



Elegoo Super Starter Kit pentru UNO

V1.0.18.12.18

Prefață

Compania noastră

Fondată în 2011, Elegoo Inc. este o companie tehnologică înfloritoare care cercetează și dezvoltă, produce și comercializează hardware open source. Avem sediul central în Silicon Valley din China, în Shenzhen, și am ajuns la peste 150 de angajați cu o fabrică de peste 10.763 de metri pătrați.

Liniile noastre de produse variază de la fire DuPont, plăci UNO R3 până la kituri complete pentru începători concepute pentru clienții de orice nivel care doresc să învețe cum să utilizeze Arduino. În plus, vindem și produse accesorii pentru Raspberry Pi, cum ar fi TFT tactil de 2,8" și STM32. În viitor, am dori să dedicăm mai multă energie și investiții produselor pentru imprimantele 3D, etc. Toate produsele noastre îndeplinesc standardele internaționale de calitate și sunt foarte apreciate pe diverse piețe din întreaga lume.

Site-ul oficial: <http://www.elegoo.com>

US Amazon storefront: <http://www.amazon.com/shops/A2WWHQ25ENKVJ1>

CA Amazon storefront: <http://www.amazon.ca/shops/A2WWHQ25ENKVJ1>

UK Amazon storefront: <http://www.amazon.co.uk/shops/AZF7WYXU5ZANW>

DE Amazon storefront: <http://www.amazon.de/shops/AZF7WYXU5ZANW>

FR Amazon storefront: <http://www.amazon.fr/shops/AZF7WYXU5ZANW>

ES Amazon storefront: <http://www.amazon.es/shops/AZF7WYXU5ZANW>

IT Amazon storefront: <http://www.amazon.it/shops/AZF7WYXU5ZANW>

Programul nostru de predare

Acest ghid este pentru începători. Veți afla toate informațiile de bază despre cum să utilizați placa de control, senzorii și componentele Arduino. Dacă doriți să studiați Arduino mai în profunzime, vă recomandăm să citiți Arduino Cookbook scris de Michael Margolis.

O parte din codul din acest manual a fost editat de Simon Monk. Simon Monk este autorul unui număr de cărți despre hardware open source. Acestea sunt disponibile pe Amazon: Cărțile lui Monk includ Programming the Arduino, 30 Arduino Projects for Evil Geniuses și Programming the Raspberry Pi.

Serviciul clienți

Fiind o companie tehnologică în continuă și rapidă creștere, ne străduim să facem tot ce ne stă în putință.

Oferim produse excelente și servicii de calitate pentru a vă satisface așteptările și ne puteți contacta simplu la service@elegoo.com sau EUservice@elegoo.com. Așteptăm cu nerăbdare să ne contactați și orice comentarii critice sau sugestii ar fi foarte valoroase pentru noi.

Tehnicienii noștri experimentați vor răspunde la toate problemele și întrebările despre produsele noastre

va răspunde imediat în termen de 12 ore (24 de ore în perioada sărbătorilor).

Motor pas cu pas

1KS



Servomotor (SG90)

1KS



Receptor IR Modul

1KS



5V Rel ay

1KS



Placa unității de control Uno R3 1KS



Modul LCD 1602 (cu conector)

1KS



Controler pas cu pas ULN2003

Conducătorul motorului

Mod ul e
1KS



Sup ply de putere

Stupi Mod
1KS



Extensie prototip

Modul
1KS



Contactați-ne :
service@elegoo.com



4 cifre 7 - segment
Afişare
1KS



1 cifră 7
segmente
Vizualizare
1KS



Comutator de
înclinare
bilă
1KS



Paleta
ventilatorului
i şi



Modul joystick
1KS

3-6 V DC
Motor
(cu fir)
fiecare
1KS



DHT11 Temperatura
şi umiditate
Modul
1KS



Senzor cu ultrasunete
1KS



Contactaţi-ne :
service@elegoo.com

Buton (mic)
SPCS



Potențiometru
1KS



Buzzer pasiv
1KS



Buzzer activ
1KS



Baterie 9V cu
conector de
conectare 1 buc



La distanță
1KS



Cablu USB
1KS



Transformarea unei
femei în bărbat
- Fir Dupont
10KS



Breadboard Jumper
Wire 65pcs



74HC595 IC
1KS



L293 D



830 de puncte
de legătură
Placă de
pâine
1KS



Contactați-ne: service@elegoo.com

Therrriistor
1KS



Diodă redresoare
2KS

Fotorezistor
Photobooth
2KS

LED
25PCS



Rezistor
20 PIECES



RGB
2KS

Tranzistor NPN
PN 2222
2KS



Contactați-ne : service@elegoo.com

Tabla de conținut

Lecția 0 Instalarea IDE-ului	9
Lecția 1 Adăugarea bibliotecilor și deschiderea monitorului serial	22
Lecția 2 Clipa	31
Lecția 3 LED	39
Lecția 4 LED-uri RGB	46
Lecția 5 Intrări digitale	55
Lecția 6 Buzzer activ	60
Lecția 7 Buzzer pasiv	64
Lecția 8 Comutator cu bilă basculantă	68
Lecția 9 Servo	72
Lecția 10 Modul senzor cu ultrasunete.....	76
Lecția 11 Senzor de temperatură și umiditate DHT11.....	81
Lecția 12 Modul Joystick analogic	87
Lecția 13 Modul receptor IR	92
Lecția 14 Afișaj LCD.....	98
Lecția 15 Termometru.....	103
Lecția 16 Opt LED-uri cu 74HC595	108
Lecția 17 Monitor serial	115
Lecția 18 Photobooth.....	121
Lecția 19 74HC595 și afișajul cu segmente.....	126
Lecția 20 Patru afișaje digitale cu șapte segmente	132
Lecția 21 Motoare de curent continuu	136
Lecția 22 Releu	146
Lecția 23 Motor pas cu pas.....	151
Lecția 24 Controlul unui motor pas cu pas prin telecomandă	159

Lecția 0 Instalarea IDE

Acasă

Arduino Integrated Development Environment (IDE) este partea software a platformei Arduino.

În această lecție veți învăța cum să vă configurați computerul pentru a utiliza Arduino și cum să începeți cu următoarele lecții.

Software-ul Arduino pe care îl veți utiliza pentru a programa Arduino este disponibil pentru Windows, Mac și Linux. Procesul de instalare este diferit pentru toate cele trei platforme și, din păcate, va trebui să instalați manual software-ul.

PASUL 1: Mergeți la <https://www.arduino.cc/en/Main/Software> și găsiți pagina de mai jos.



ARDUINO 1.8.5

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software.

This software can be used with any Arduino board. Refer to the [Getting Started](#) page for Installation instructions.

Windows Installer
Windows ZIP file for non admin install

Windows app [Get](#)

Mac OS X 10.7 Lion or newer

Linux 32 bits
Linux 64 bits
Linux ARM

[Release Notes](#)
[Source Code](#)
[Checksums \(sha512\)](#)

Versiunea disponibilă pe acest site este de obicei cea mai recentă versiune și versiunea reală poate fi mai nouă decât versiunea din imagine.

PASUL 2: Descărcați software-ul de dezvoltare care este compatibil cu sistemul de operare. sistem informatic. Să luăm Windows ca exemplu.



Faceți clic pe *Windows Installer*.

Support the Arduino Software

Consider supporting the Arduino Software by contributing to its development. (US tax payers, please note this contribution is not tax deductible). Learn more on how your contribution will be used.

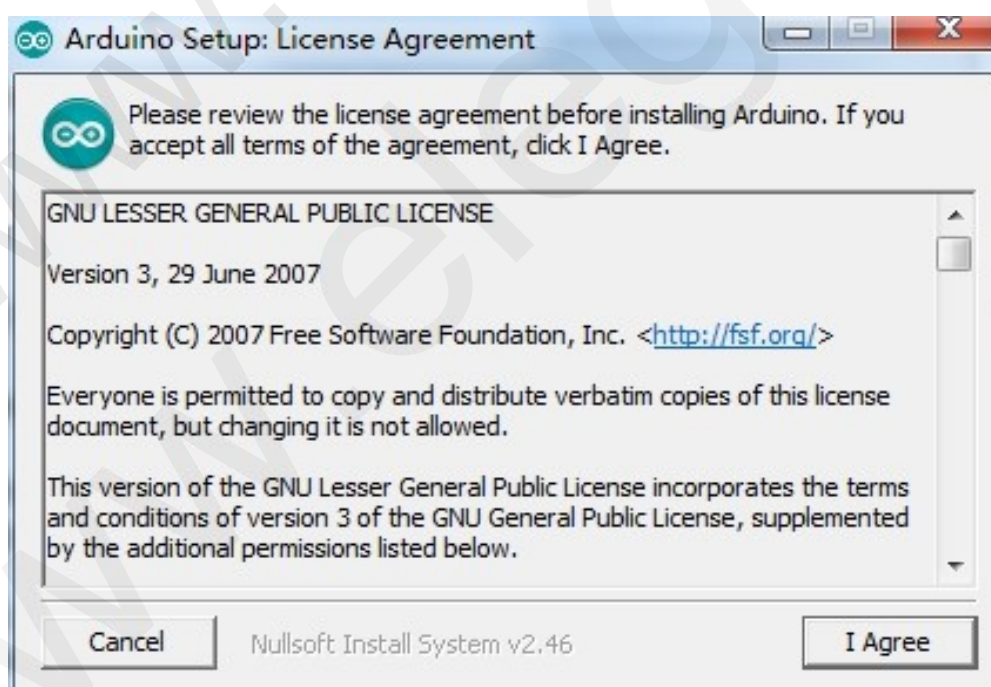


Faceți clic pe *JUSTDOWNLOAD*.

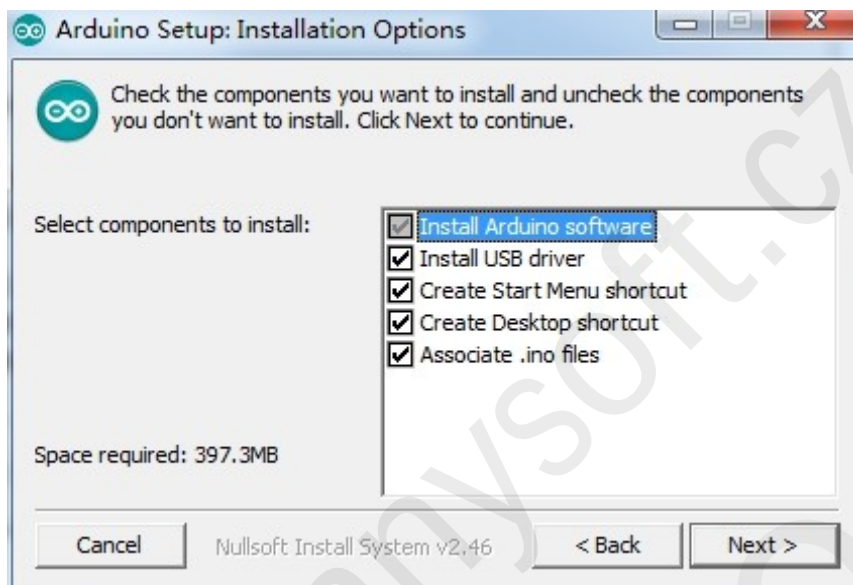
De asemenea, puteți descărca Arduino IDE de pe site-ul nostru oficial <http://www.elegoo.com/download/>.

Instalarea Arduino (Windows)

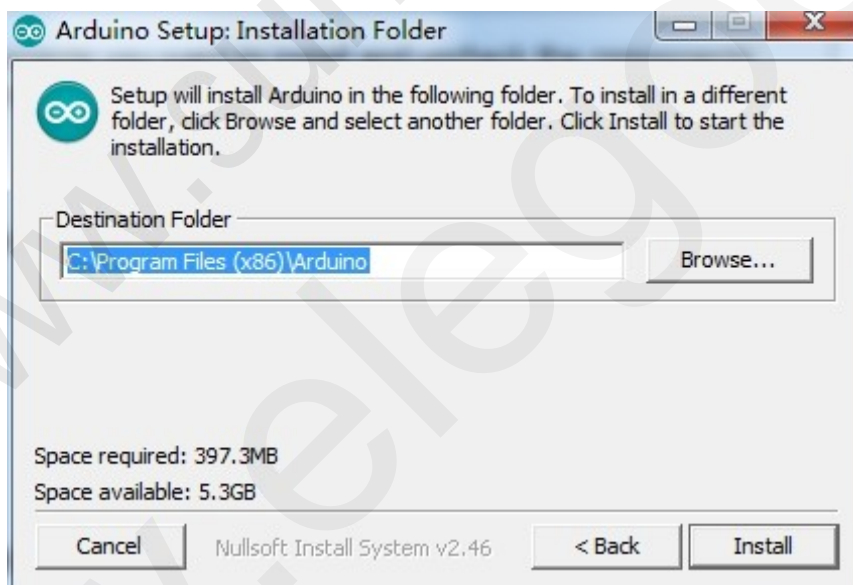
Instalați Arduino utilizând exe. Pachet de instalare.



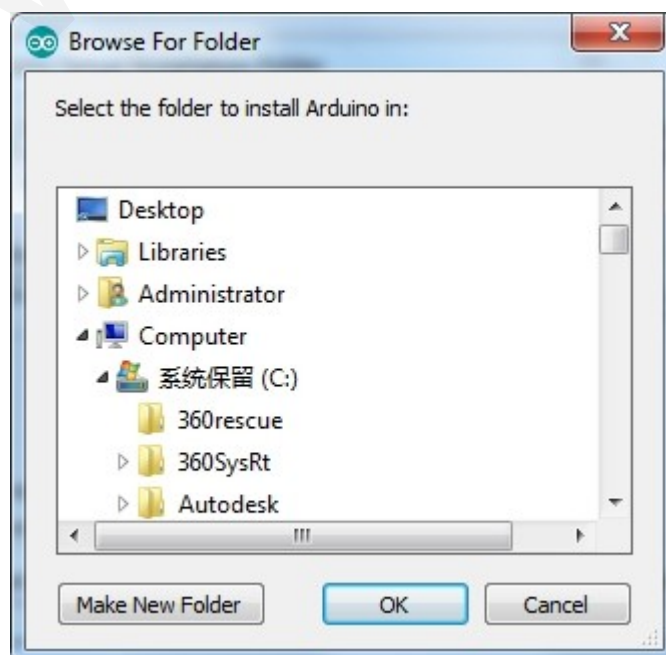
Atunci când faceți clic pe butonul *Sunt de acord*, va apărea următoarea interfață



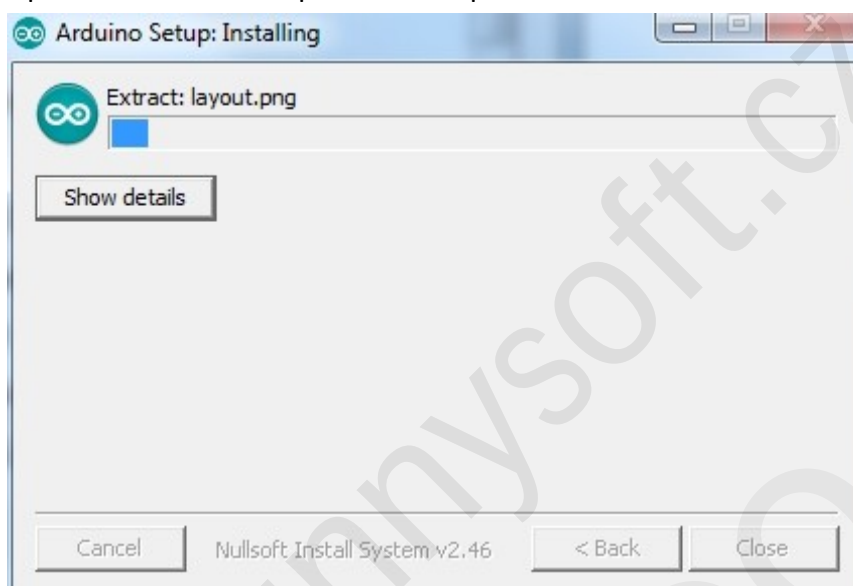
Faceți clic pe *Următorul*



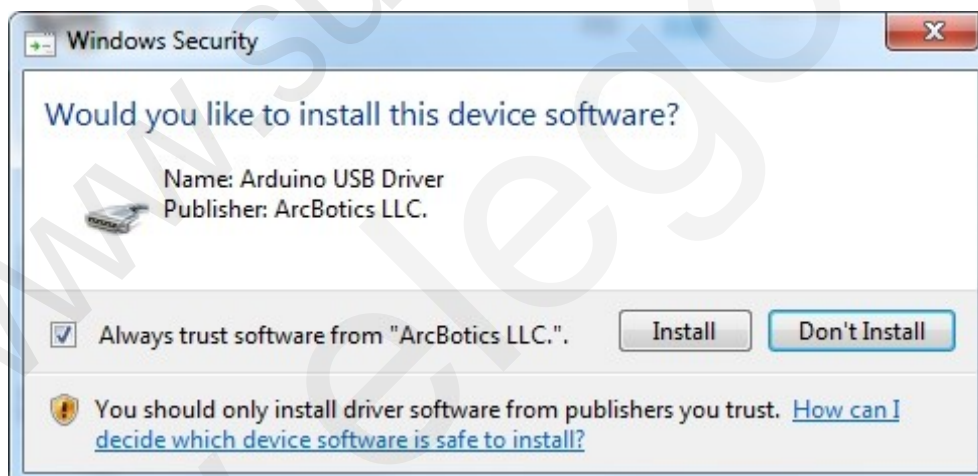
Apăsați butonul **Browse...** pentru a selecta o cale de instalare sau specificați direct directorul dorit.



Faceți clic pe butonul *Instalare* pentru a începe instalarea



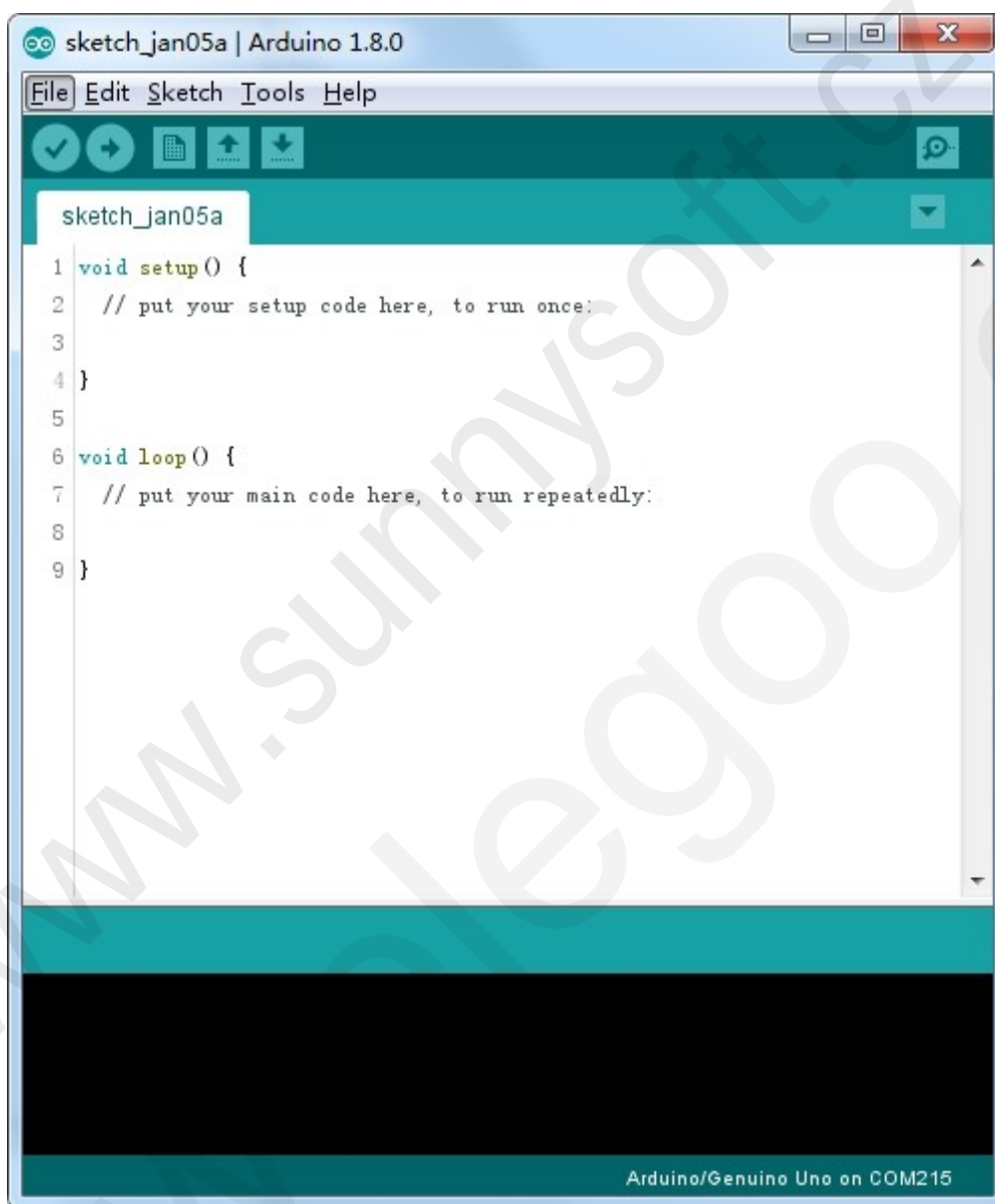
În final, va apărea următoarea interfață, faceți clic pe *Instalare* pentru a finaliza instalarea.



