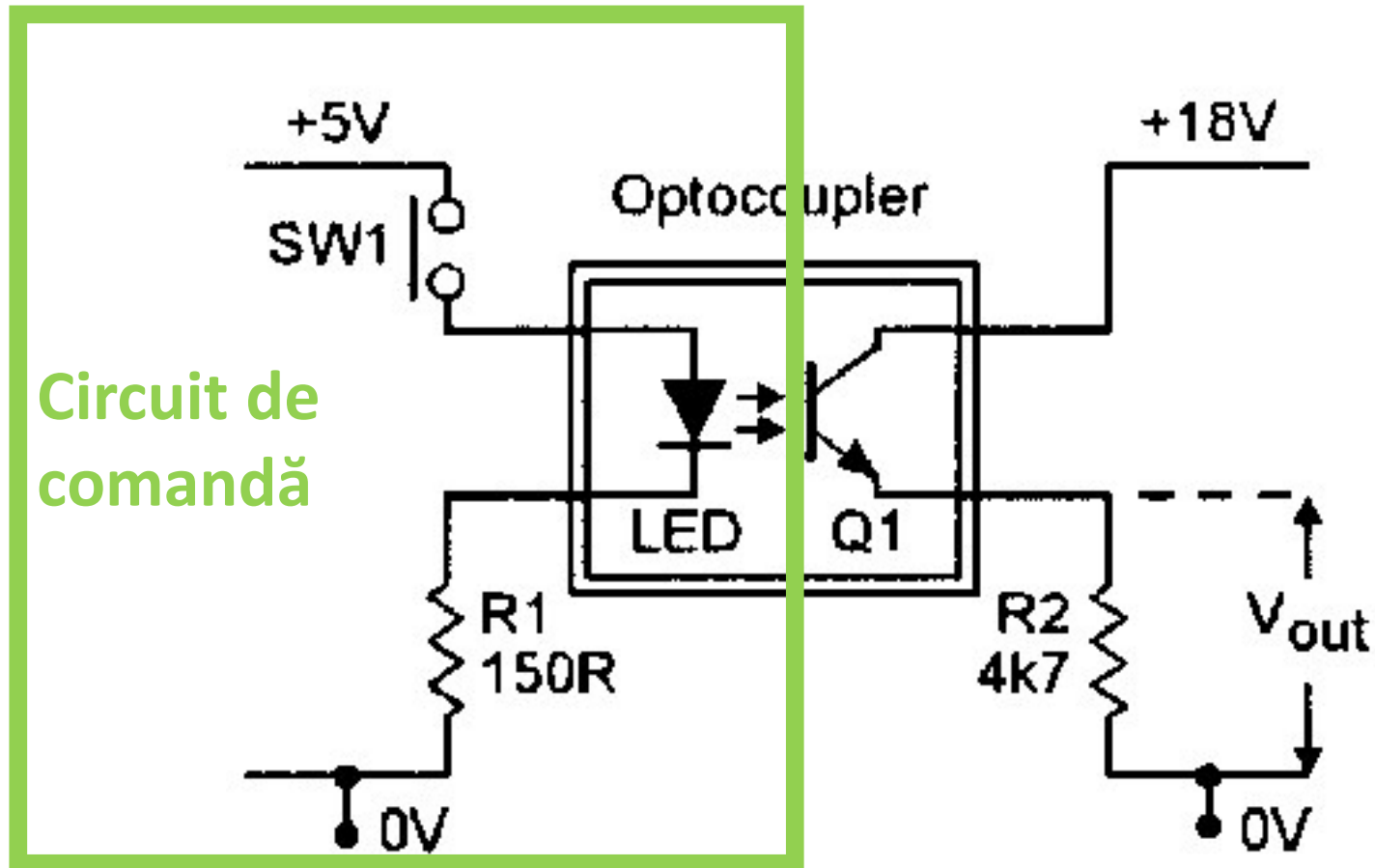
Circuit de forță și circuit de comandă - circuite cu releu -