Următoarea pictogramă va apărea apoi pe desktop



Faceți dublu clic pentru a intra în mediul de dezvoltare dorit



Dacă alegeți să instalați APK-ul direct, vă rugăm să săriți peste următorul conținut și să continuați cu lecția următoare. Cu toate acestea, dacă doriți să învățați o metodă de instalare diferită de cea din APK, vă rugăm să citiți acest cuvânt.

Deconectați fișierul zip descărcat, faceți dublu clic pe program și introduceți ceea ce aveți nevoie în IDE.



arduino-1.8.5-windows

此电脑 > Storage (D:) > arduino 1.8.5 > arduino-1.8.5-windows > arduino-1.8.5

名称	修改日期	类型	大小
drivers	2017/12/6 10:15	文件夹	
examples	2017/12/6 10:15	文件夹	
hardware	2017/12/6 10:15	文件夹	
java	2017/12/6 10:15	文件夹	
lib	2017/12/6 10:15	文件夹	
libraries	2017/12/6 10:15	文件夹	
reference	2017/12/6 10:15	文件夹	
tools	2017/12/6 10:15	文件夹	
tools-builder	2017/12/6 10:15	文件夹	
arduino	2017/10/2 15:37	应用程序	395 KB
arduino.l4j	2017/10/2 15:37	配置设置	1 KB
arduino_debug	2017/10/2 15:37	应用程序	393 KB
arduino_debug.l4j	2017/10/2 15:37	配置设置	1 KB
arduino-builder	2017/10/2 15:37	应用程序	3,214 KB
libusb0.dll	2017/10/2 15:37	应用程序扩展	43 KB
msvcp100.dll	2017/10/2 15:37	应用程序扩展	412 KB
msvcr100.dll	2017/10/2 15:37	应用程序扩展	753 KB
revisions	2017/10/2 15:37	文本文档	84 KB
wrapper-manifest	2017/10/2 15:37	XML 文档	1 KB

 **arduino**

sketch_dec06a | Arduino 1.8.5

File Edit Sketch Tools Help

sketch_dec06a

```

1 void setup() {
2   // put your setup code here, to run once:
3
4 }
5
6 void loop() {
7   // put your main code here, to run repeatedly:
8
9 }

```

Arduino/Genuino Uno on COM3

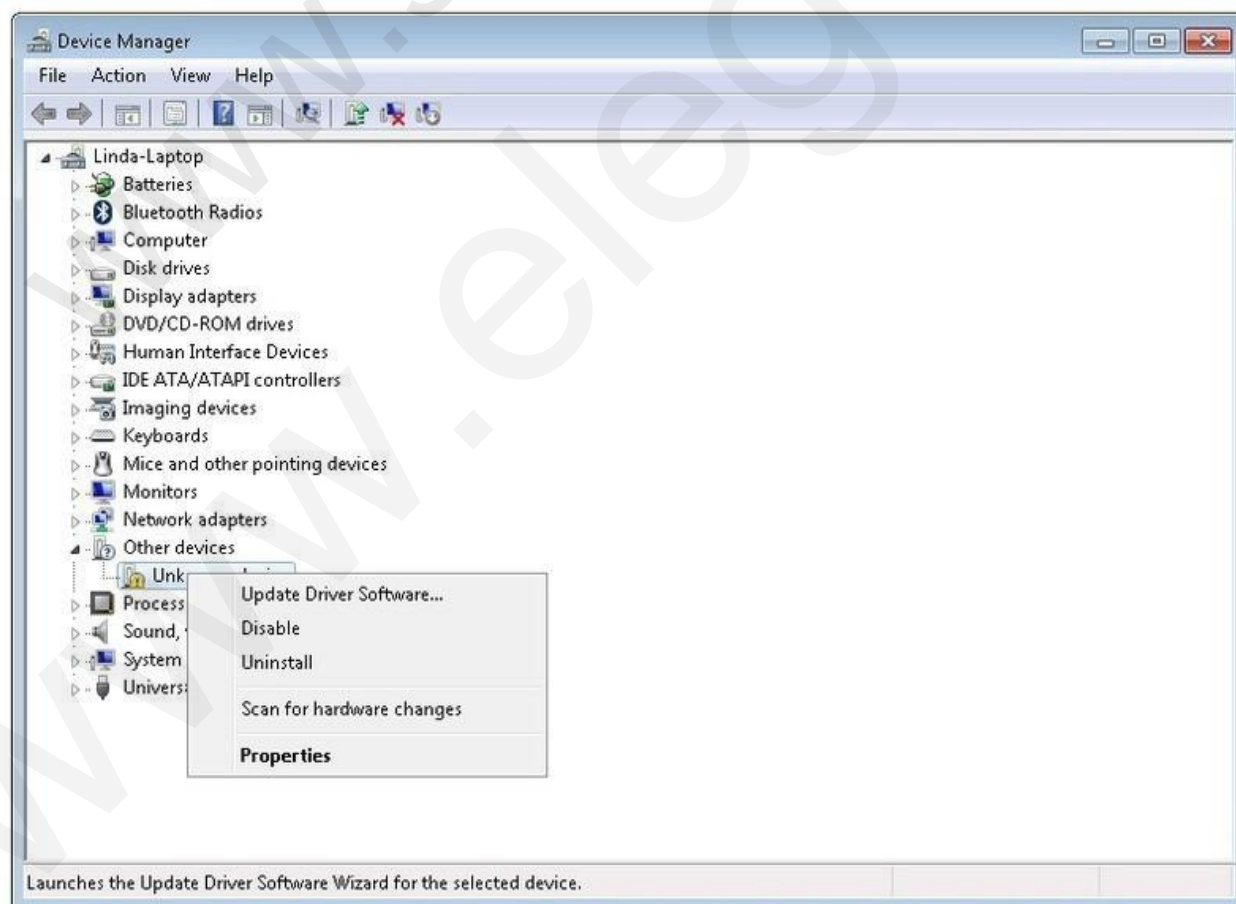
Cu toate acestea, această metodă de instalare necesită o instalare separată a driverului.

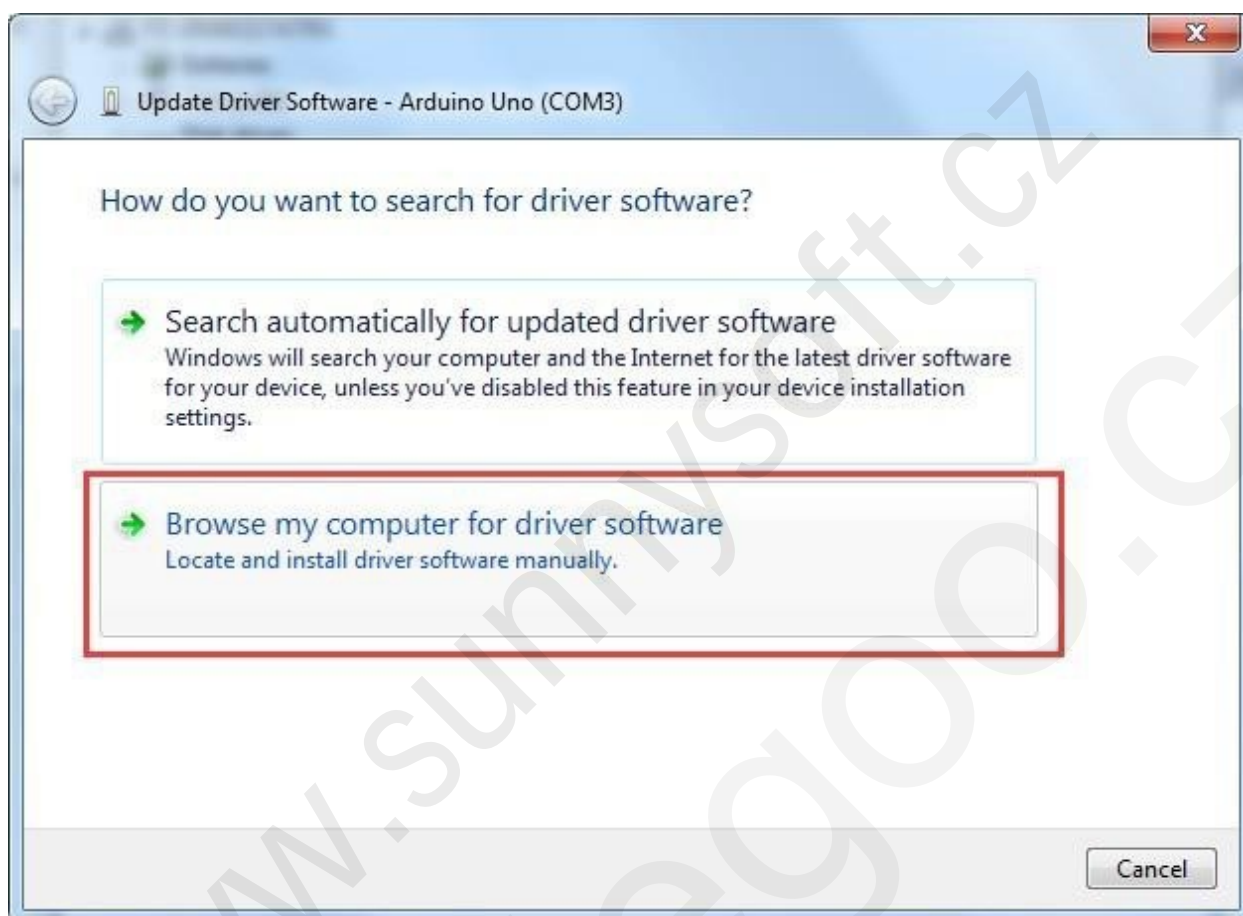
Dosarul Arduino conține atât programul Arduino în sine, cât și driverele care vă permit să conectați Arduino la computer utilizând un cablu USB. Înainte de a rula programul Arduino, veți instala driverele USB.

Conectați un capăt al cablului USB la Arduino și celălalt la mufa USB a computerului. LED-ul de alimentare se va aprinde, iar Windows poate afișa mesajul "Found New Hardware". Ignorați acest mesaj și anulați orice încercare a Windows de a instala automat drivere.

Cea mai fiabilă metodă de instalare a driverelor USB este utilizarea Managerului de dispozitive. Acesta poate fi accesat în moduri diferite, în funcție de versiunea de Windows. În Windows 7, trebuie mai întâi să deschideți Panoul de control, apoi să selectați Vizualizare icoane și ar trebui să găsiți Device Manager în listă.

Sub "Alte dispozitive" ar trebui să vedeți o pictogramă "dispozitiv necunoscut" cu un mic triunghi galben de avertizare lângă ea. Acesta este Arduino.

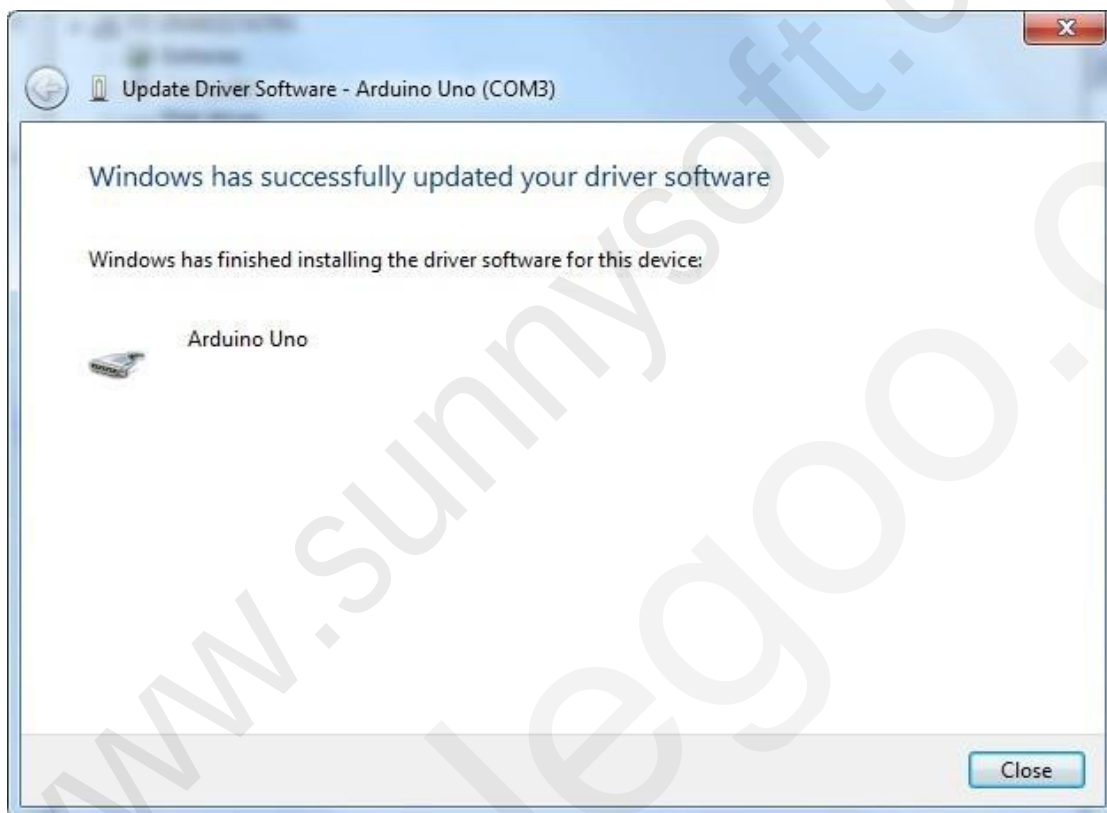




Faceți clic dreapta pe dispozitiv și selectați opțiunea din meniul de sus (Update driver software...). Vi se va solicita apoi să selectați "Automatically search for updated driver software" sau "Browse my computer for driver software". Selectați opțiunea de căutare și navigați la X\arduino1.8.0\drivers.



Faceți clic pe "Next" (Următorul) și este posibil să vedeți un avertisment de securitate, dacă este cazul, activați instalarea software-ului. După instalarea software-ului, va apărea un mesaj de confirmare.



Utilizatorii Windows pot sări peste instrucțiunile de instalare pentru Mac și Linux și să meargă la Lecția 1. Utilizatorii de Mac și Linux pot continua să citească această secțiune.

Instalarea Arduino (Mac OS X)

Pasul 1: Descărcați software-ul Arduino (IDE)

-Deschideți URL:<https://www.arduino.cc/en/Main/Software> cu browserul dvs.



-clic pe Mac OSX 10.7 Lion sau o versiune ulterioară

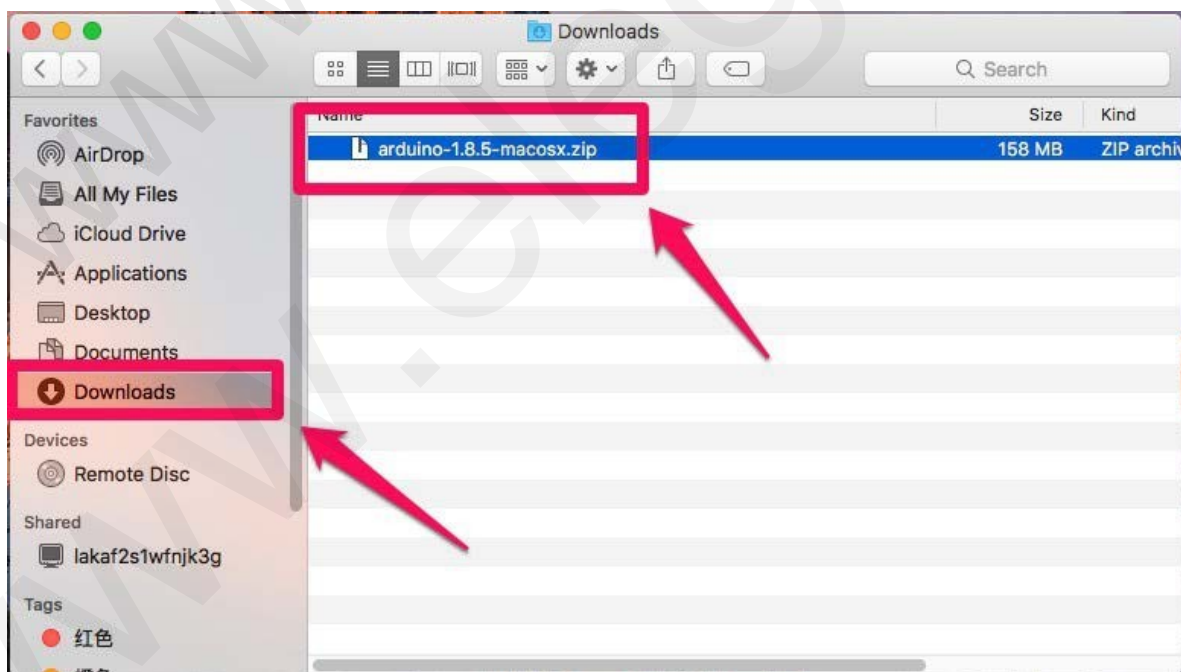
-click JUST DOWNLOAD

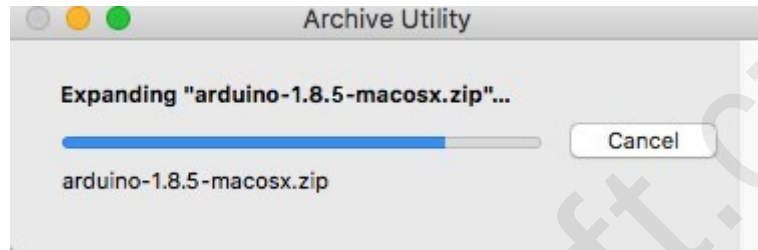
Support the Arduino Software

Consider supporting the Arduino Software by contributing to its development. (US tax payers, please note this contribution is not tax deductible). Learn more on how your contribution will be used.

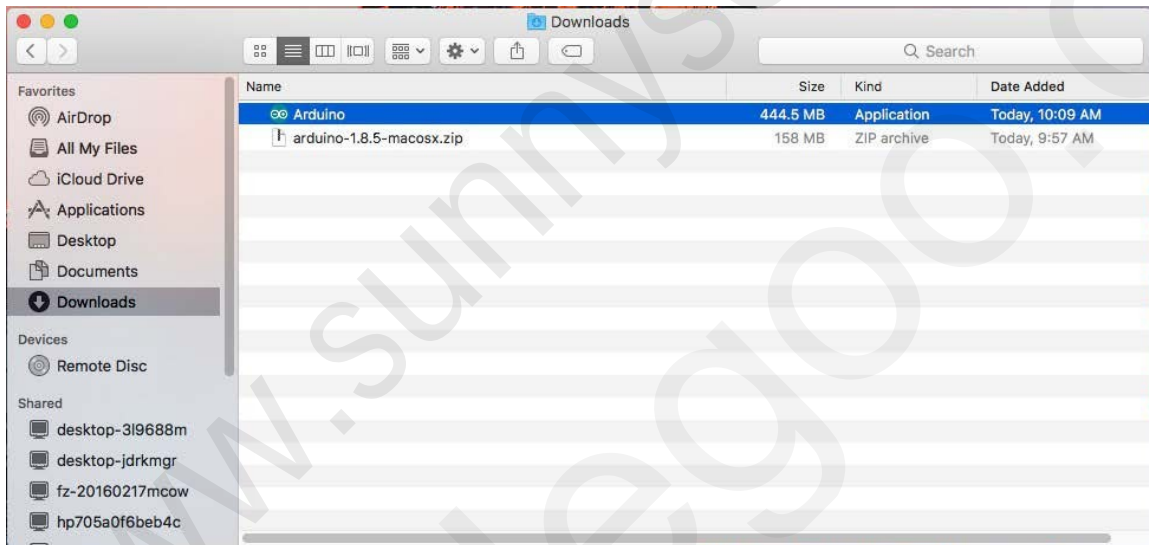


Descărcați și extrageți fișierul ZIP, faceți dublu clic pe Arduino.app și intrați în Arduino IDE; sistemul vă va solicita să instalați biblioteca Java runtime dacă nu o aveți pe computer. Odată ce instalarea este finalizată, puteți lansa Arduino IDE.

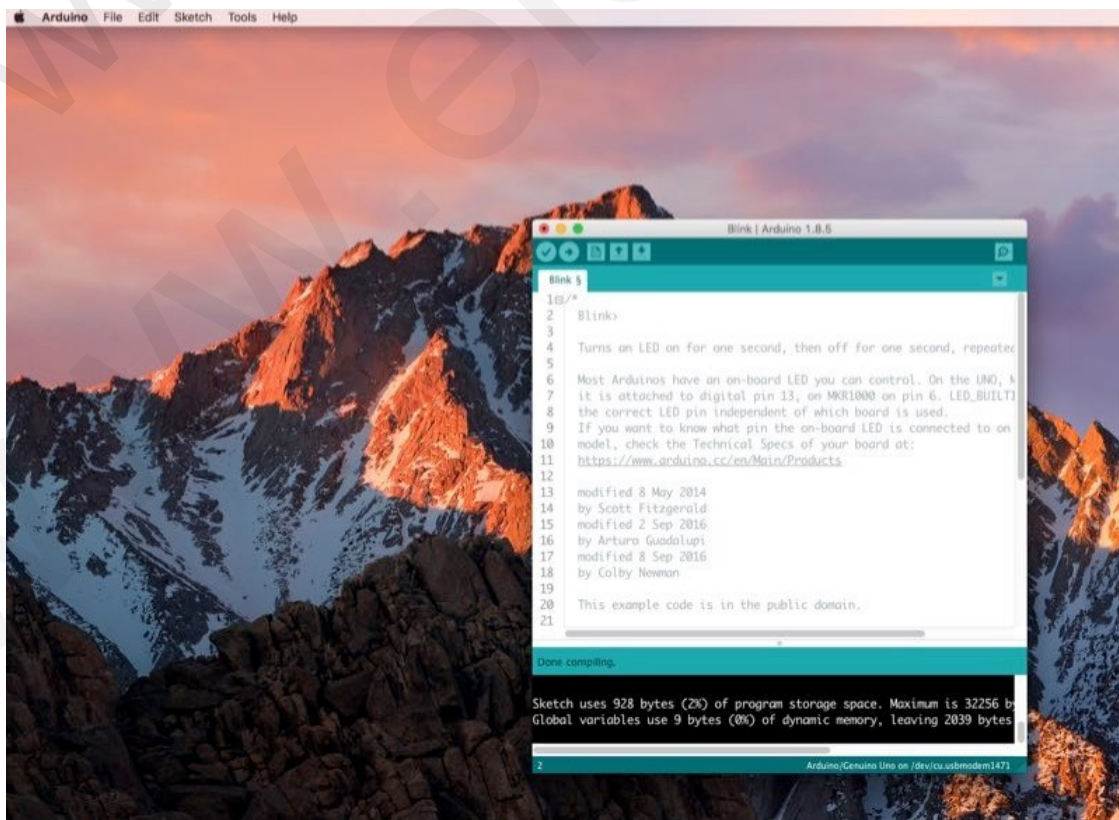




-fișierul Arduino este o aplicație Arduino, faceți dublu clic și deschideți-o.



-Arduino IDE



Instalarea Arduino (Linux)

Va trebui să utilizați comanda `make install`. Dacă utilizați Ubuntu, vă recomandăm instalați Arduino IDE din Ubuntu Software Center.



`arduino-1.8.5-linux32.tar.xz`



`arduino-1.8.5-linux64.tar.xz`

SFATURI: Dacă întâmpinați probleme la instalarea driverelor, vă rugăm să citiți secțiunea UNO R3, MEGA, NANO DRIVER FAQ.



INSTALL ARDUINO ON LINUX.pdf



UNO R3, MEGA, NANO DRIVER FAQ

Lecția 1 Adăugarea bibliotecilor și deschiderea monitorului serial

Instalarea altor biblioteci Arduino

Odată ce sunteți familiarizat cu software-ul Arduino și utilizați funcțiile încorporate, puteți doriți să extindeți capacitățile Arduino cu biblioteci suplimentare.

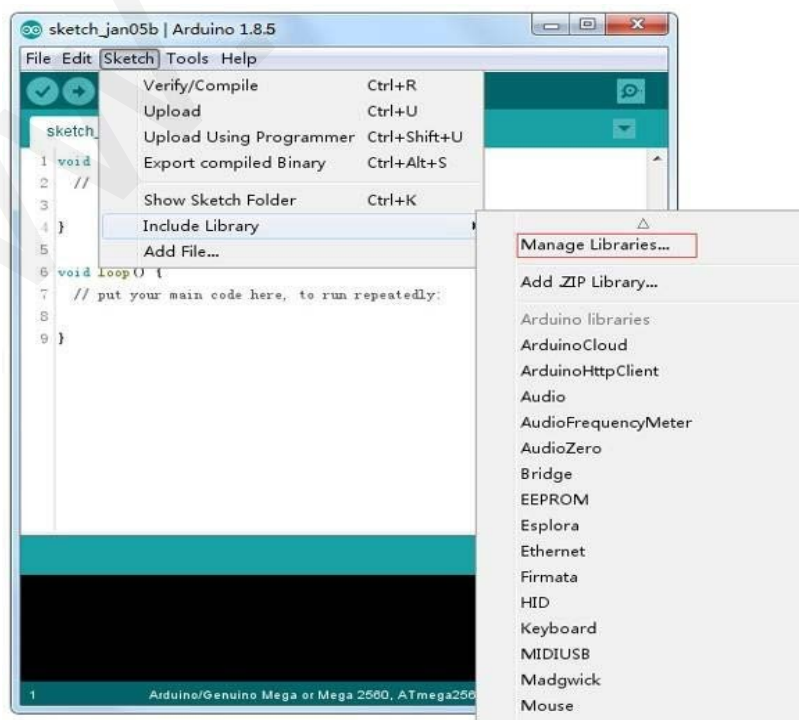
Ce sunt bibliotecile?

Bibliotecile sunt un set de coduri care facilitează conectarea la un senzor, afișaj, modul etc. De exemplu, biblioteca încorporată Liquid Crystal facilitează comunicarea cu afișajele LCD cu caractere. Sute de alte biblioteci sunt disponibile pentru descărcare pe internet. O listă a bibliotecilor încorporate și a unora dintre aceste biblioteci suplimentare este furnizată în referință. Pentru a utiliza bibliotecile suplimentare, trebuie să le instalați.

Cum să instalați biblioteca

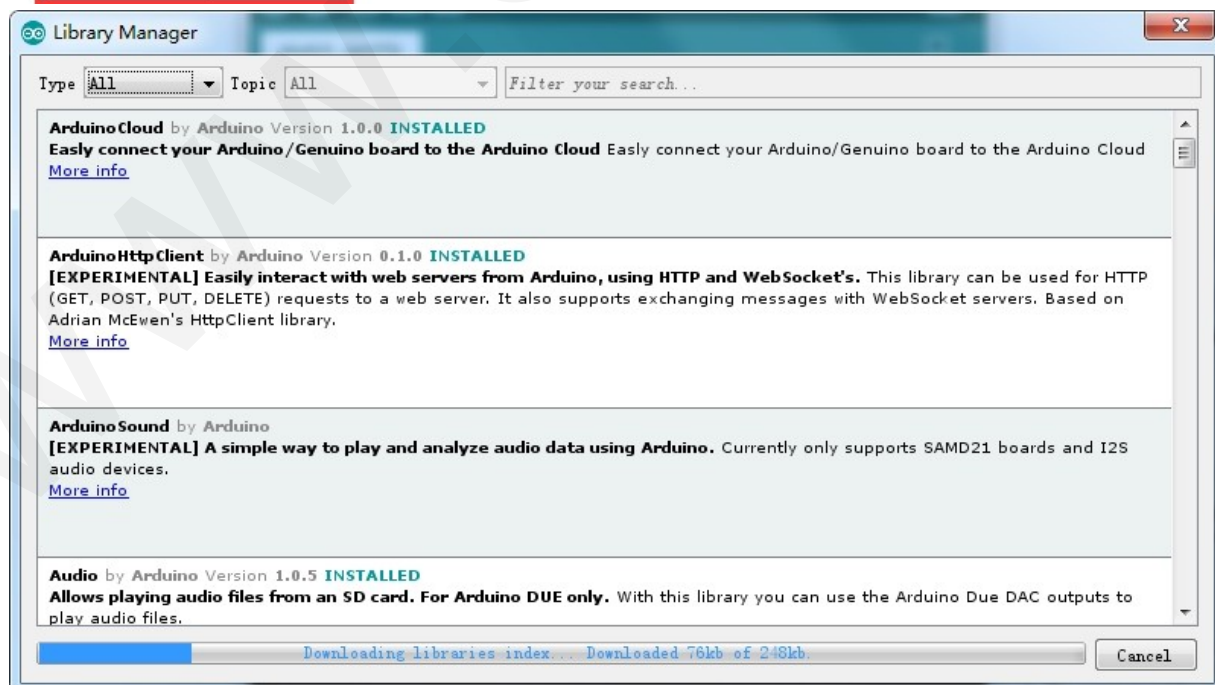
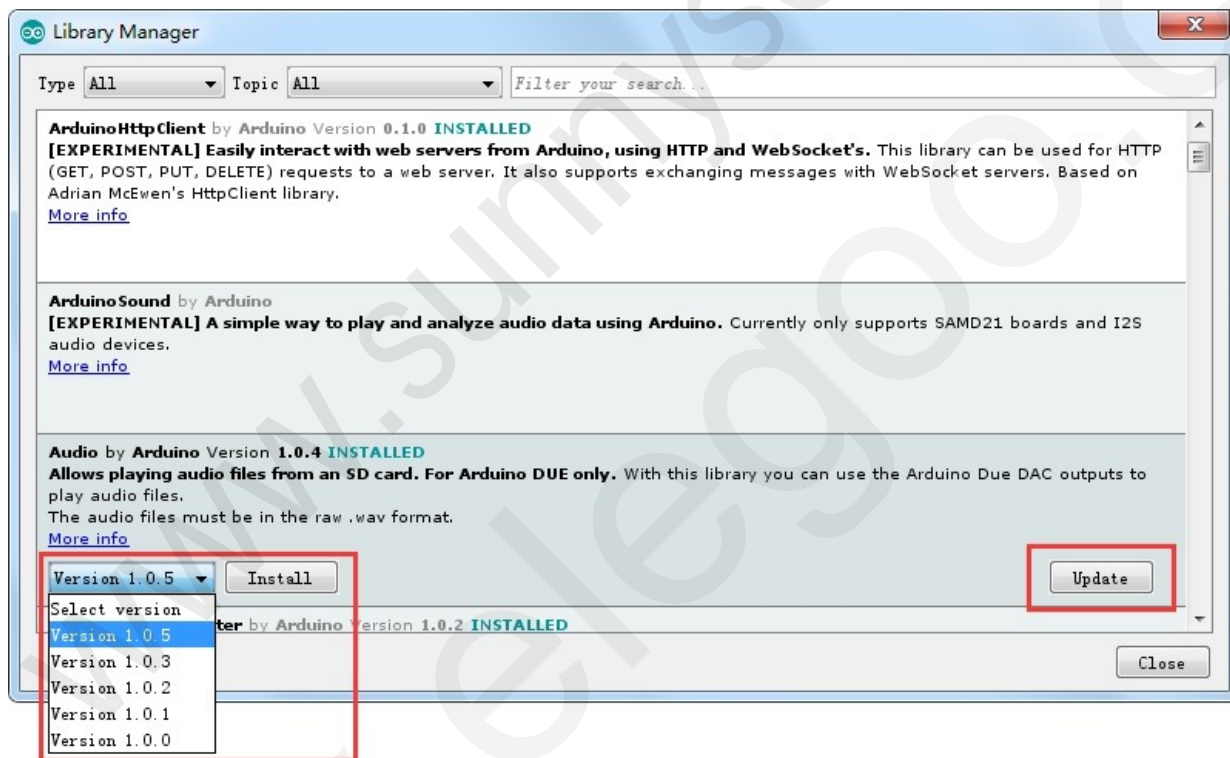
Utilizarea managerului de bibliotecă

Pentru a instala o bibliotecă nouă în Arduino IDE, puteți utiliza Library Manager (disponibil de la versiunea IDE 1.8.5). Deschideți IDE și faceți clic pe meniul "Sketch" și apoi pe Include Library > Manage Libraries.

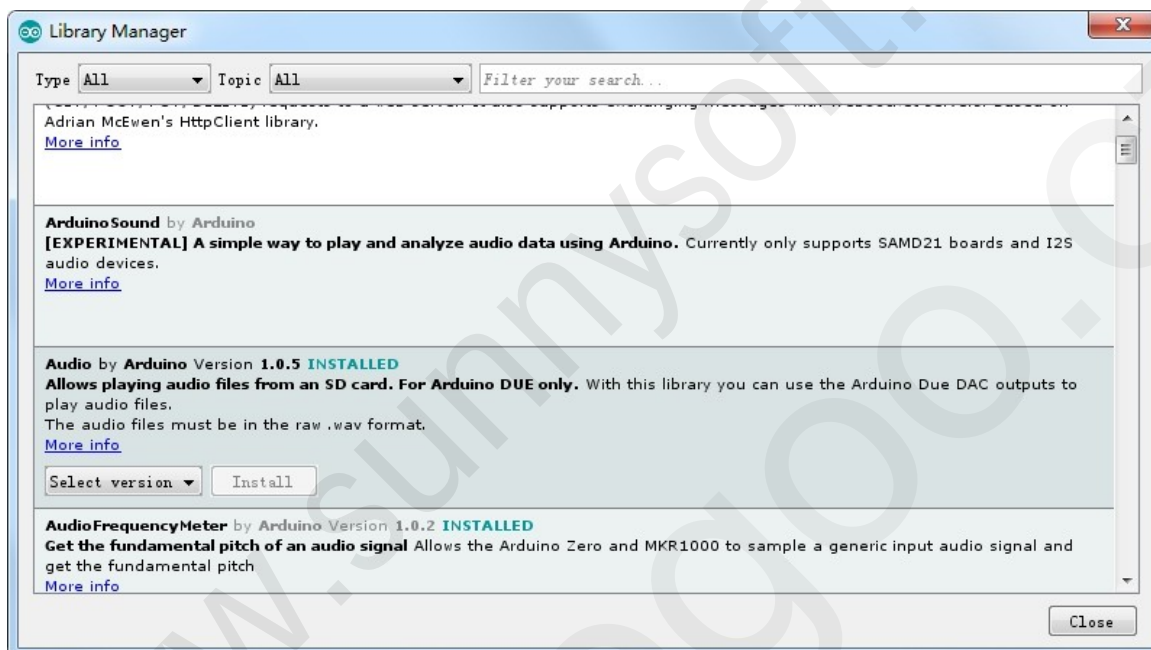


Managerul de biblioteci se va deschide și va lista bibliotecile care sunt deja instalate sau gata de a fi instalate. În acest exemplu, vom instala biblioteca Bridge. Derulați prin listă pentru a o găsi și apoi selectați versiunea bibliotecii pe care doriți să o instalați. Uneori este disponibilă o singură versiune a unei biblioteci. Dacă meniul pentru selectarea unei versiuni nu apare, nu vă faceți griji: acest lucru este normal.

Uneori trebuie să ai răbdare cu el, așa cum se vede în imagine. Restaurați-l și așteptați.



În final, faceți clic pe butonul de instalare și așteptați ca IDE-ul să instaleze noua bibliotecă. Descărcarea poate dura mai mult, în funcție de viteza conexiunii dvs. Când se termină, marcajul Instalat ar trebui să apară lângă biblioteca Bridge. Puteți închide managerul de biblioteci.

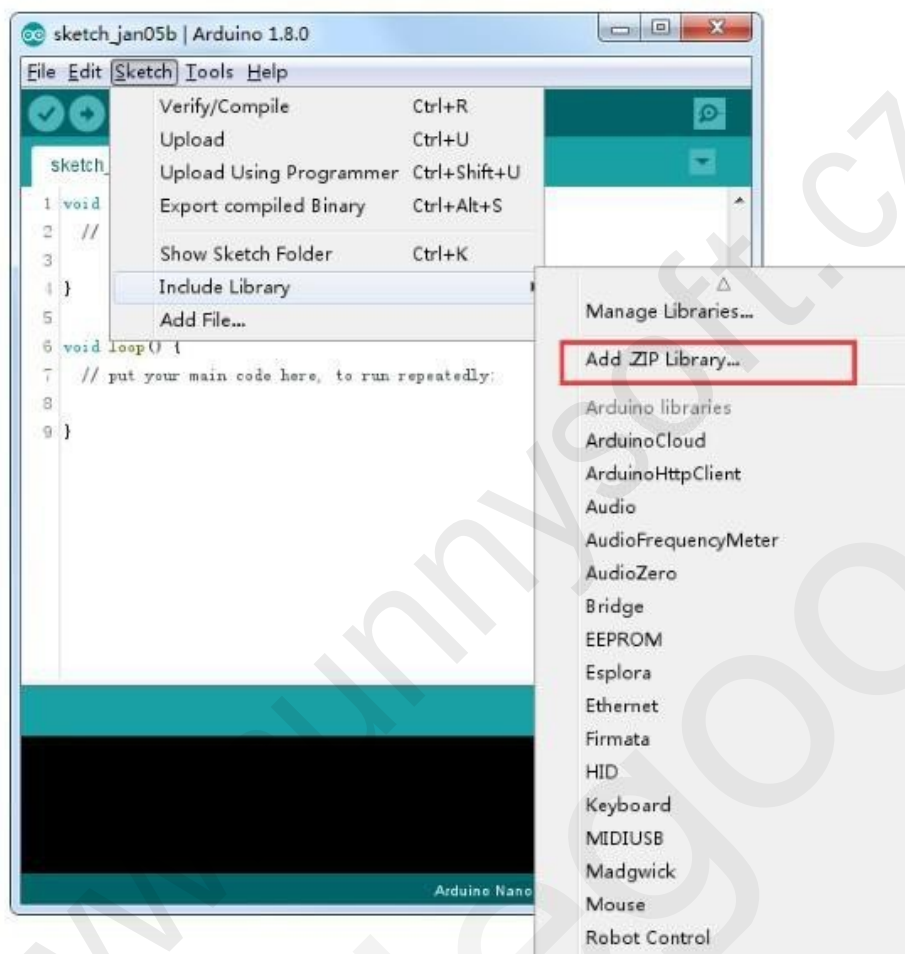


Acum puteți găsi noua bibliotecă în meniul Include library. Dacă doriți să adăugați propria bibliotecă, deschideți o nouă problemă pe [Github](#).