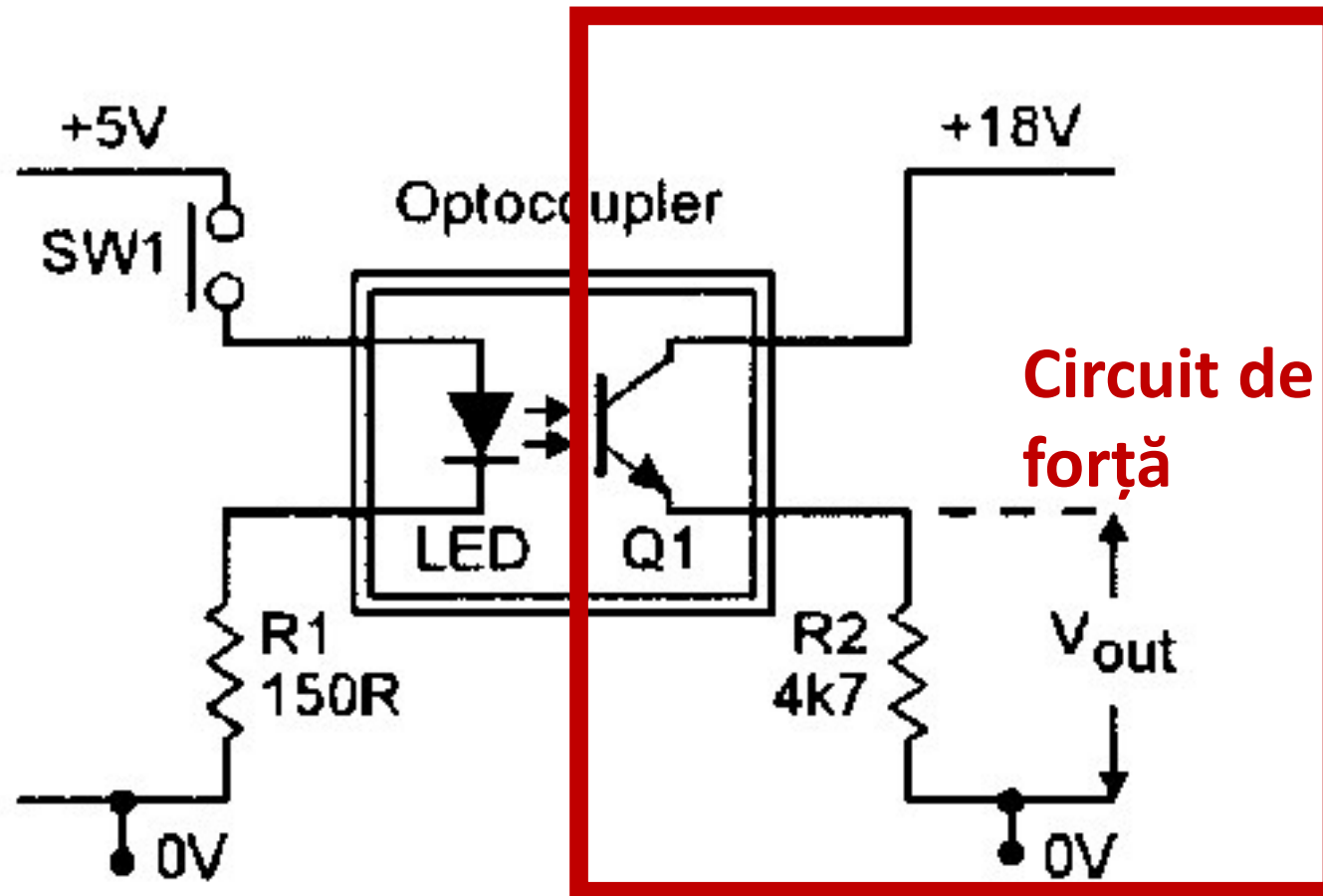
Circuit de forță și circuit de comandă - circuite cu optocuplor -



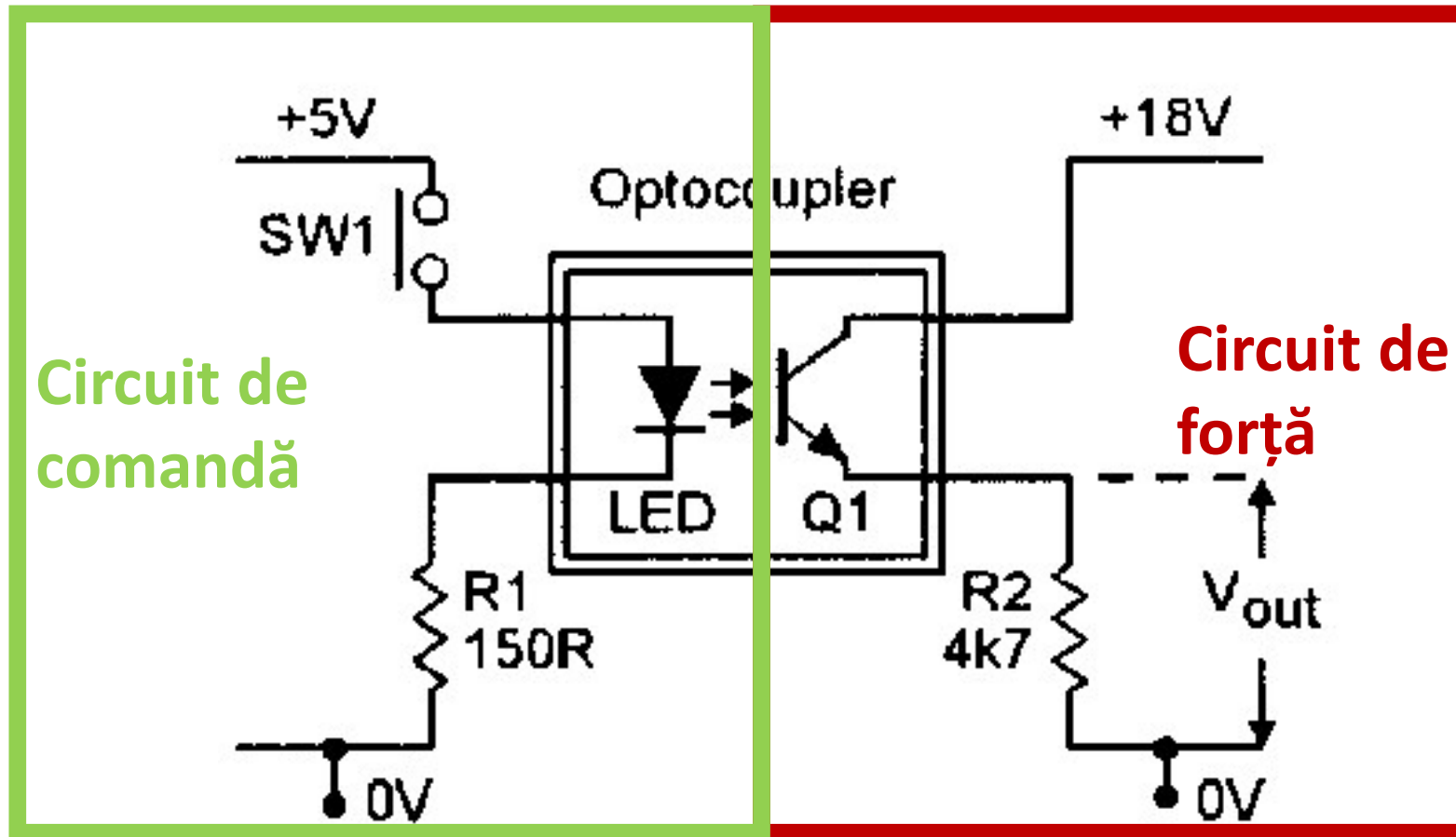
Circuit de forță și circuit de comandă - circuite cu optocuplor -