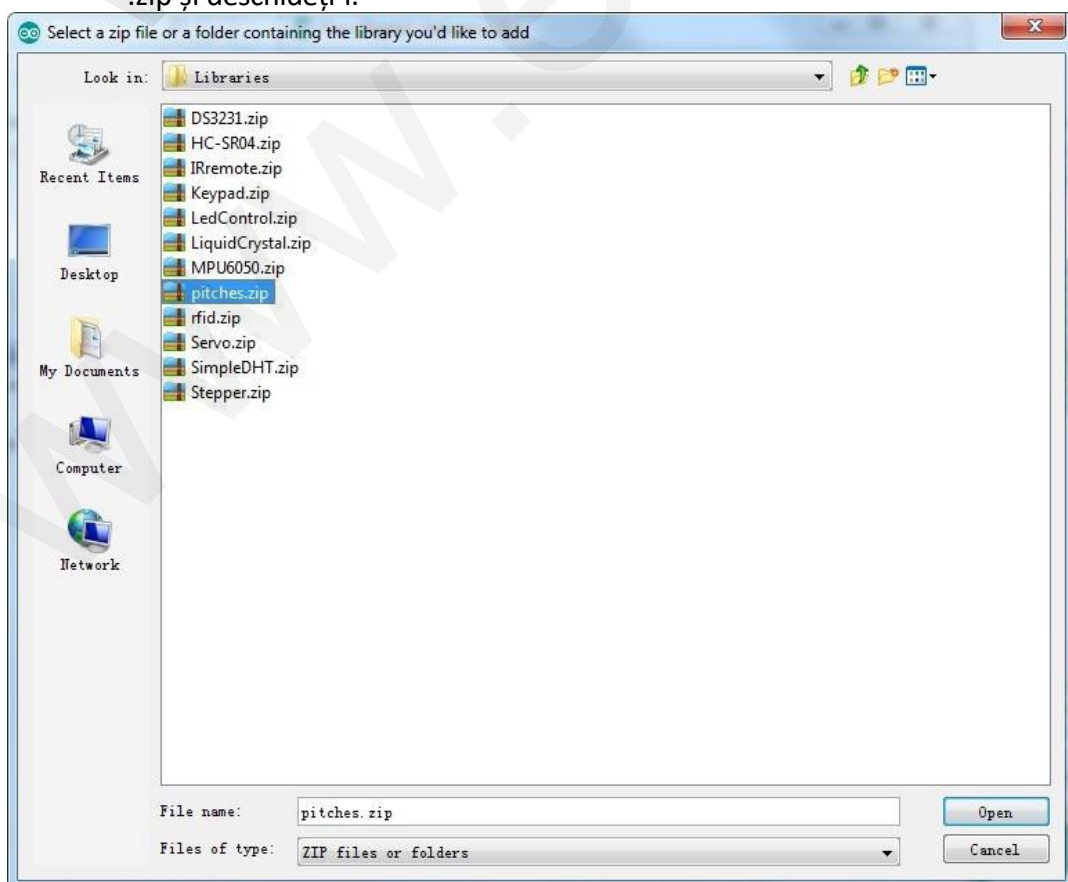
Importul unei biblioteci .zip

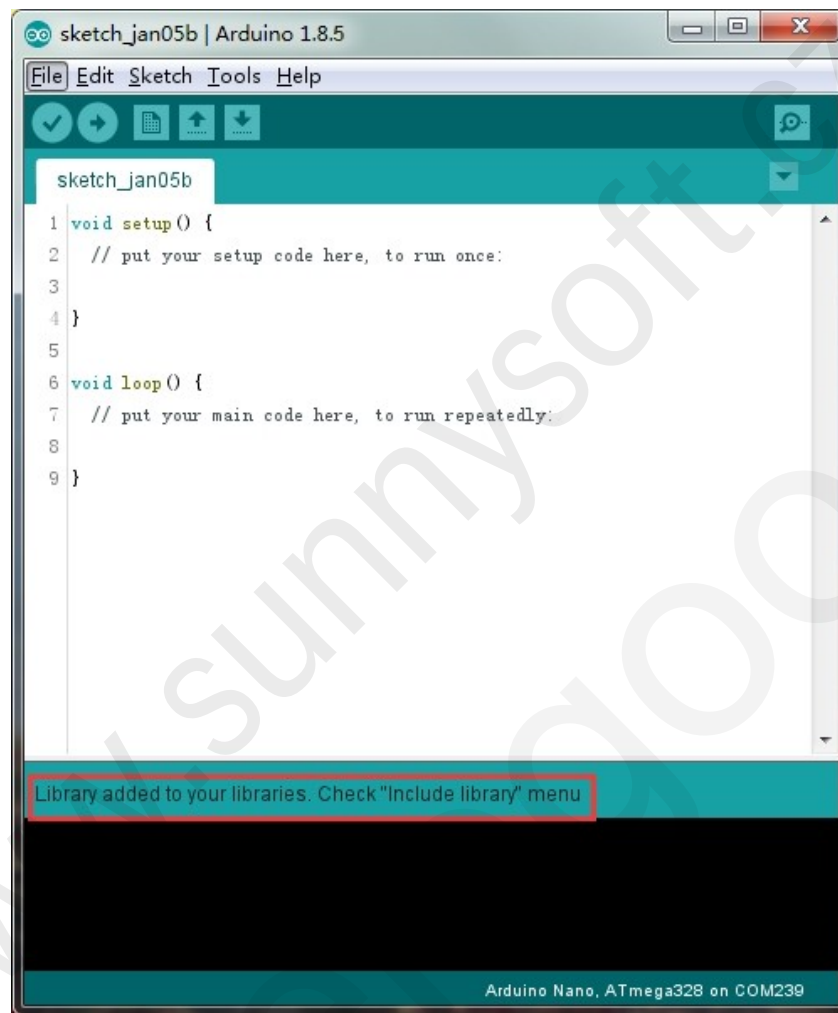
Bibliotecile sunt adesea distribuite sub forma unui fișier ZIP sau a unui dosar. Numele folderului este numele bibliotecii. În interiorul folderului se află un fișier .cpp, un fișier .h și adesea un fișier keywords.txt, un folder de exemple și alte fișiere necesare bibliotecii. Începând cu versiunea 1.0.5, puteți instala biblioteci terțe în IDE. Nu dezarhivați biblioteca descărcată, lăsați-o așa cum este.

În Arduino IDE, mergeți la Sketch > Include Library. În partea de sus a ferestrei Selectați "Add .ZIPLibrary" din lista derulantă.



Vi se va solicita să selectați biblioteca pe care doriți să o adăugați. Navigați la locația .zip și deschideți-l.





Reveniți la Sketch > Import Library. Biblioteca ar trebui să apară acum în partea de jos a meniului derulant. Aceasta este gata pentru a fi utilizată în schița dvs. Fișierul zip va fi extras în folderul libraries din Arduino sketchesdirectory.

Notă: biblioteca va fi disponibilă pentru utilizare în schițe, dar exemplele pentru bibliotecă nu vor fi expuse în File > Examples până când IDE-ul nu este repornit.

Aceste două abordări sunt cele mai comune. Abordări similare pot fi utilizate pe sistemele MAC și Linux. Instalarea manuală, care va fi prezentată mai jos ca o alternativă, poate fi utilizată rar și poate fi omisă de utilizatorii fără nevoi.

Instalare manuală

Pentru a instala biblioteca, mai întâi ieșiți din aplicația Arduino. Apoi dezarhivați fișierul ZIP care conține biblioteca. De exemplu, dacă instalați o bibliotecă numită

"ArduinoParty", dezarhivați fișierul ArduinoParty.zip. Acesta ar trebui să conțină un dosar numit ArduinoParty și, în cadrul acestuia, fișiere precum ArduinoParty.cpp și ArduinoParty.h. (Dacă fișierele .cpp și .h nu se află în dosar, va trebui să creați unul. În acest caz, veți crea un folder numit "ArduinoParty" și veți muta în el toate fișierele care se aflau în fișierul ZIP, cum ar fi ArduinoParty.cpp și ArduinoParty.h). Trageți și plasați folderul ArduinoParty în acest folder (folderul bibliotecii). În Windows, acesta se va numi probabil "My Documents\Arduino\libraries". Pentru utilizatorii Mac, se va numi probabil "Documents/Arduino/libraries". Pe Linux, va fi folderul "libraries" din caietul dvs. de schițe.

Dosarul bibliotecii Arduino ar trebui să arate acum astfel (pe

Windows): **My**

Documents\Arduino\libraries\ArduinoParty\ArduinoParty.cpp **My**
Documents\Arduino\libraries\ArduinoParty\ArduinoParty.h **My**
Documents\Arduino\libraries\ArduinoParty\ArduinoParty\Examples

sau după cum urmează (pe Mac și Linux):

Documents/Arduino/Libraries/ArduinoParty/ArduinoParty.cpp
Documents/Arduino/Libraries/ArduinoParty/ArduinoParty.h
Documents/Arduino/Libraries/ArduinoParty/Examples

....

Este posibil să existe mai multe fișiere decât fișierele .cpp și .h, asigurați-vă doar că sunt toate acolo. (Biblioteca nu va funcționa dacă introduceți fișierele .cpp și .h direct în bibliotecă sau dacă sunt imbricate în Următorul

dosar. De exemplu: documents\Arduino\libraries\ArduinoParty.cpp

și

Documents\Arduino\libraries\ArduinoParty\ArduinoParty\ArduinoParty.cpp nu va funcționa).

Reporniți aplicația Arduino. Asigurați-vă că noua bibliotecă apare în Sketch->Importați bibliotecă în meniul software. Asta este! Ați instalat biblioteca!

Arduino Serial Monitor (Windows, Mac, Linux)

Arduino Integrated Development Environment (IDE) este partea software a platformei

Arduino. Și deoarece utilizarea terminalului este o parte atât de importantă a lucrului cu

www.sunnysoft.cz

sistem

www.sunnysoft.cz
www.elegoo.cz

Arduino și alte microcontrolere, au decis să conecteze un terminal serial la software. În mediul Arduino, acest terminal se numește monitor serial.

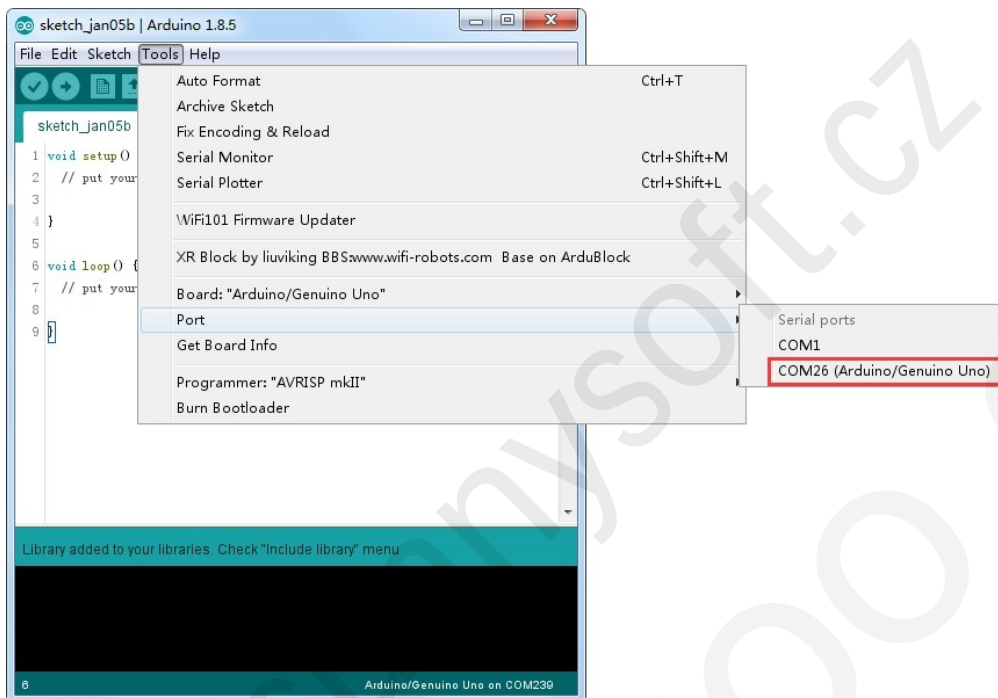
Stabilirea unei conexiuni

Monitorul serial vine cu toate versiunile IDE-ului Arduino. Pentru a-l utiliza pentru a-l deschide, faceți clic pe pictograma Serial Monitor.

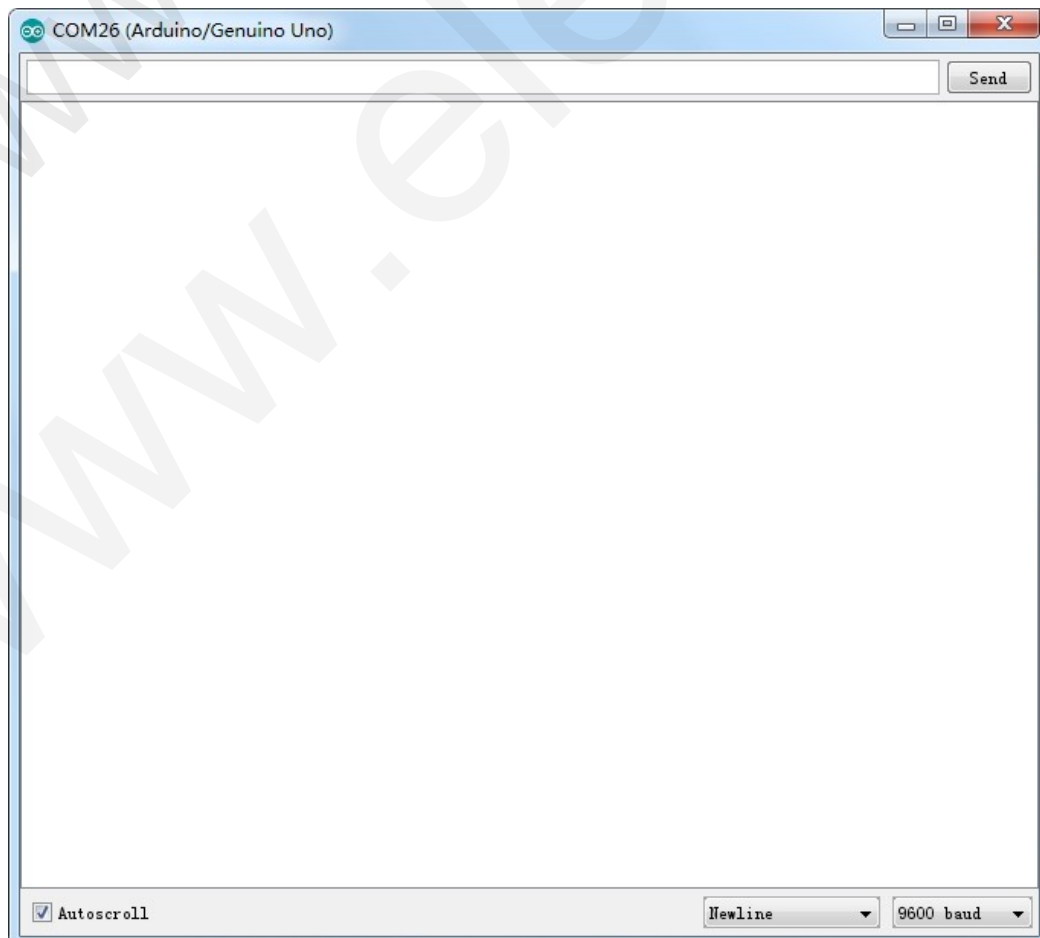


Selectarea portului de deschis în Monitorul serial este aceeași cu selectarea portului pentru încărcarea codului Arduino. Mergeți la Tools -> Serial Port și selectați portul corect.

Indicații: alegeți același port COM pe care îl aveți în DeviceManager.

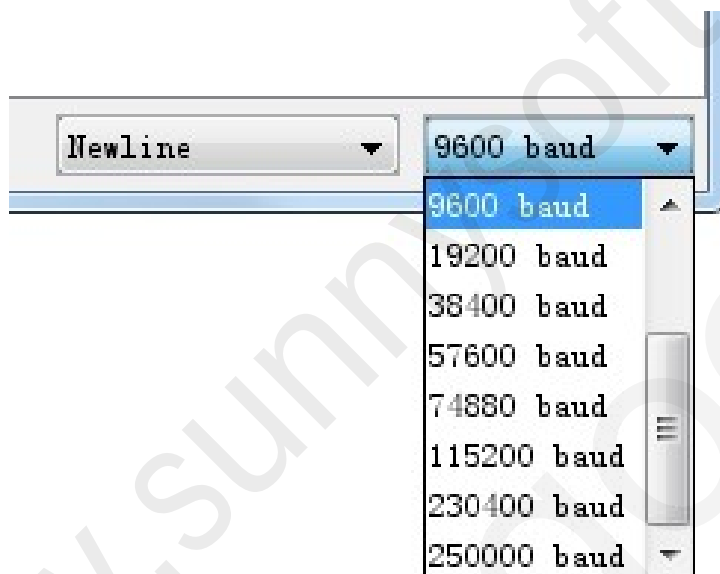


Când îl deschideți, ar trebui să vedeți ceva similar:



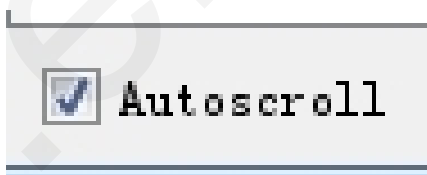
Setări

Monitorul serial are opțiuni de configurare limitate, dar este suficient pentru cele mai multe dintre nevoile dvs. de comunicare serială. Prima setare pe care o puteți modifica este rata de baud. Faceți clic pe meniul derulant Viteză de baud.



viteză) selectați rata de baud corectă (9600 baud).

În cele din urmă, puteți bifa caseta din colțul din stânga jos pentru a stabili dacă terminalul trebuie să defileze automat sau nu.



Avantaje

Un monitor serial este o modalitate rapidă și ușoară de a stabili o conexiune serială la Arduino. Dacă lucrați deja în Arduino IDE, nu este nevoie să deschideți un terminal separat pentru a vizualiza datele.

Dezavantaje

Lipsa setărilor face ca Monitorul Serial să fie destul de dorit și pentru utilizatorii avansați comunicarea serială poate să nu fie suficientă.

Lecția 2 Clipește

Prezentare generală

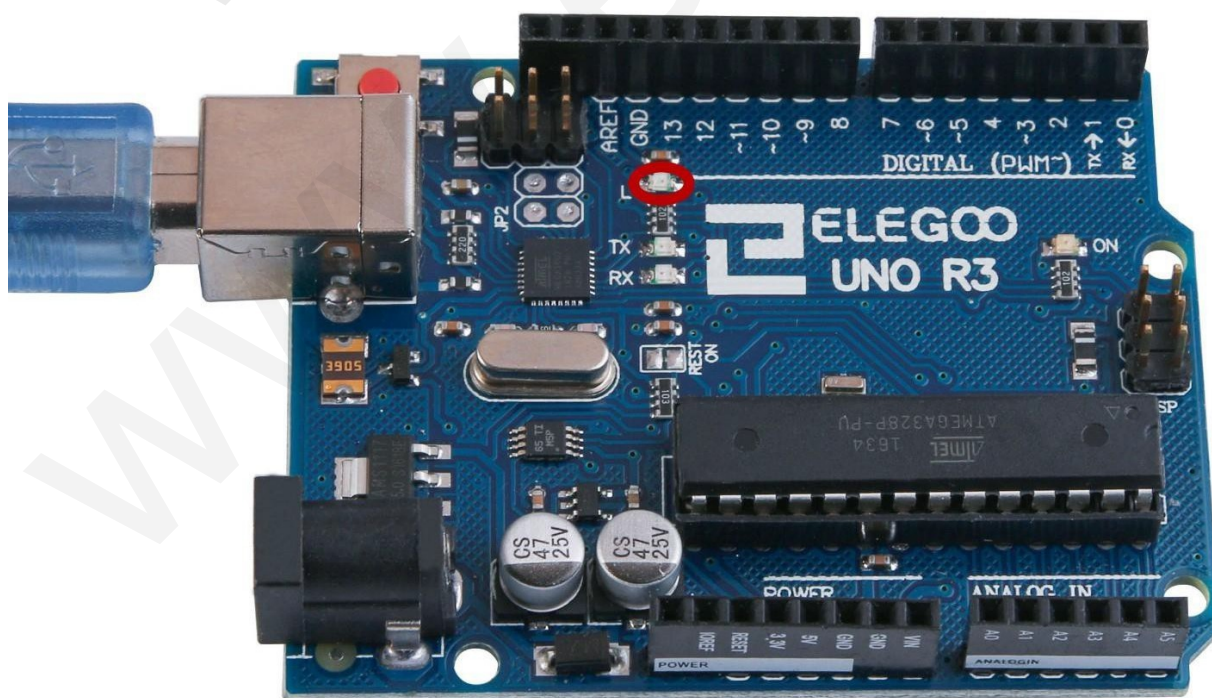
În această lecție, veți învăța cum să programați placa de control UNO R3 pentru a conecta LED-ul Arduino încorporat și cum să descărcați programe în pași de bază.

Componentă necesară:

(1) x Elegoo Uno R3

Principiul

Placa UNO R3 are un număr de conectori pe ambele părți care sunt utilizați pentru conectarea la mai multe dispozitive electronice și "scuturi" plug-in care îi extind capacitățile. De asemenea, are un LED pe care îl puteți controla din schițele dumneavoastră. Acest LED este încorporat în placa UNO R3 și este adesea denumit LED "L", deoarece așa este marcat pe placă.



31/162

Este posibil să observați că LED-ul "L" de pe placa UNO R3 clipește atunci când o conectați la o priză USB. Acest lucru se datorează faptului că plăcile vin de obicei cu schița "Blink" preinstalată.

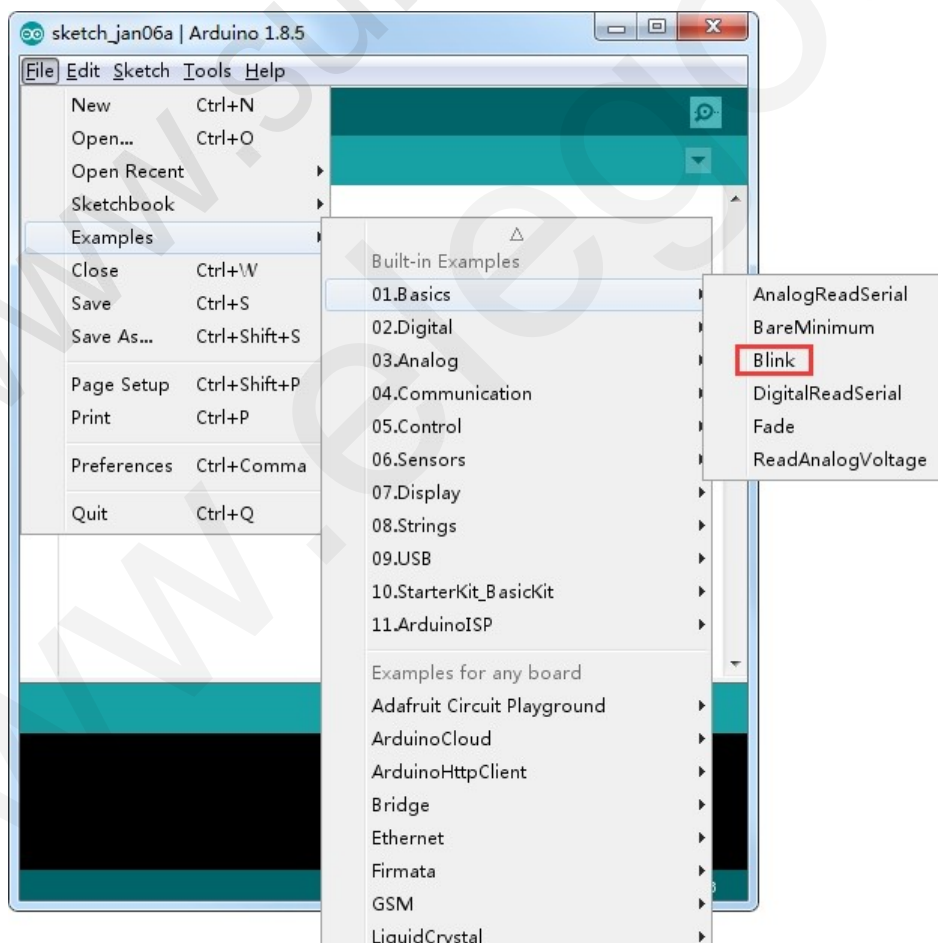
În această lecție, vom reprograma placa UNO R3 folosind propria noastră schiță Blink și apoi modificați viteza la care clipește.

În Lecția 0, ați configurat IDE-ul Arduino și v-ați asigurat că ați găsit portul serial corect pentru a vă conecta la placa UNO R3. Acum este momentul să testați această conexiune și să programați placa UNO R3.

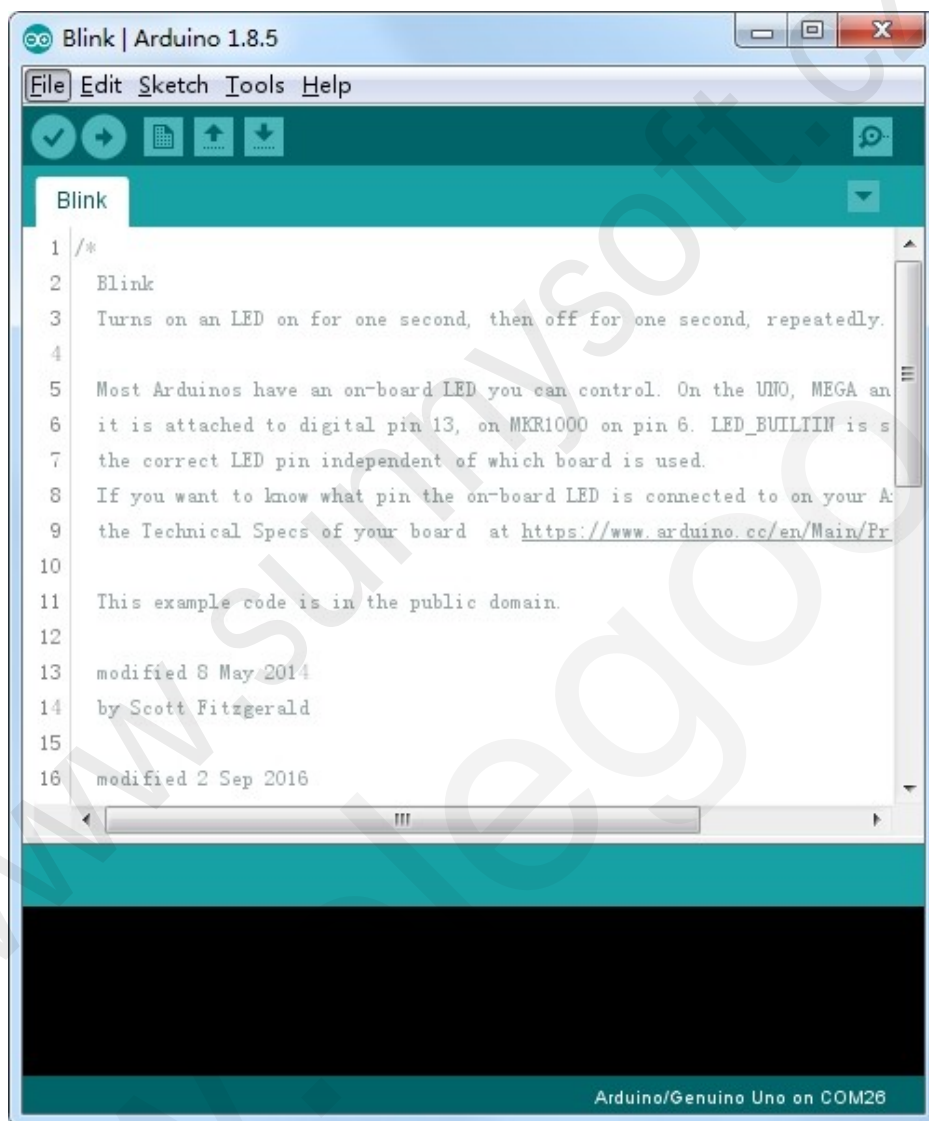
Arduino IDE conține o colecție extinsă de exemple de schițe pe care le puteți încărca și utiliza.

Acesta include o schiță de exemplu pentru aprinderea intermitentă a LED-ului "L".

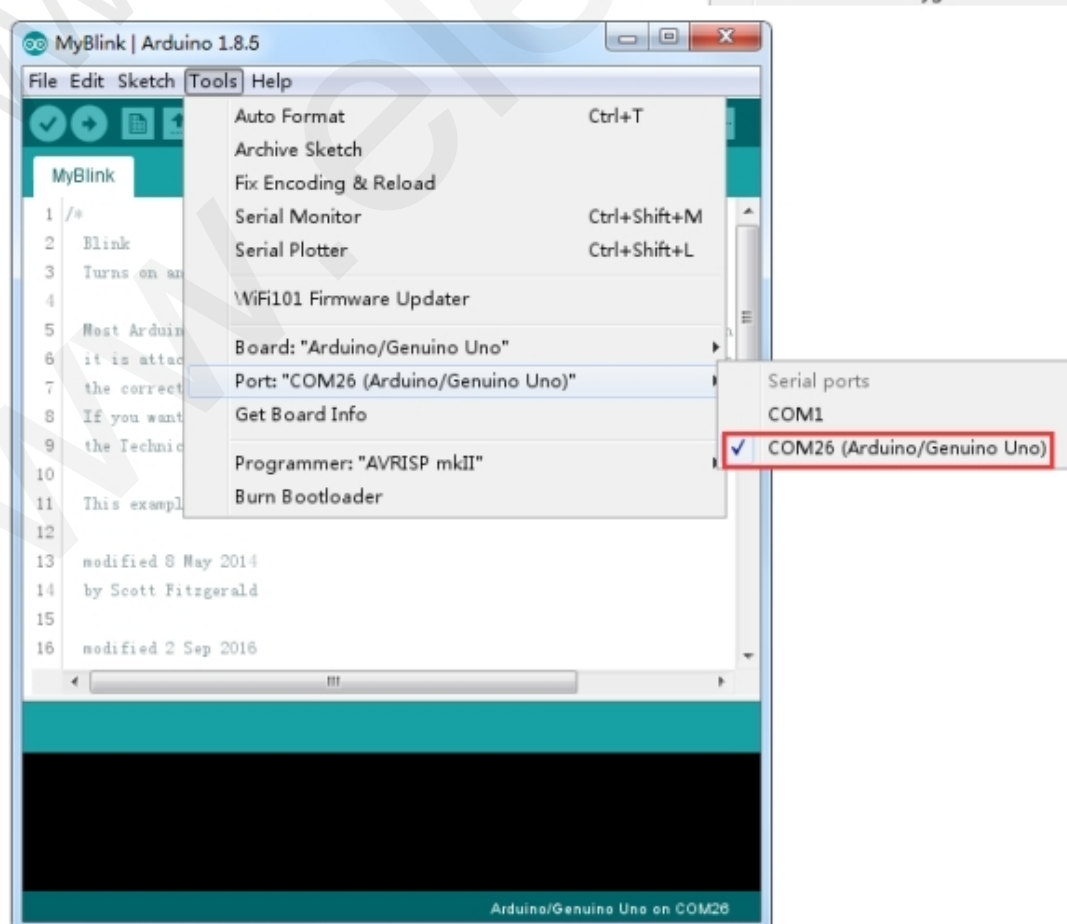
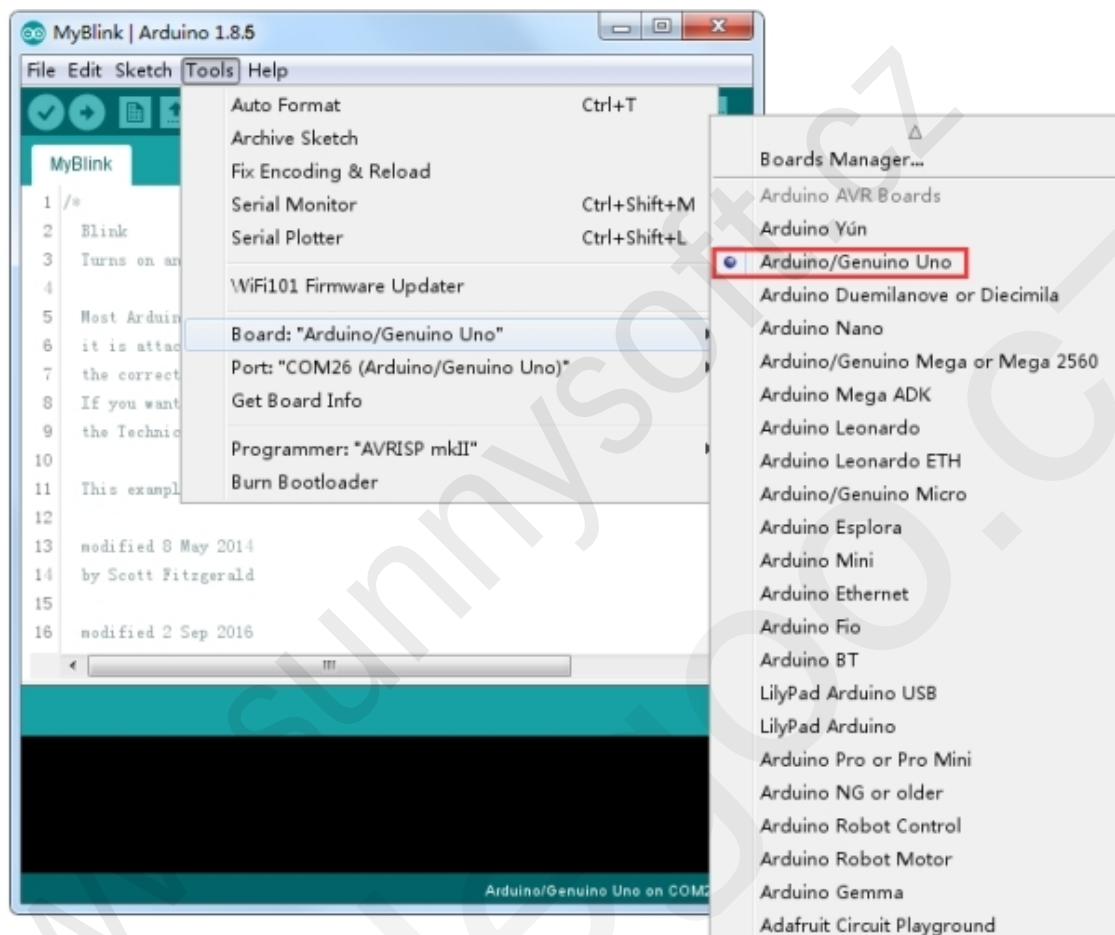
Încărcați schița "Blink", care poate fi găsită în meniul IDE sub File > Examples > 01.Basics.



Când deschideți fereastra schiței, măriți pentru a vedea întreaga schiță.

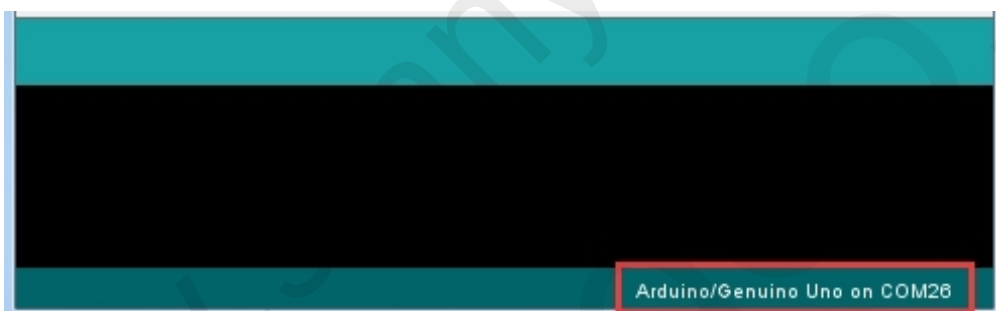


Conectați placa Arduino la computer utilizând un cablu USB și verificați dacă elementele "Board Type" și "Serial Port" sunt setate corect.



Notă: Este posibil ca tipul de placă și portul serial de aici să nu coincidă cu imaginea. Dacă utilizați o placă 2560, atunci trebuie să selectați Mega 2560 ca tip de placă, celelalte opțiuni pot fi făcute în același mod. Și portul serial afișat este diferit pentru fiecare, deși COM 26 este selectat aici, acesta poate fi COM3 sau COM4 în computerul dvs. Portul COM corect ar trebui să fie COMX (arduino XXX), care este în conformitate cu criteriile de certificare.

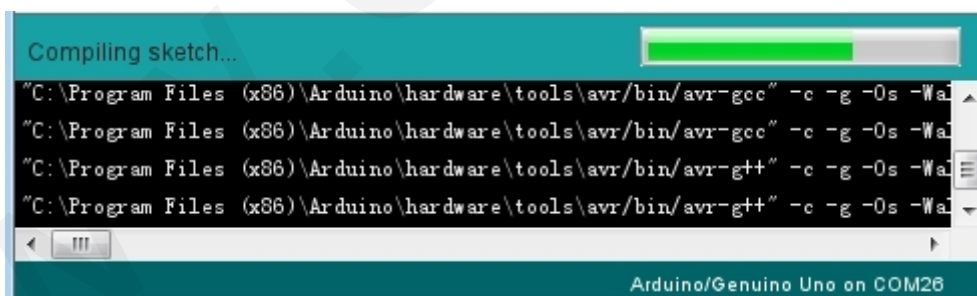
Arduino IDE vă va afișa setările curente ale plăcii în partea de jos a ferestrei.



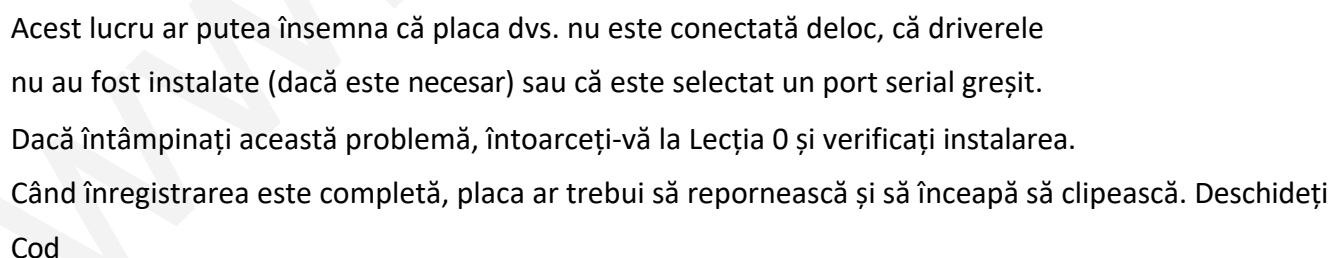
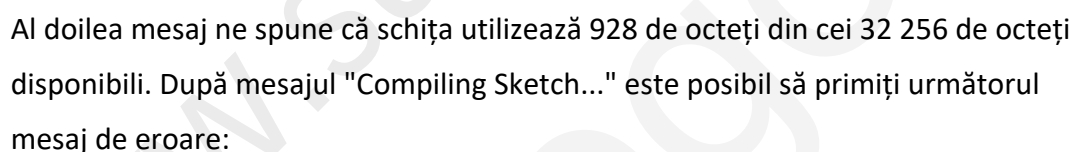
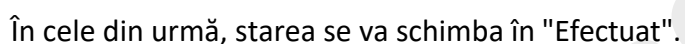
Faceți clic pe butonul "Încărcare". Al doilea buton din stânga pe bara de instrumente.



Dacă urmăriți zona de stare a IDE, veți vedea o bară de progres și o serie de mesaje. Mai întâi veți vedea mesajul "Compiling Sketch..." (Compilarea schiței...). Aceasta va converti schița într-un format potrivit pentru încărcarea pe placă.



Apoi starea se va schimba în "Înregistrare". În acest moment, LED-urile de pe Arduino ar trebui să înceapă să clipească pe măsură ce schița este transferată.



Tot ce se află între /* și */ în partea de sus a schiței este un comentariu de bloc; acesta explică pentru ce este schița

servește.

Comentariile pe o singură linie încep cu caracterul `//` și tot ce se află până la sfârșitul liniei respective este considerat comentariu.

Prima linie de cod este:

```
int led = 13;
```

După cum se explică în comentariul de mai sus, aceasta este denumirea pinului la care este conectat LED-ul. La majoritatea Arduino, inclusiv UNO și Leonardo, acesta este 13.

În continuare avem funcția de "configurare". După cum s-a menționat în comentariu, această funcție este reactivată atunci când este apăsat butonul de resetare. De asemenea, este executată ori de câte ori placa este resetată din orice motiv, de exemplu atunci când este conectată pentru prima dată la alimentare sau după ce a fost încărcată o schiță. `void setup() {`

```
// inițializează pinul digital ca ieșire. pinMode(led,
OUTPUT);
}
```

Fiecare schiță Arduino trebuie să aibă o funcție de "configurare" și un loc unde ați putea dori să adăugați

instrucțiuni personalizate, este între `{` și `}`.

În acest caz, există o singură comandă, care, după cum se menționează în comentarii, spune placa Arduino că vom folosi pinul LED ca ieșire.

Funcția "loop" este, de asemenea, o parte obligatorie a schiței. Spre deosebire de funcția "setup", care este executată o singură dată după o resetare, funcția "loop" este executată din nou imediat după finalizarea comenzilor.

```
void loop() {
    digitalWrite(led,);          // aprindeți LED-ul (HIGH este
    nivelul de tensiune)delay(1000);      // așteptați o
    secundă
    digitalWrite(led,);          // stingeți LED-ul astfel încât tensiunea să fie
    LOW delay(1000);              // așteptați o secundă.
}
```

În interiorul funcției de buclă, comenzile aprind mai întâi LED-ul pinului (HIGH), apoi "întârzie" timp de 1000 de milisecunde (1 secundă), apoi sting LED-ul pinului și fac pauză pentru încă o secundă.

Acum faceți ca LED-ul să clipească mai repede. După cum probabil ați ghicit, cheia pentru acest lucru este

în modificarea parametrului din () pentru comanda "delay".


```
30 // the loop function runs over and over again forever
31 void loop() {
32   digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the volt
33   delay(500) // wait for a second
34   digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the vo
35   delay(500) // wait for a second
36 }
```

Acest timp de întârziere este exprimat în milisecunde, deci dacă doriți ca LED-ul să clipească de două ori mai repede, modificați valoarea de la 1000 la 500. Astfel, fiecare întârziere va face o pauză de o jumătate de secundă, nu de o secundă întreagă.