Circuit de forță și circuit de comandă - circuite cu optocuplor -



Circuit de forță și circuit de comandă - circuite cu optocuplor -



Puncte de reper

- ✓ **Prezentarea micro – controllerului integrat ATTiny 45;**
- ✓ **Familiarizarea cu conceptul de „modulație în lățime a impulsului”;**
- ❖ **Familiarizarea cu conecptele de „circuit de forță și comandă”;**
- ❖ Concretizarea aplicației propuse (implementare);

Puncte de reper

- ✓ **Prezentarea micro – controllerului integrat ATTiny 45;**
- ✓ **Familiarizarea cu conceptul de „modulație în lățime a impulsului”;**
- ✓ **Familiarizarea cu conceptele de „circuit de forță și comandă”;**
- ❖ **Concretizarea aplicației propuse (implementare).**

Componente necesare

- Programator - SparkFun USBTinyISP – TinyAVR;
- Placă pentru test / prototip (eng. Bread Board);
- Micro – controller ATtiny 45 sau 85;
- Un optocuplor A3120 sau alt model echivalent;
- Tranzistor MOSFET IRF640N;
- Stabilizator liniar de tensiune L7805CV;
- Două condensatoare electrolitice având capacitatea între 1 – 10 [μ F];
- Două butoane cu revenire;
- Diodă electro – luminiscentă (L.E.D.)
- Rezistențe (3 x 10 [k Ω], 2 x 100 [Ω], 1 x 22 [Ω]);
- Fire pentru conexiune;
- Cleme cu șurub pentru prinderea firelor;
- Diverse sarcini (ex. lampă cu incandescență sau ventilator de computer BLDC);

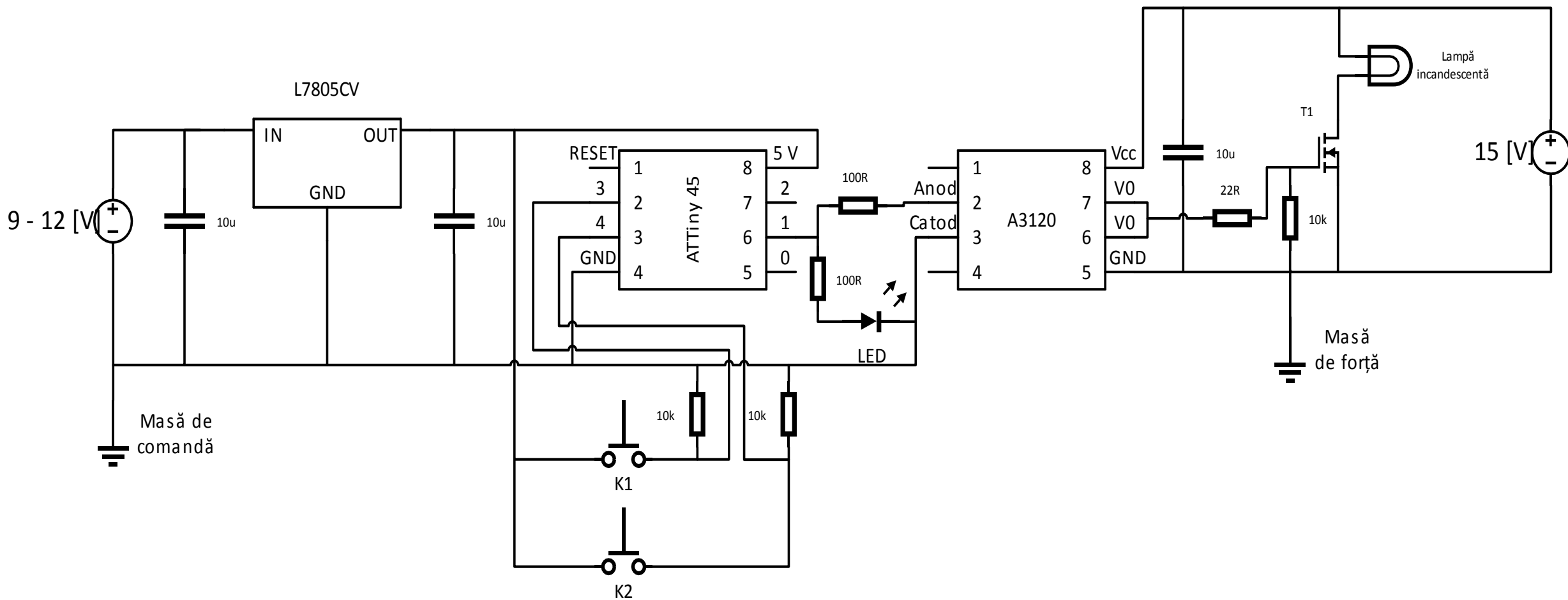
Aparatură necesară

- Sursă de tensiune reglabilă;
- Osciloscop digital;
- Aparat de măsură
- Computer personal;

Programe și fișiere necesare

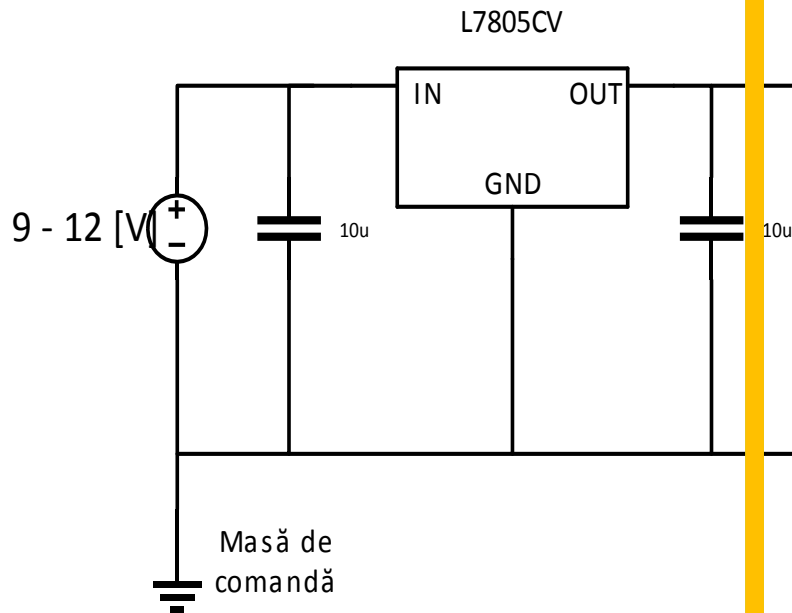
- Mediul de programare și testare Arduino IDE;
- Biblioteca și compilatorul pentru ATTiny 45 / 85 etc...;
- Program de instalare al programatorului (eng. driver);