Reîncărcați schița și ar trebui să vedeți că LED-ul începe să clipească mai repede.

Lecția 3 LED

Prezentare generală

În această lecție veți învăța cum să modificați luminozitatea unui LED folosind valori diferite ale rezistenței.

Componentă necesară:

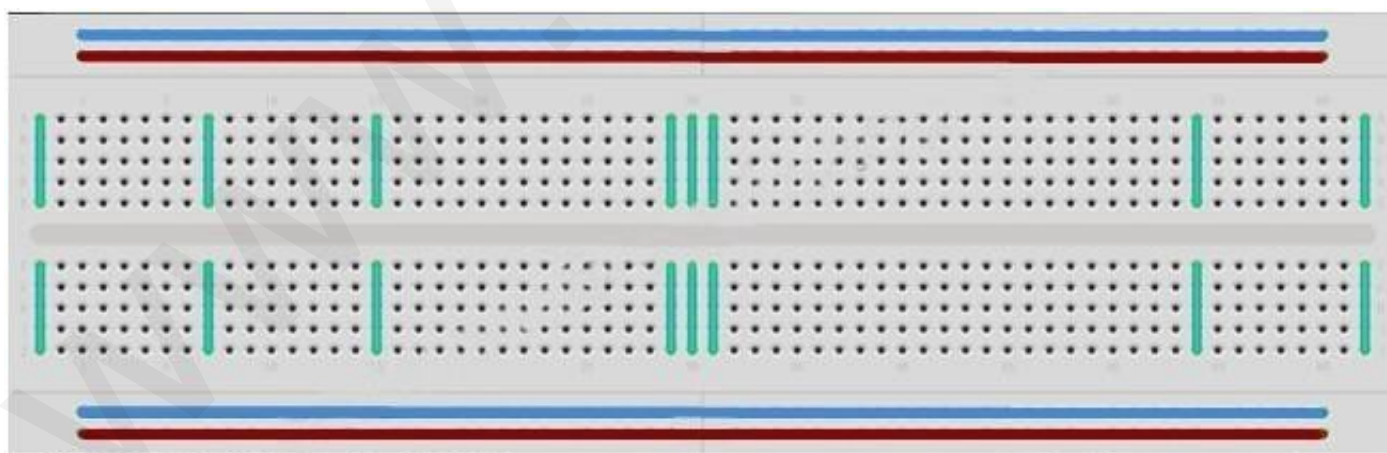
- (1) x Elegoo Uno R3
- (1) x 5 mm LED roșu
- (1) x 220 ohmi rezistență
- (1) x 1k ohm rezistență
- (1) rezistor x 10k ohm
- (2) x fire M-M (fire de interconectare de la mascul la mascul)

Introducere în dosar

PLACĂ DE PÂINE MB-102 :

Placa permite prototiparea rapidă a circuitelor fără a fi necesară lipirea conexiunilor.

Mai jos este un exemplu.



Breadboards sunt disponibile într-o varietate de dimensiuni și configurații. Cel mai simplu tip este doar o rețea de găuri într-un bloc de plastic. În interior sunt benzi metalice care asigură conexiuni electrice între găurile din rânduri mai scurte. Prin introducerea picioarelor a două componente diferite în același rând, acestea sunt conectate electric. Canalul adânc care curge pe mijloc indică faptul că există o întrerupere în conexiune, ceea ce înseamnă că puteți introduce un cip cu pinii de o parte și de alta a canalului fără ca aceștia să se unească. Unele plăci de breadboard au două benzi de găuri de-a lungul marginilor lungi ale plăcii, care sunt separate de grila principală. Acestea au benzi care se întind pe întreaga lungime a plăcii în interior și oferă o modalitate de a conecta tensiunea comună. Acestea sunt de obicei în perechi pentru +5V și masă. Aceste benzi sunt denumite șine și permit conectarea alimentării la mai multe componente sau puncte de pe placă. Deși plachetele sunt excelente pentru prototipuri, acestea au unele limitări. Deoarece sunt îmbinări temporare, ele nu sunt la fel de fiabile ca îmbinările prin lipire. Dacă aveți probleme intermitente cu un circuit, este posibil să se datoreze unei conexiuni proaste pe placă.

LED:

LED-urile sunt indicatoare luminoase excelente. Acestea consumă foarte puțină energie electrică și durează practic pentru totdeauna.

În această lecție, veți utiliza probabil cel mai comun LED: un LED roșu de 5 mm.

Diametrul de 5 mm înseamnă diametrul LED-ului. Nu puteți conecta LED-ul direct la o baterie sau la o sursă de alimentare deoarece 1) LED-ul are un conductor pozitiv și unul negativ și nu se va aprinde dacă este amplasat greșit și 2) LED-ul are un conductor pozitiv și unul negativ.

2) LED-ul trebuie să fie utilizat cu o rezistență pentru a limita sau "accelera" cantitatea de curent care trece prin diodă; în caz contrar, se va arde!



Dacă nu folosiți o rezistență pe un LED, acesta poate fi distrus aproape instantaneu, deoarece prin el trece prea mult curent, care îl încălzește și distruge "joncțiunea" în care este produsă lumina.

Există două moduri de a determina care fir al LED-ului este pozitiv și care este negativ. În primul rând, firul pozitiv este mai lung.

În al doilea rând, în punctul în care conductorul negativ intră în corpul LED-ului, marginea plată a carcasei

LED.

Dacă aveți un LED care are o parte plată lângă un fir mai lung, ar trebui să presupunem că firul mai lung este pozitiv.

REZISTENȚĂ:

După cum sugerează și numele, rezistoarele rezistă fluxului de electricitate. Cu cât valoarea rezistorului este mai mare, cu atât acesta opune mai multă rezistență și cu atât mai puțin curent electric trece prin el. Folosim acest lucru pentru a controla cât de mult curent electric trece prin LED și, prin urmare, cât de puternic strălucește.

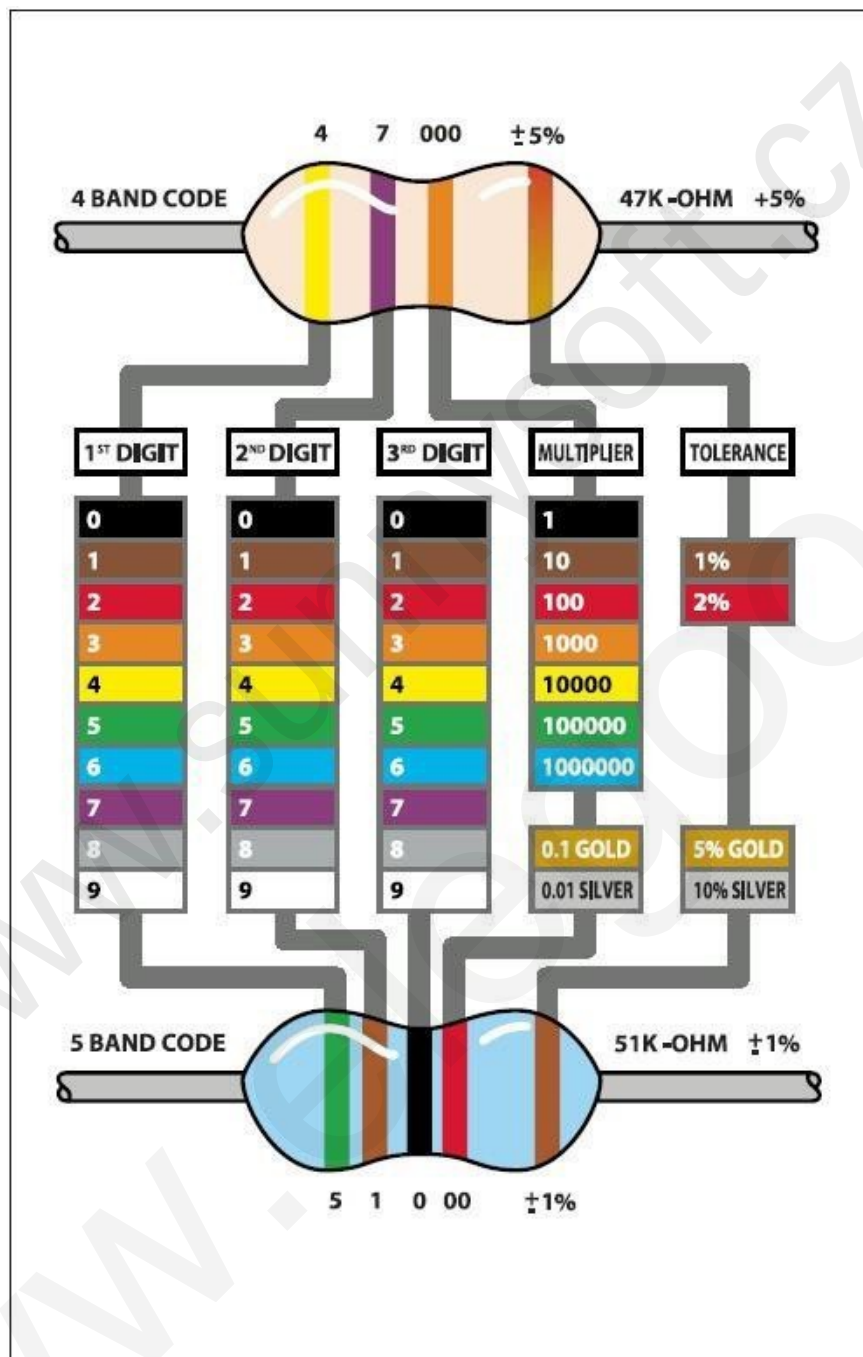


Dar mai întâi, mai multe despre...

Unitatea de rezistență se numește ohm, care este de obicei abreviată la Ω în litera greacă Omega. Deoarece ohm-ul este o valoare de rezistență mică (nu pune deloc multă rezistență), ne referim și la valorile rezistenței în $k\Omega$ (1.000 Ω) și $M\Omega$ (1.000.000 Ω). Acestea sunt numite kiloohmi și megaohmi.

În această lecție vom folosi trei valori diferite ale rezistenței: 220 Ω , 1k Ω și 10k Ω . Toate aceste rezistențe arată la fel, doar că au dungi colorate diferite pe ele. Aceste dungi vă spun ce valoare are rezistorul.

Marcajul de culoare al rezistorului are trei dungi colorate și o dungă aurie la un capăt.



Spre deosebire de LED-uri, rezistențele nu au un conductor pozitiv și unul negativ. Ele pot fi conectate în ambele direcții.

Dacă această metodă vi se pare prea complicată, puteți citi direct inelul colorat de pe rezistențele noastre și puteți determina valoarea rezistenței. Sau puteți utiliza în schimb un multimetru digital.

Conectarea

Diagrama

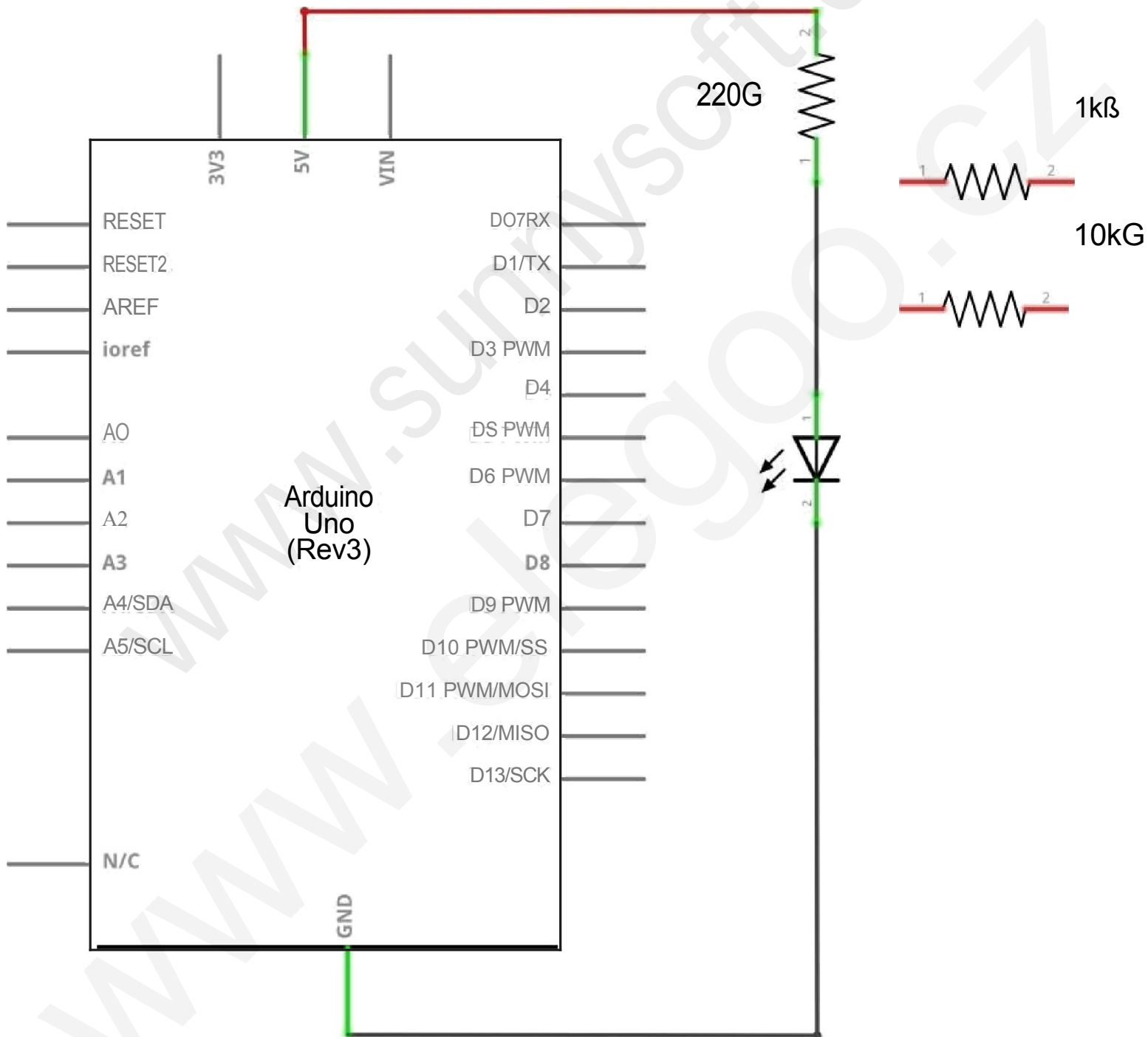
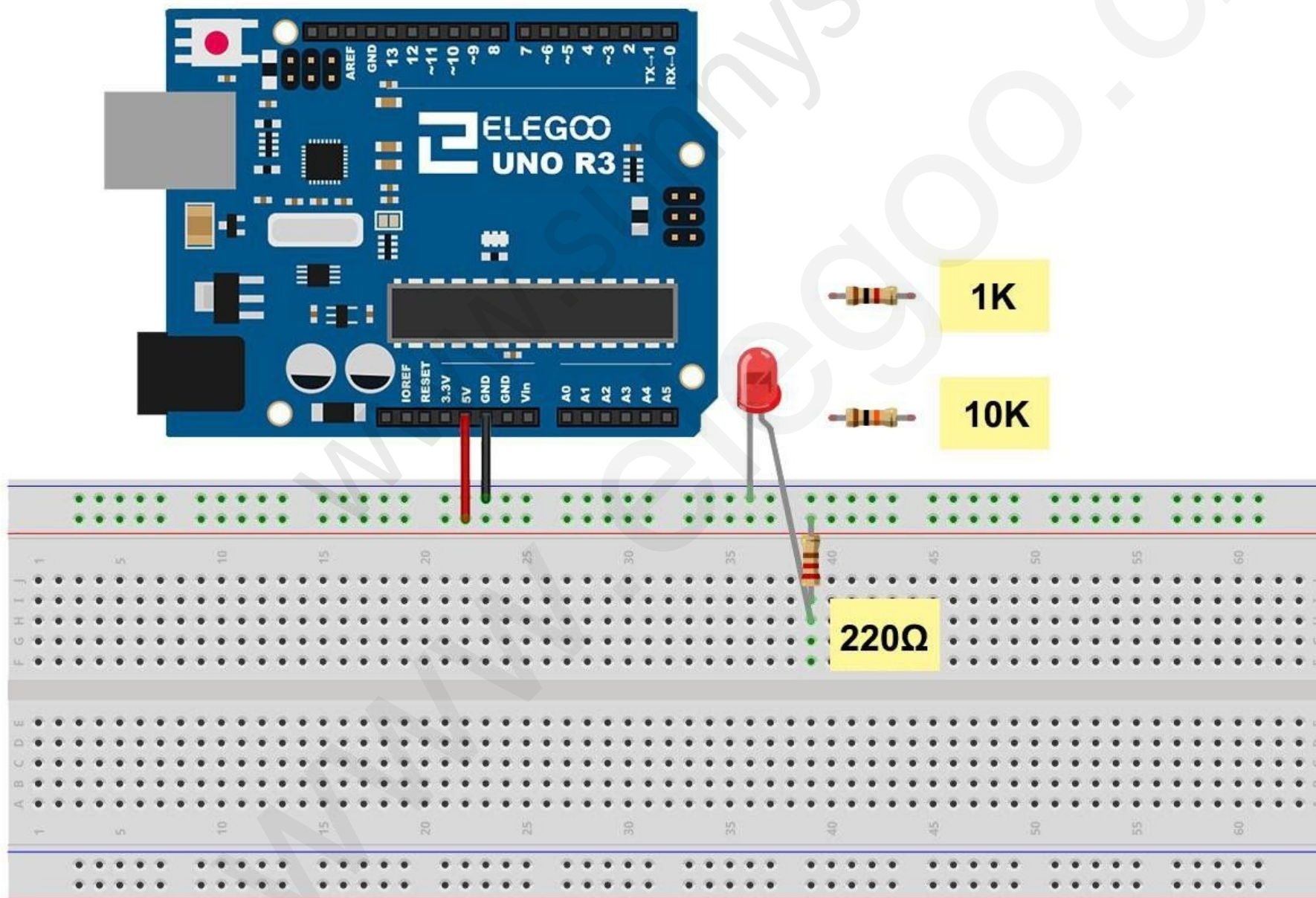


Diagrama de cablare



UNO este o sursă adecvată de 5 V pe care o folosim pentru a alimenta LED-ul și rezistorul. Cu UNO nu trebuie să faceți nimic altceva decât să îl conectați la un cablu USB.