Schema generală

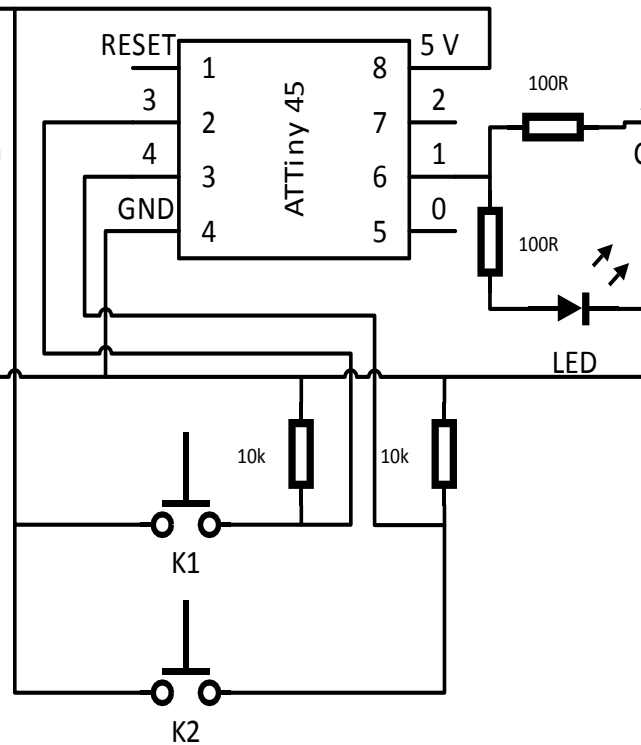


Schema generală

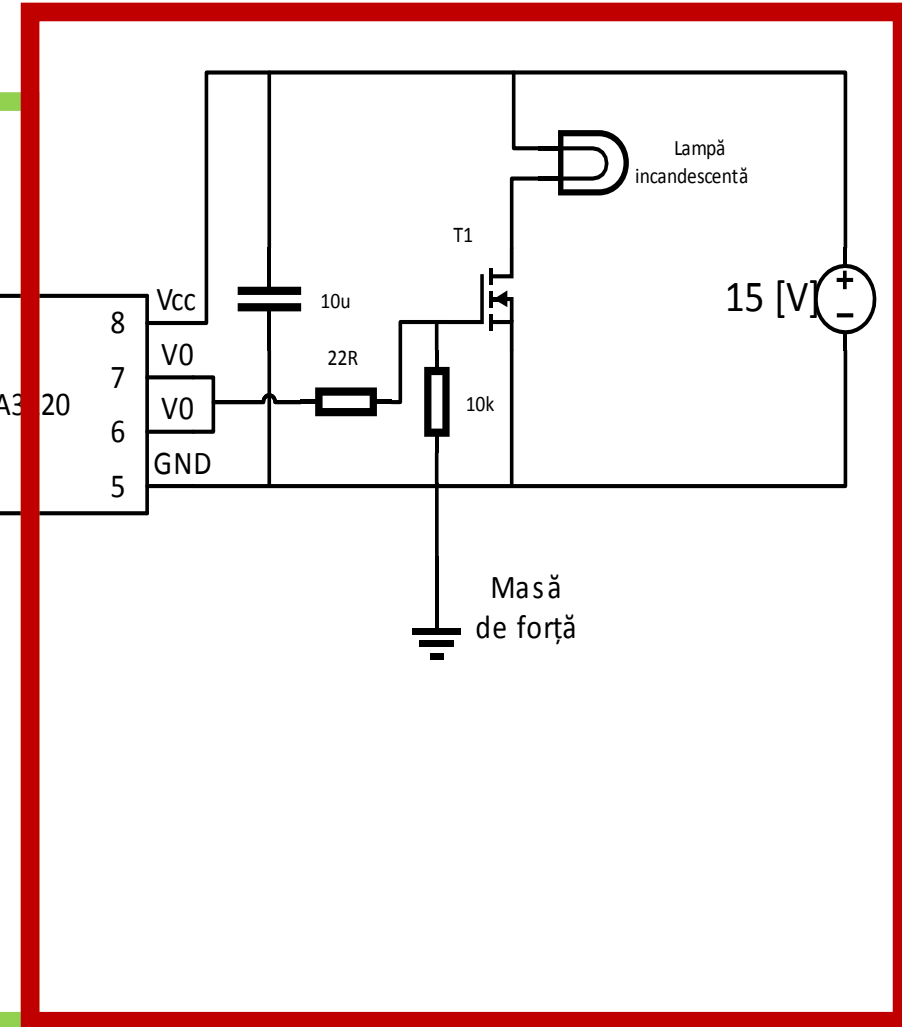
Alimentare circuit de comandă



Circuit de comandă

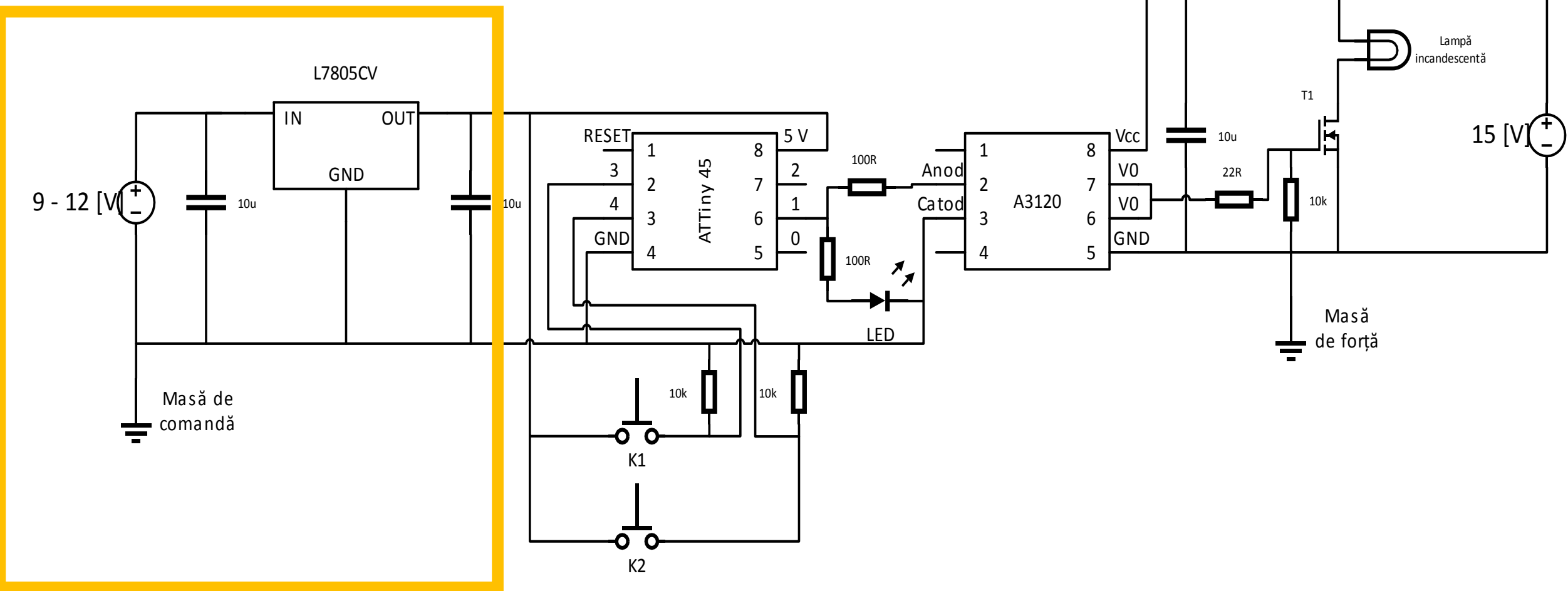


Circuit de forță

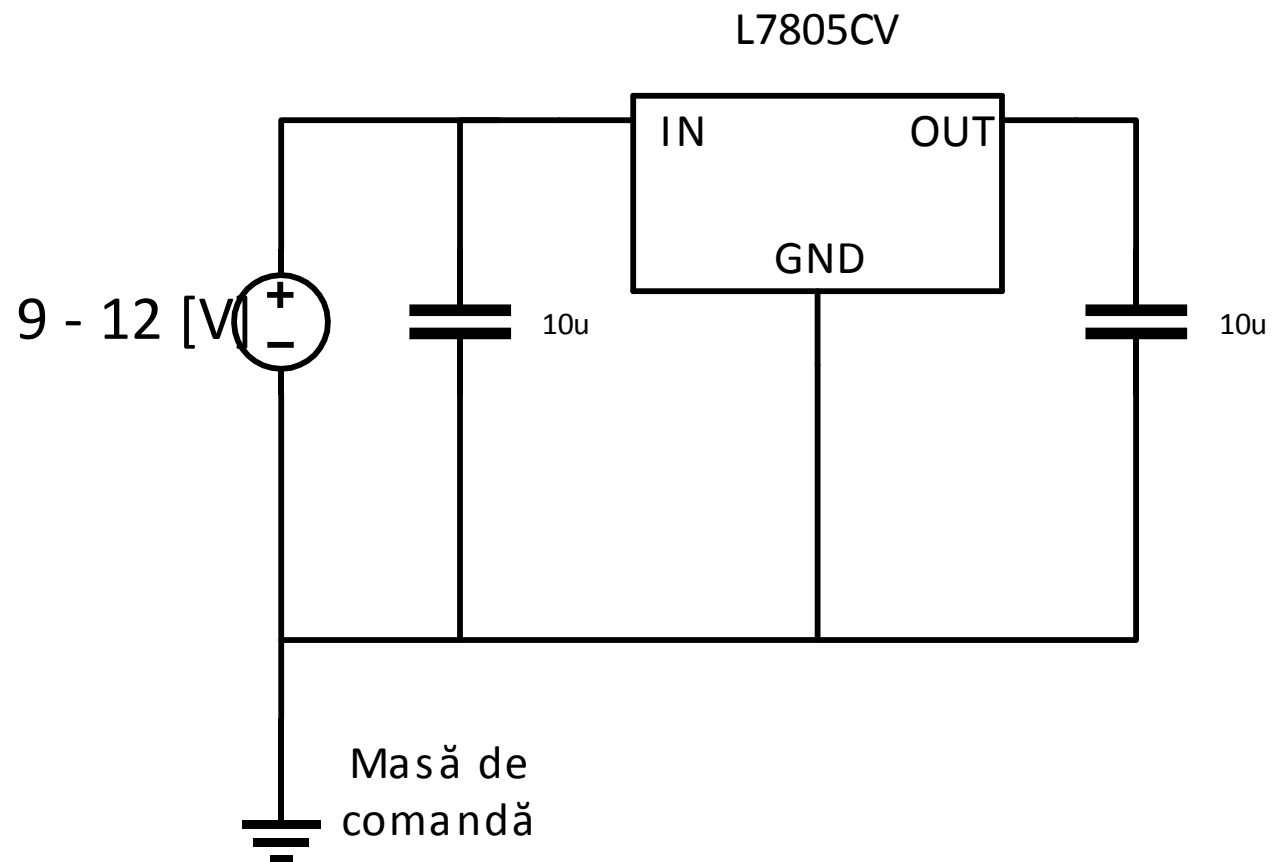


Schema generală

Alimentare circuit de comandă

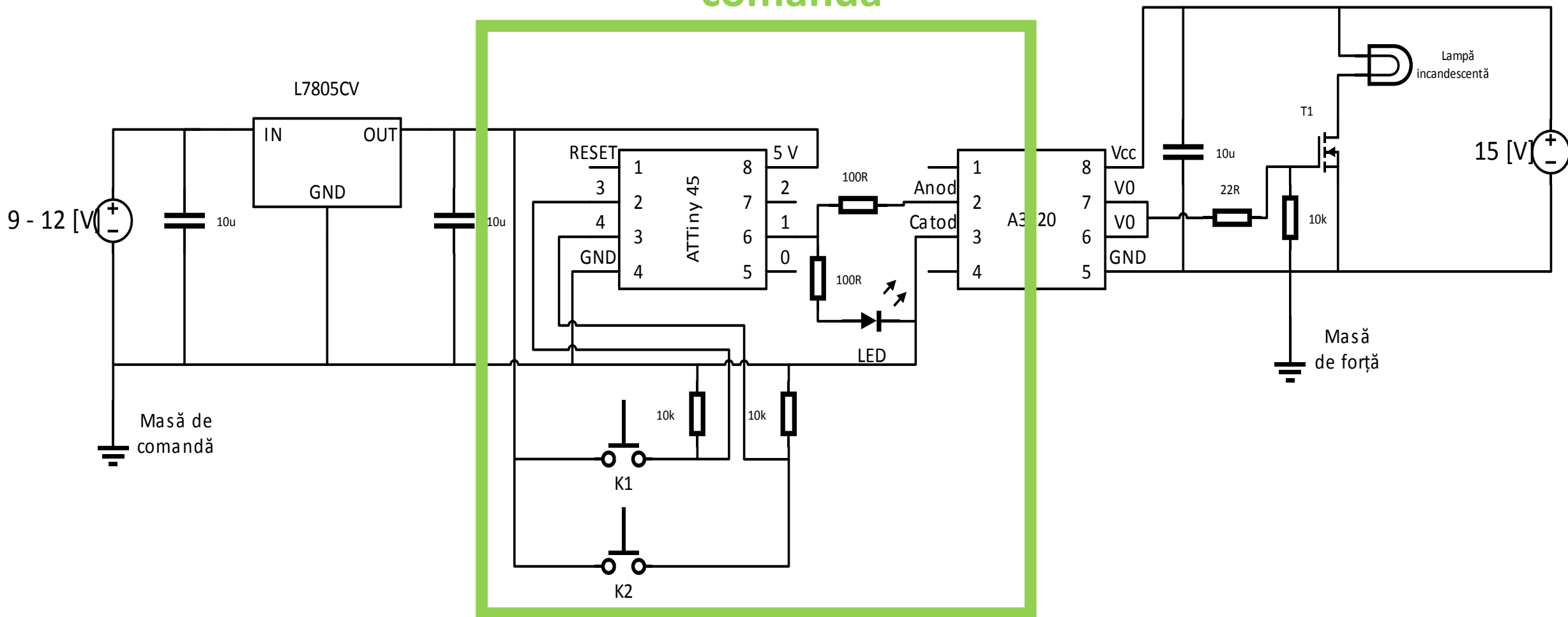