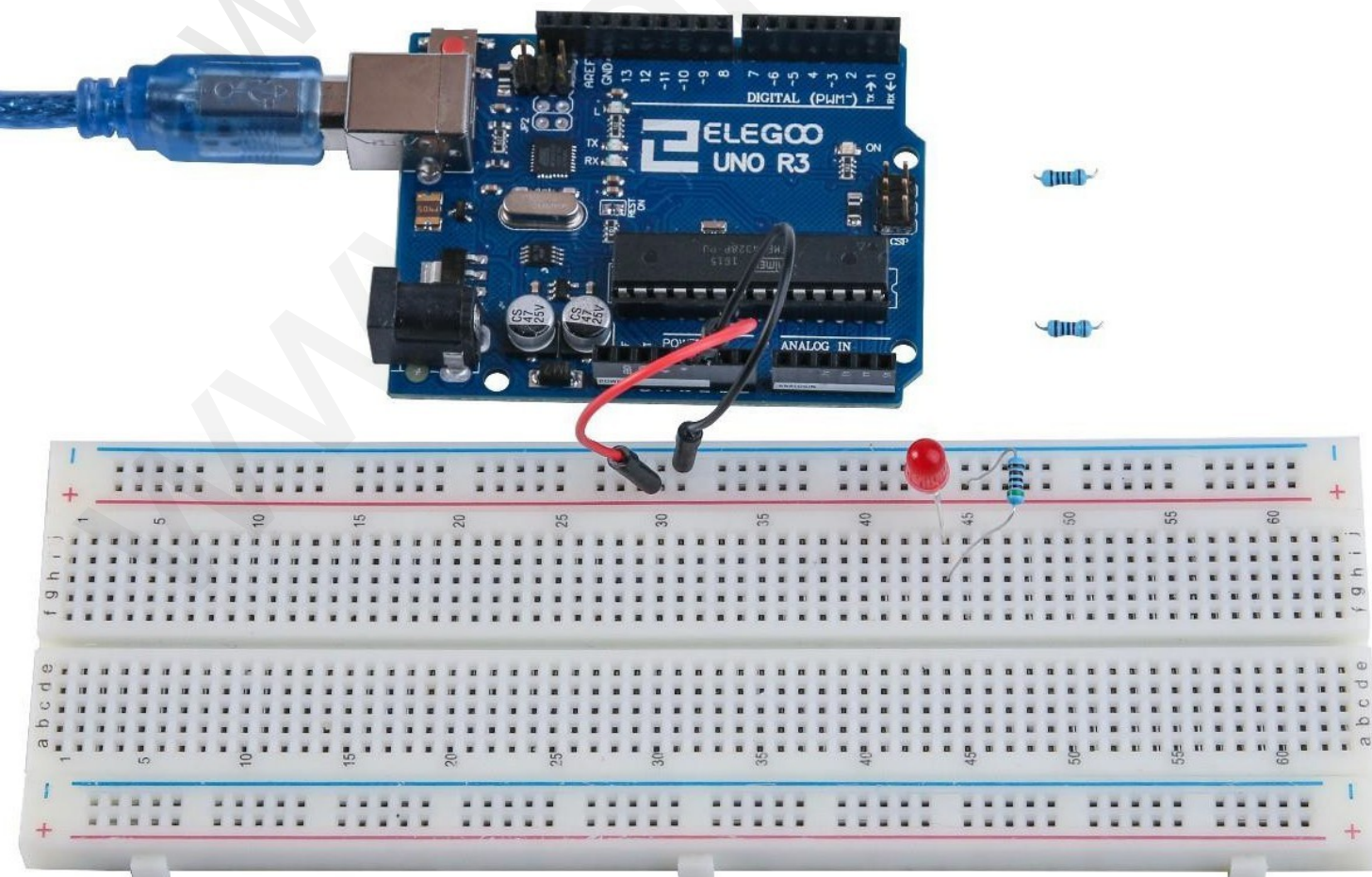
Cu un rezistor de $220\ \Omega$ în poziție, LED-ul ar trebui să fie destul de luminos. Dacă înlocuiți rezistorul de $220\ \Omega$ cu un rezistor de $1\ \text{k}\Omega$, LED-ul va fi ușor mai slab. În cele din urmă, cu un rezistor de $10\ \text{k}\Omega$ în loc, LED-ul va fi aproape vizibil. Scoateți firul roșu de jumper de pe breadboard și atingeți-l în gaură și scoateți-l pentru a acționa ca un comutator. Diferența ar trebui să fie doar vizibilă.

În acest moment aveți 5 V pe un picior al rezistorului, celălalt picior al rezistorului merge la partea pozitivă a LED-ului, iar cealaltă parte a LED-ului merge la GND. Cu toate acestea, dacă mutăm rezistorul astfel încât să fie în spatele LED-ului, așa cum se arată mai jos, LED-ul va fi în continuare aprins.

Probabil că veți dori să puneți la loc rezistența de $220\ \Omega$.

Nu contează pe ce parte a LED-ului plasăm rezistorul, atâta timp cât acesta este acolo.

Exemplu de imagine



Lecția 4 LED-uri RGB

Prezentare generală

LED-urile RGB sunt o modalitate simplă și distractivă de a adăuga puțină culoare proiectelor dumneavoastră. Deoarece acestea sunt ca 3 LED-uri obișnuite într-unul singur, modul în care sunt utilizate și cablate nu este foarte diferit.

Acestea sunt de obicei disponibile în 2 versiuni: variante: cu anod comun sau cu catod comun.

Anodul comun utilizează 5 V pe pinul comun, în timp ce catodul comun este se conectează la sol.

Ca în cazul oricărui LED, trebuie să conectăm mai multe rezistențe (3 în total) pentru a limita curentul absorbit.

În schița noastră, începem cu LED-ul în roșu, apoi trecem la verde, apoi la albastru și, în final, înapoi la roșu. Acest lucru ne va duce prin majoritatea culorilor care pot fi obținute.

Componentă necesară:

- (1) x Elegoo Uno R3
- (1) x 830 puncte de legătură Placă de pâine
- (4) x fire M-M (fire de conectare mascul la mascul)
- (1) x LED RGB
- (3) rezistențe x 220 ohm

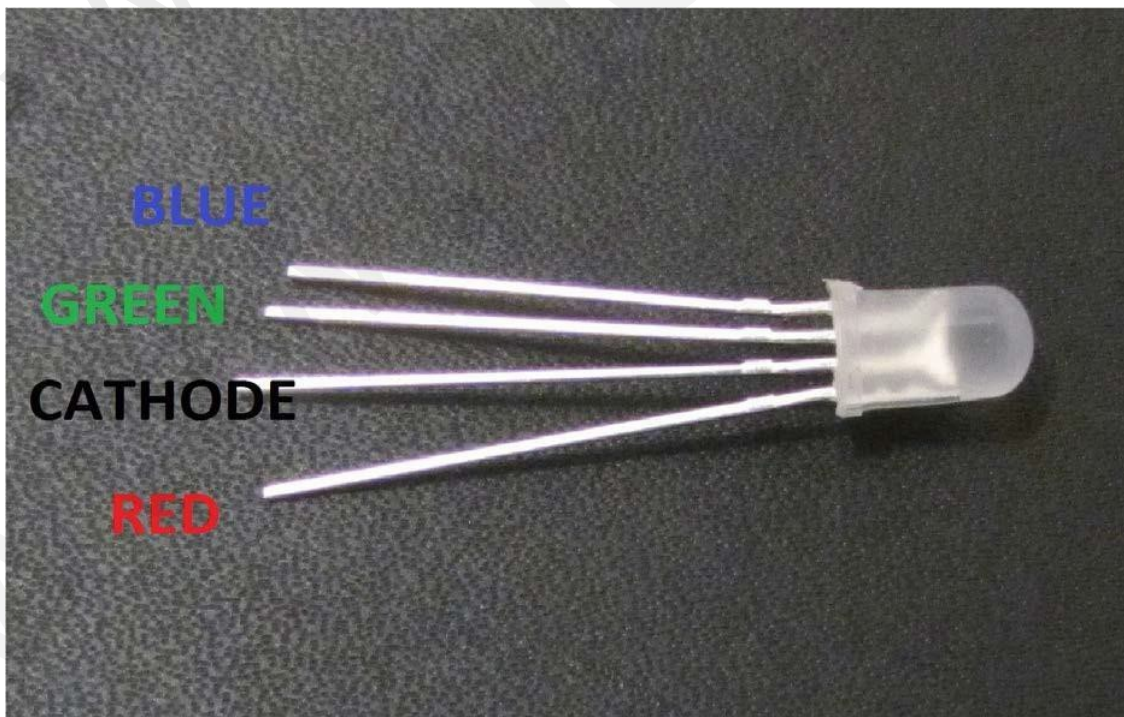
Introducere în dosar

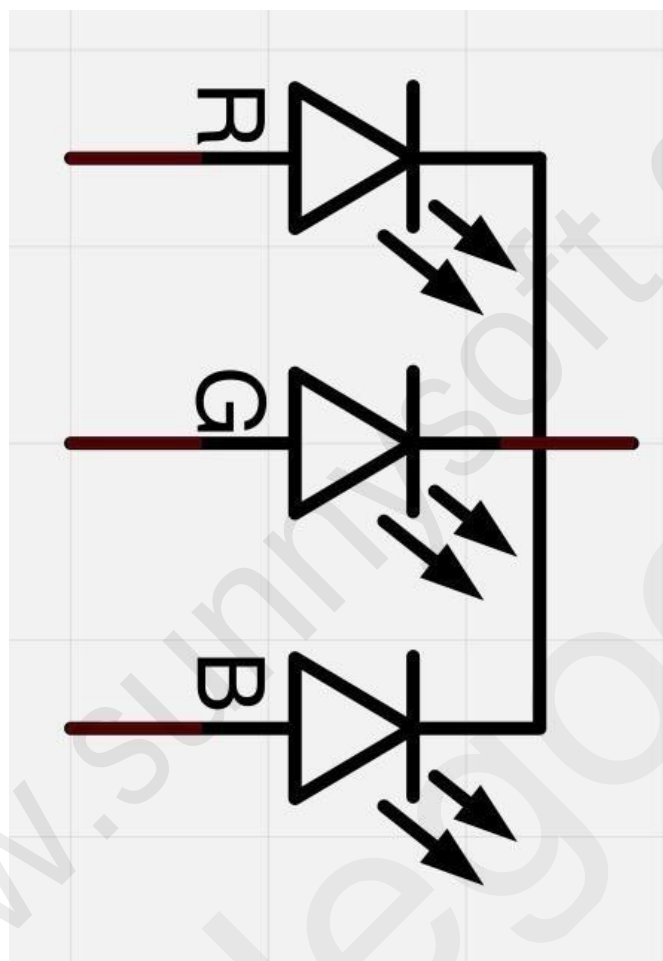
RGB:

La prima vedere, LED-urile RGB (roșu, verde și albastru) arată la fel ca LED-urile obișnuite. Cu toate acestea, în interiorul unui pachet obișnuit de LED-uri există de fapt trei LED-uri, unul roșu, unul verde și da, unul albastru. Prin controlul luminozității fiecărui LED, puteți amesteca practic orice culoare doriți.

Amestecăm culorile în același mod în care ați amesteca culorile pe o paletă - prin ajustarea luminozității fiecăruia dintre cele trei LED-uri. Cel mai complicat mod de a face acest lucru ar fi să folosim rezistențe de valori diferite (sau rezistențe variabile), așa cum am făcut în Lecția 2, dar este mult de lucru! Din fericire pentru noi, placa UNO R3 are o funcție `analogWrite` pe care o puteți utiliza cu pinii marcați cu simbolul ~ pentru a emite cantități variabile de energie către LED-urile corespunzătoare.

LED-ul RGB are patru fire. Un fir duce la conexiunea pozitivă a fiecăruia dintre LED-urile individuale din pachet, iar un fir este conectat la toate cele trei părți negative ale LED-ului.





În fotografii puteți vedea 4 electrozi LED. Fiecare pin separat pentru verde, albastru sau roșu se numește anod. Întotdeauna conectați un "+" la acesta. Catodul se conectează la "-" (masă). Dacă îl conectați invers, LED-ul nu se va aprinde.

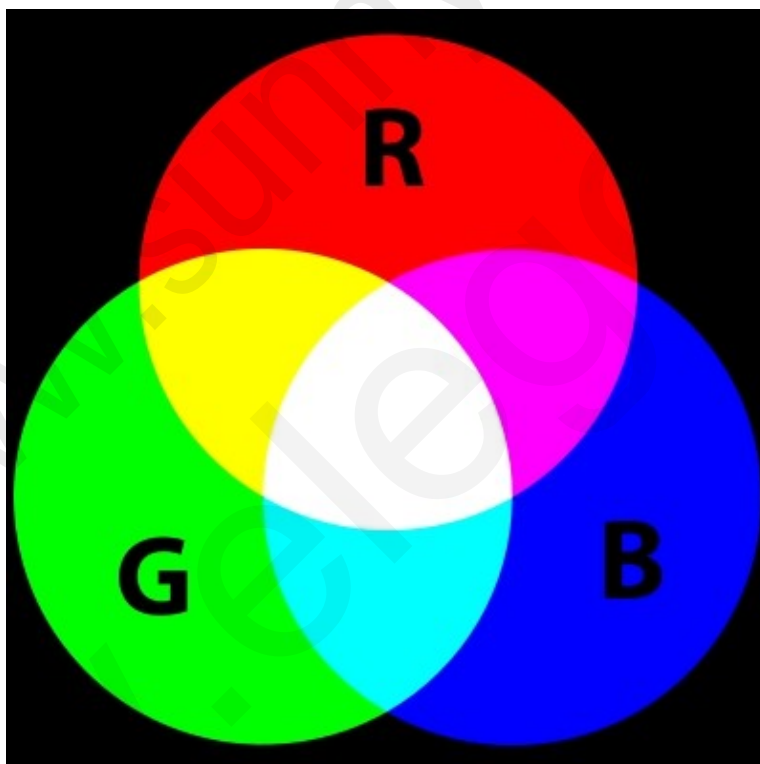
Conexiunea negativă comună a carcsei LED este al doilea pin din partea plății. Acesta este, de asemenea, cel mai lung dintre cele patru fire și va fi conectat la masă.

Fiecare LED din interiorul carcsei necesită propriul rezistor de 220Ω pentru a preveni trecerea unui curent prea mare prin acesta. Cei trei pini pozitivi ai LED-urilor (unul roșu, unul verde și unul albastru) sunt conectați la pinii de ieșire ai UNO cu ajutorul acestor rezistențe.

CULOARE: VERDE, VERDE, VERDE, VERDE, VERDE, VERDE, VERDE, VERDE, VERDE, VERDE:

Motivul pentru care puteți amesteca orice culoare atunci când variați cantitatea de lumină roșie, verde și albastră se datorează faptului că ochiul dumneavoastră are trei tipuri de receptori de lumină (roșu, verde și albastru). Ochiul și creierul procesează cantitatea de lumină roșie, verde și albastră și o transformă în spectrul de culori.

Prin utilizarea a trei LED-uri, ne jucăm într-un fel cu ochiul. Aceeași idee este utilizată în televizoare, unde punctele de culoare roșu, verde și albastru sunt plasate unul lângă altul pe ecranul LCD pentru a forma fiecare pixel.



Dacă setăm aceeași luminozitate a tuturor celor trei LED-uri, culoarea generală a luminii va fi albă. Dacă stingem LED-ul albastru, astfel încât doar LED-urile roșu și verde să aibă aceeași luminozitate, lumina va arăta galben.

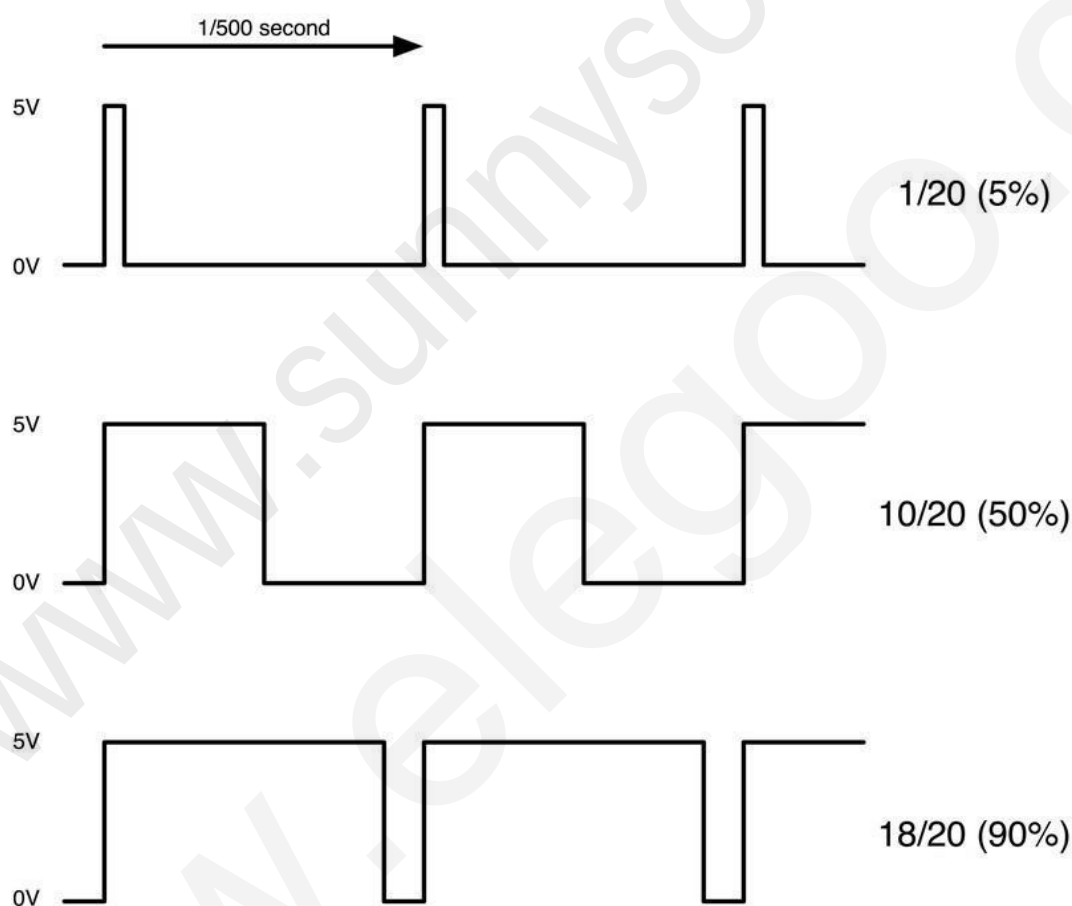
Putem controla luminozitatea fiecărei părți roșii, verzi și albastre a LED-ului separat, permițându-vă să amestecați orice culoare.

Negrul nu este atât de mult o culoare, cât absența luminii. Deci, cel mai aproape de negru pe care îl putem obține cu LED-ul nostru este să oprim toate cele trei culori.

Teorie (PWM)

Modularea lăţimii impulsurilor (PWM) este o tehnică de gestionare a puterii. Aici o folosim şi pentru a controla luminozitatea LED-urilor individuale.

Diagrama de mai jos prezintă semnalul de la unul dintre pinii PWM de pe UNO.

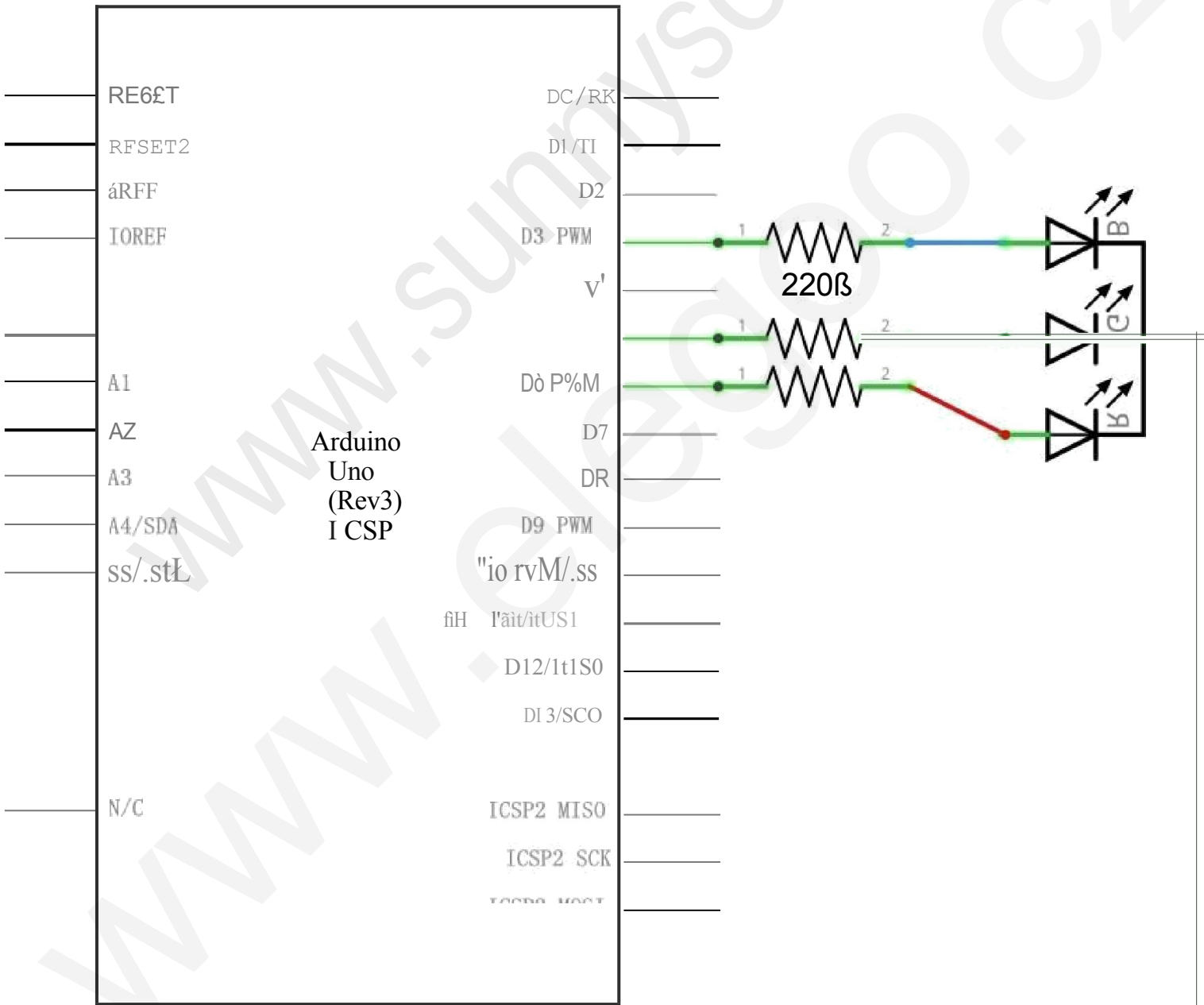


Aproximativ la fiecare 1/500 de secundă apare un impuls la ieşirea PWM. Durata acestui impuls este controlată de funcţia "analogWrite". Astfel, funcţia "analogWrite(0)" nu creează niciun impuls, iar funcţia "analogWrite(255)" creează un impuls care durează tot timpul până la sosirea următorului impuls, astfel încât ieşirea este de fapt încă activă.