Alimentarea circuitului de comandă

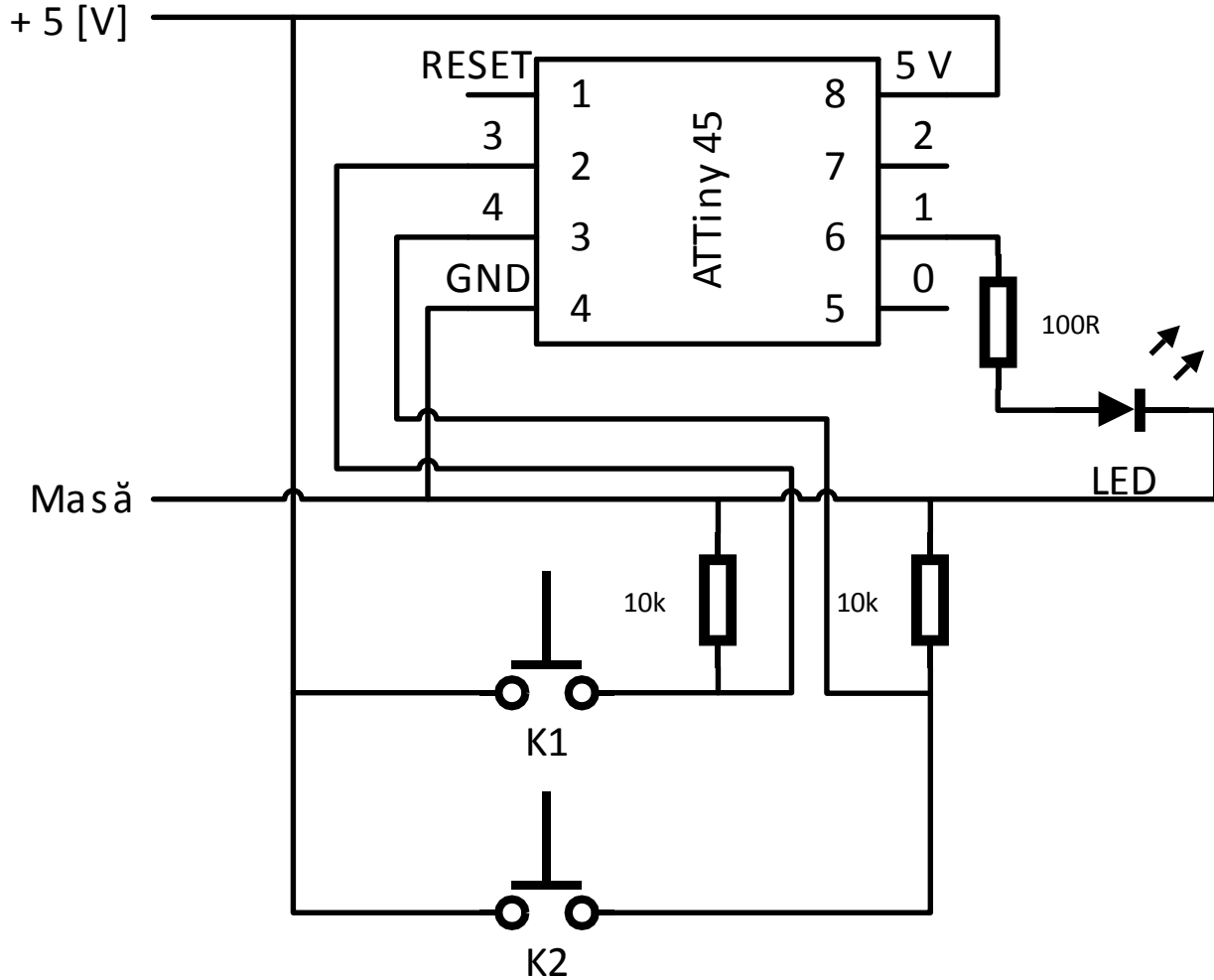


Schema generală

Circuit de comandă



Circuitul de comandă



Implementarea logicii de comandă

Secvențe de cod – declarații inițiale

```
const int pwm_pin = 0;
```

```
int factor_de_umplere = 0;
```

```
int intarziere = 10;
```

```
const int buton_crescator = 3;
```

```
const int buton_descrescator = 4;
```

```
int stare_buton_crescator = 0;
```

```
int stare_buton_descrescator = 0;
```

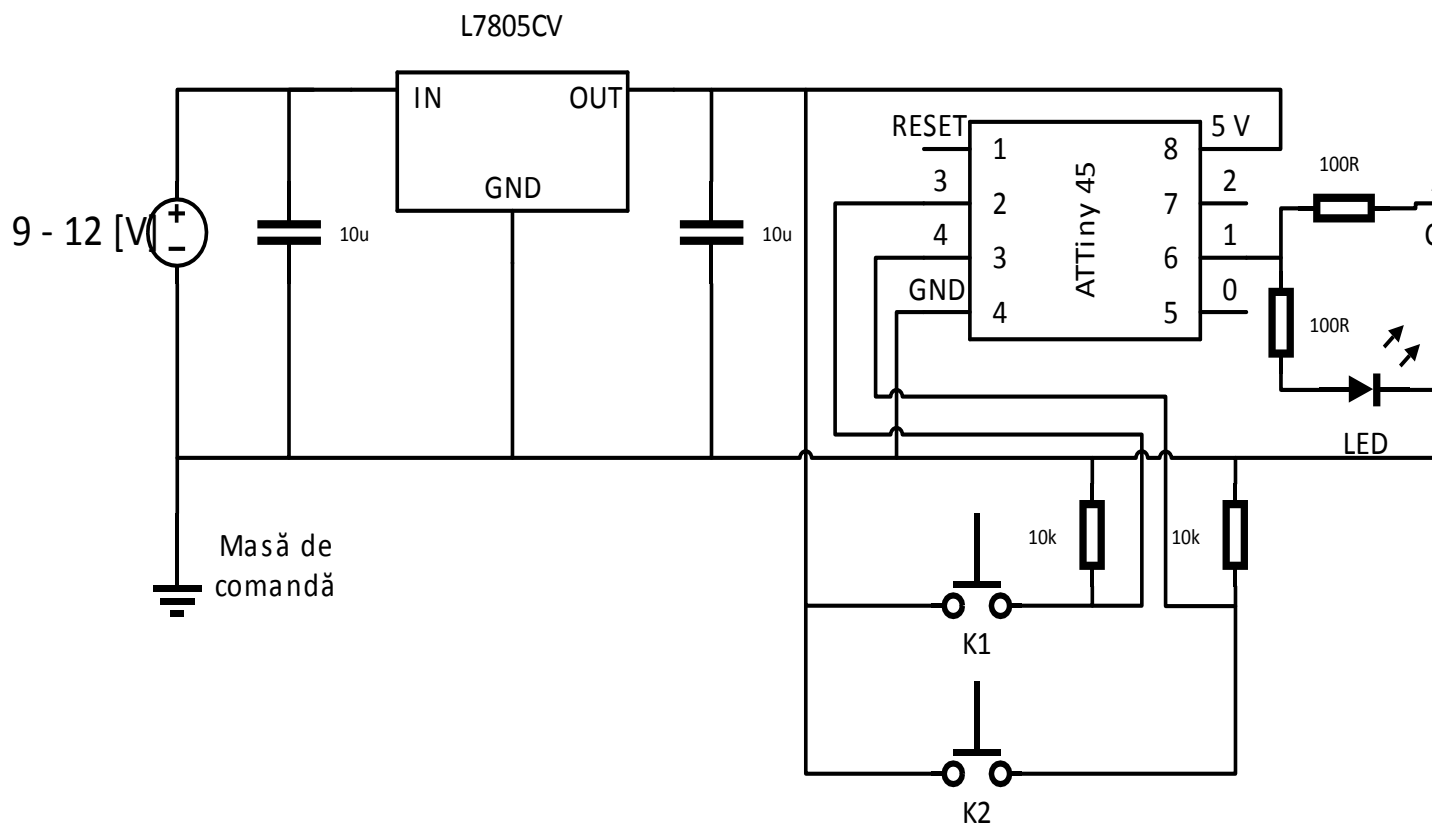
Secvențe de cod – bucla de inițializare

```
void setup() {  
  pinMode(buton_crescator, INPUT);  
  pinMode(buton_descrescator, INPUT);  
  pinMode(pwm_pin, OUTPUT);  
}
```

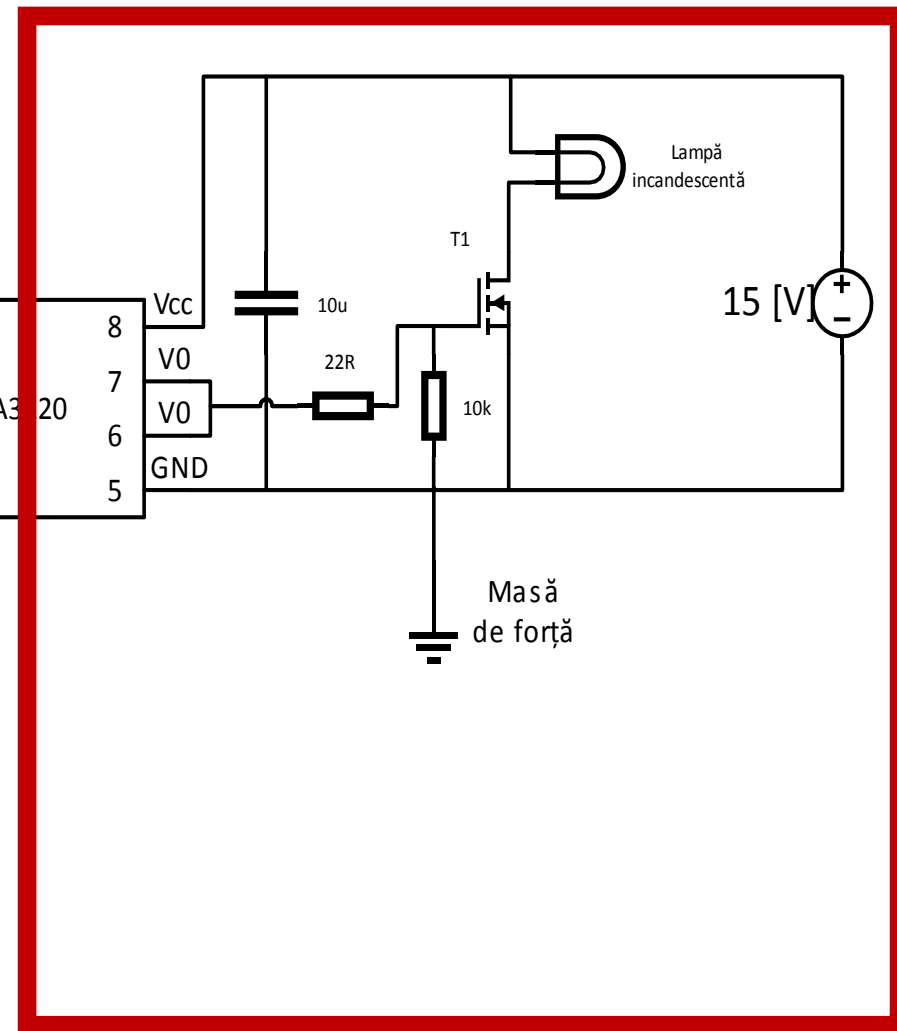
Secvențe de cod – bucla infinită de program

```
void loop() {  
  stare_buton_crescator = digitalRead(buton_crescator);  
  if (stare_buton_crescator == HIGH) factor_de_umplere++;  
  if (factor_de_umplere >= 255) factor_de_umplere = 255;  
  
  stare_buton_descrescator = digitalRead(buton_descrescator);  
  if (stare_buton_descrescator == HIGH) factor_de_umplere--;  
  if (factor_de_umplere <= 0) factor_de_umplere = 0;  
  
  analogWrite(pwm_pin, factor_de_umplere);  
  delay(intarziere);  
}
```

Schema generală



Circuit de forță



Circuitul de forță