Dacă introducem o valoare în analogWrite care este undeva între 0 şi 255, creăm un impuls. Dacă impulsul de ieşire este ridicat doar pentru 5% din timp, atunci ceea ce conducem va primi doar 5% din întreaga ieşire.

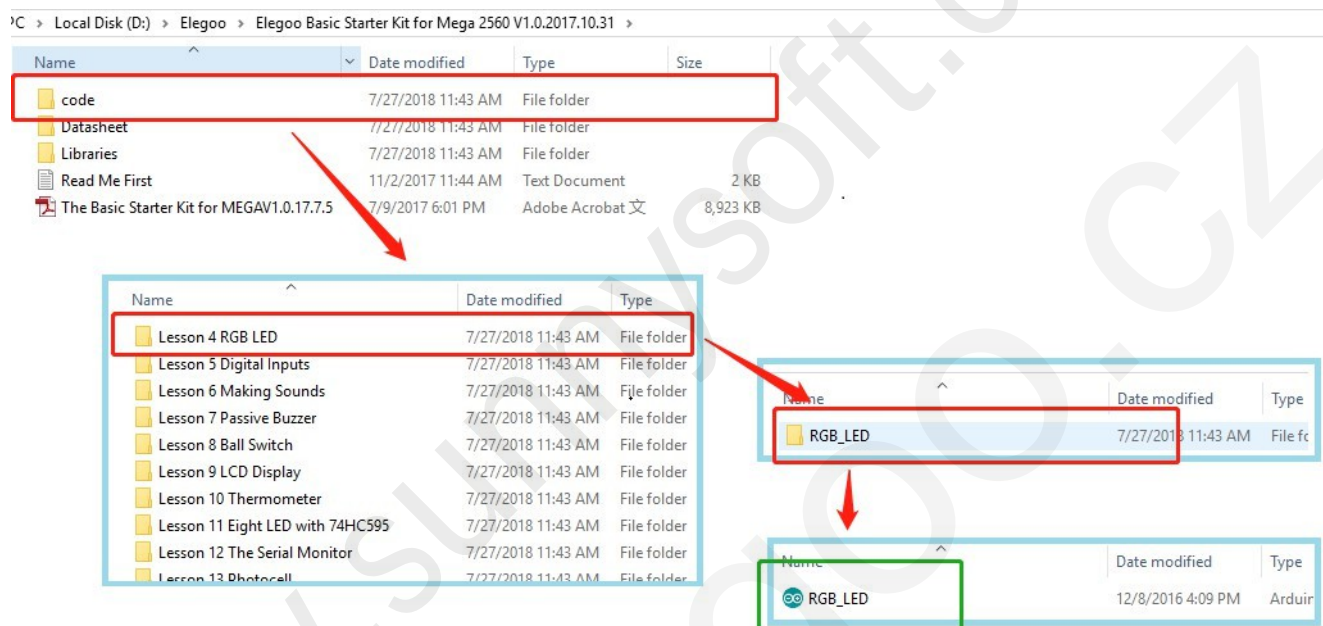
Cu toate acestea, dacă tensiunea de ieşire este de 5 V 90% din timp, atunci sarcina primeşte 90% din energia furnizată acesteia. Nu putem vedea LED-urile aprinzându-se şi stingându-se la această rată, aşa că ni se pare că ele îşi schimbă doar luminozitatea.

Conectarea
Diagrama



Cod

După conectare, deschideți Sketch în calea folderului: faceți clic pe **UPLOAD** pentru a încărca programul.



Consultați [Lecția 2](#) pentru detalii privind înregistrarea programului în caz

de erori. Schița începe prin determinarea pinilor care vor fi utilizați pentru

fiecare culoare:

// Definirea pinului

#define BLUE 3

#define GREEN 5

#define RED 6

Următorul pas este scrierea funcției "setup". După cum am învățat în lecțiile anterioare, funcția de configurare va rula o singură dată după ce Arduino este resetat. În acest caz, trebuie doar să definim cei trei pini pe care îi folosim ca ieșiri.

```
void setup()
{
  pinMode(RED, OUTPUT);
  pinMode(GREEN, OUTPUT);
  pinMode(BLUE, OUTPUT);
  digitalWrite(RED, HIGH);
  digitalWrite(GREEN, LOW);
  digitalWrite(BLUE, LOW);
}
```

Înainte de a analiza funcția "loop", să ne uităm la ultima funcție din schiță.

Variabile de definire

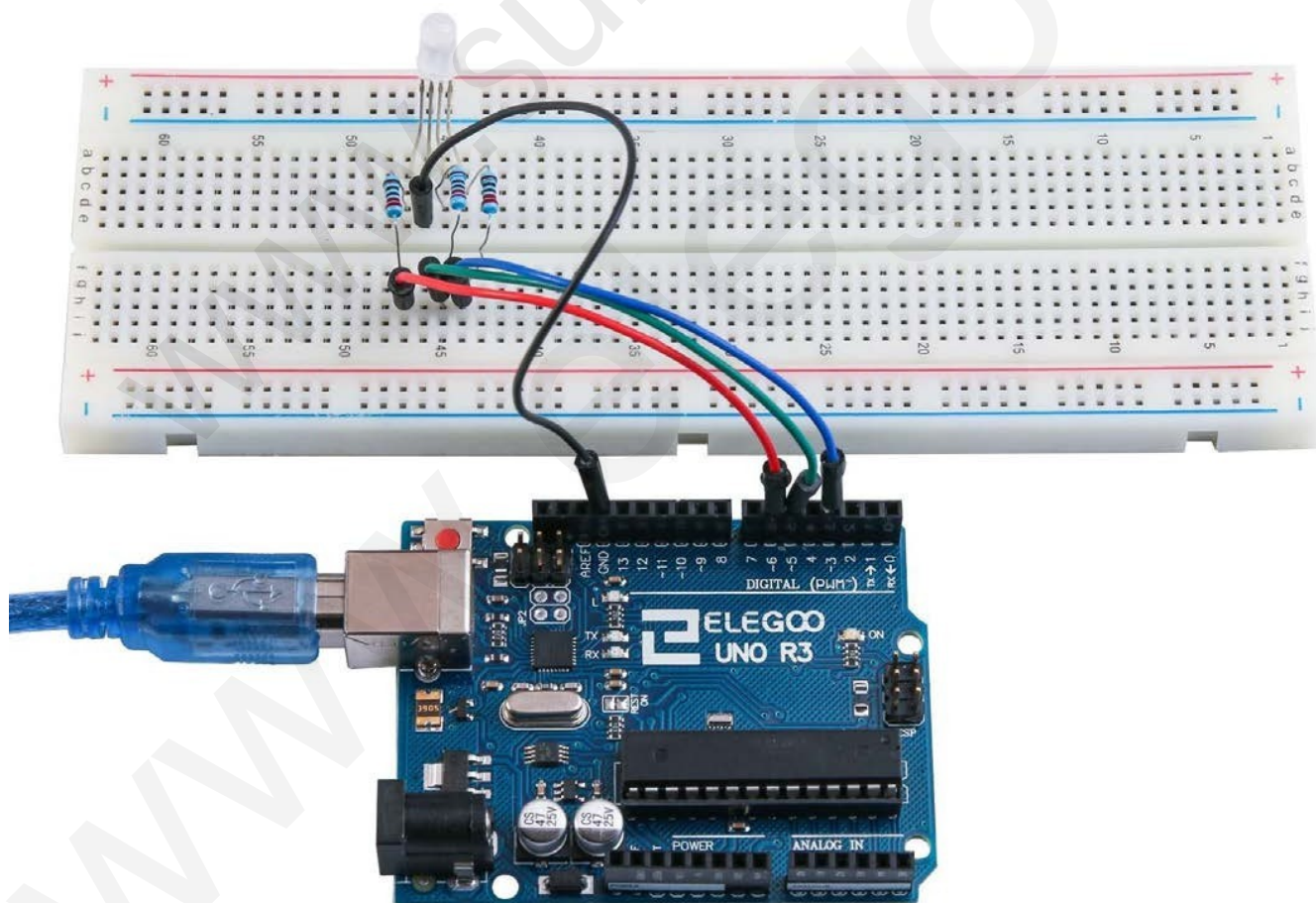
`redValue = 255; // selectați o valoare între 1 și 255 pentru a schimba culoarea.`

`greenValue = 0; blueValue = 0;`

Această funcție acceptă trei argumente, unul pentru luminozitatea LED-urilor roșu, verde și albastru. În fiecare caz, numărul va fi în intervalul 0 - 255, unde 0 înseamnă oprit și 255 înseamnă luminozitate maximă. Funcția apelează apoi "analogWrite" pentru a seta luminozitatea fiecărui LED.

Încercați să adăugați câteva culori personalizate la schiță și vedeți ce efect are LED-ul.

Exemplu de imagine



Lecția 5 Intrări digitale

Prezentare generală

În această lecție, veți învăța cum să utilizați butoane cu intrări digitale pentru a aprinde și stinge LED-uri.

Apăsarea butonului aprinde LED-ul, iar apăsarea celui de-al doilea buton stinge LED-ul.

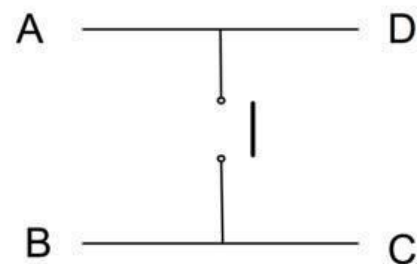
Componentă necesară:

- (1) x Elegoo Uno R3
- (1) x 830 puncte de legare Placă de sârmă
- (1) x 5 mm LED roșu
- (1) x 220 ohmi rezistență
- (2) x întrerupătoare cu buton
- (7) x fire M-M (fire de conectare mascul la mascul)

Introducere în dosar

COMUTATOARE CU BUTON:

Întrerupătoarele sunt componente foarte simple. Atunci când apăsați un buton sau rotiți o pârghie, ele conectează două contacte astfel încât electricitatea să poată trece prin ele. Micile comutatoare tactile utilizate în această lecție au patru conexiuni, ceea ce poate fi puțin derutant.



De fapt, există doar două conexiuni electrice. În interiorul carcasei comutatorului, pinii B și C, precum și A și D sunt conectați împreună.

Diagrama

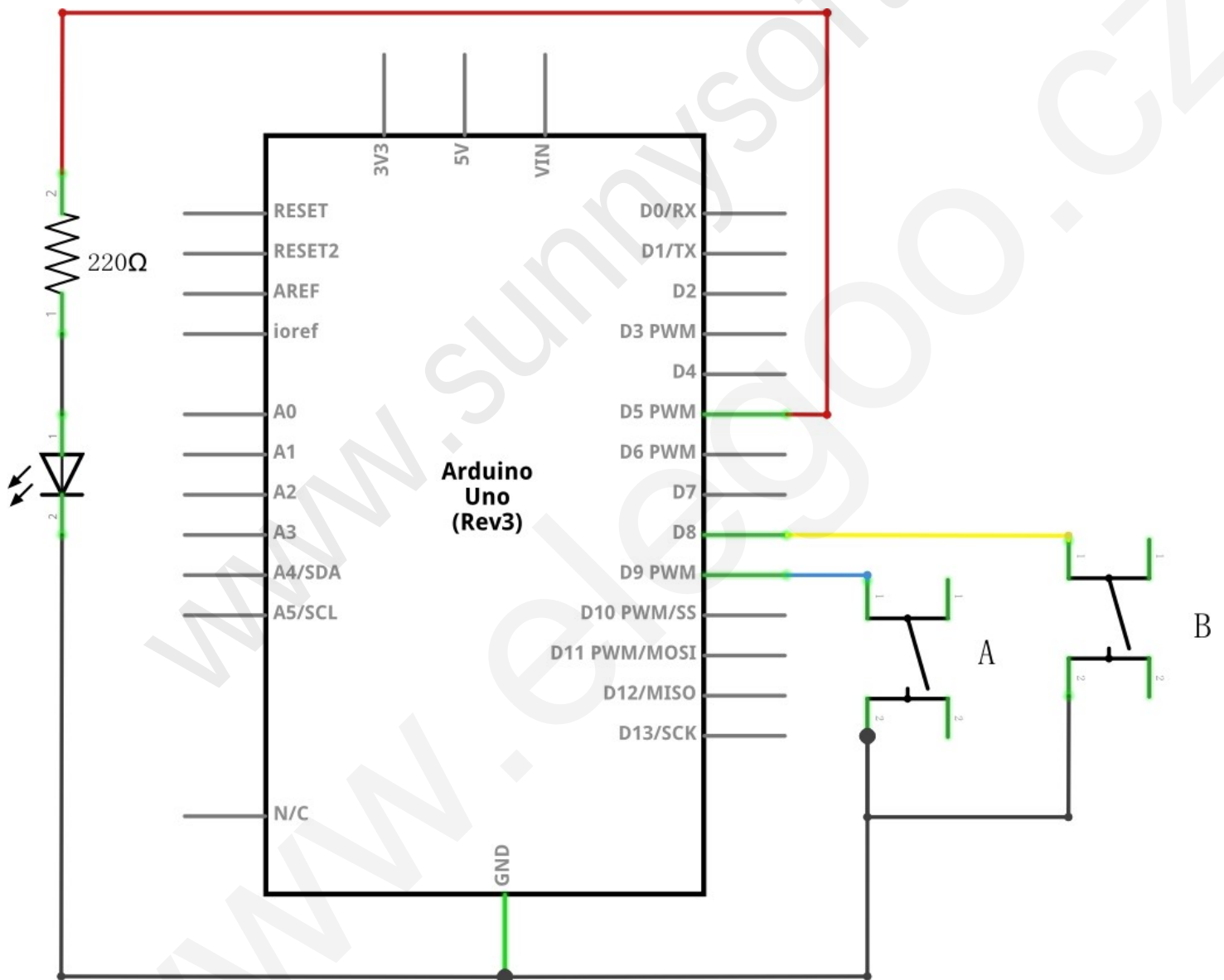
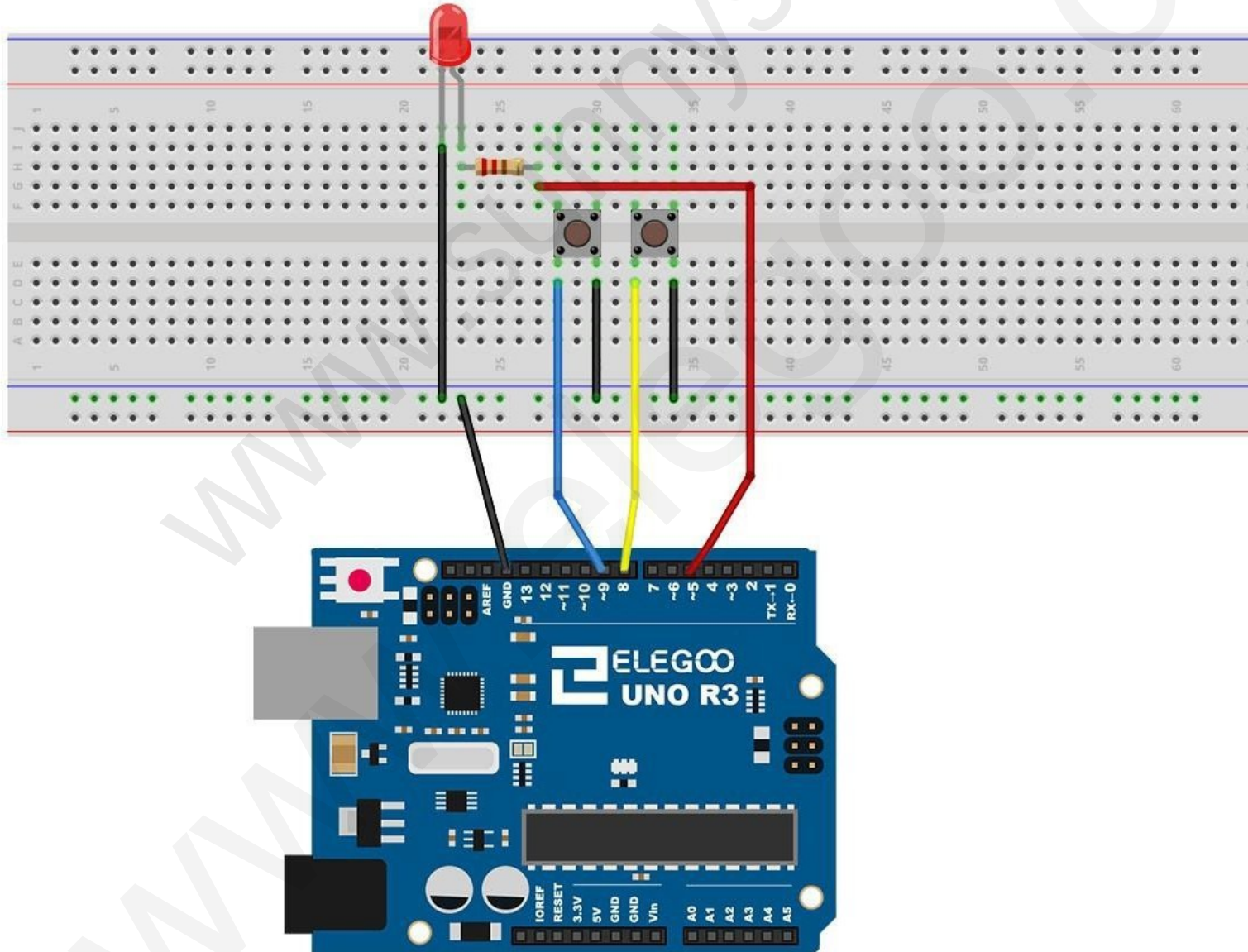


Diagrama de cablare



Deși corpurile comutatoarelor sunt pătrate, știfturile ies din părțile opuse ale comutatorului. Aceasta înseamnă că știfturile vor fi suficient de depărtate numai dacă sunt poziționate corect pe placă.

Rețineți că LED-ul trebuie să aibă firul negativ mai scurt la stânga.

Cod

După conectare, deschideți programul din folderul de cod - [Lecția 5 Intrări digitale](#) și apăsați butonul **UPLOAD** pentru a încărca programul. Dacă primiți un mesaj de eroare, mergeți la [Lecția 2](#) pentru detalii privind tutorialul de încărcare a programului.

Încărcați schița pe placa UNO. Apăsați butonul din stânga pentru a aprinde LED-ul și apăsați butonul din dreapta pentru a-l stinge.

În prima parte a schiței sunt definite trei variabile pentru cei trei pini care urmează să fie utilizați. "ledPin" este pinul de ieșire, iar "buttonApin" se va referi la comutatorul cel mai apropiat de partea superioară a plăcii și "buttonBpin" la celălalt comutator.

Funcția "setup" definește ledPin ca OUTPUT ca de obicei, dar acum trebuie să ne ocupăm de două intrări. În acest caz, vom utiliza setarea pinMode la "INPUT_PULLUP" după cum urmează:

```
pinMode(buttonApin, INPUT_PULLUP);  
pinMode(buttonBpin, INPUT_PULLUP);
```

Modul pinului INPUT_PULLUP înseamnă că pinul trebuie utilizat ca intrare, dar dacă nu este conectat nimic altceva la acesta, ar trebui să fie "tras în sus" la HIGH. Cu alte cuvinte, valoarea implicită pentru intrare este HIGH, cu excepția cazului în care este tras în LOW ca urmare a apăsării unui buton.

Prin urmare, comutatoarele sunt conectate la GND. Atunci când comutatorul este apăsător, acesta conectează pinul de intrare la GND, deci nu mai este HIGH.

Deoarece intrarea este în mod normal HIGH și trece la LOW numai atunci când este apăsată butoane, logica este puțin întortocheată. Vom aborda acest aspect în

funcția "loop". `void loop()`

```
{  
    dacă (digitalRead(buttonApin) == LOW)  
    {  
        digitalWrite(ledPin, HIGH);
```



```
}  
dacă (digitalRead(buttonBpin) == LOW)
```

```

{
    digitalWrite(ledPin, LOW);
}
}

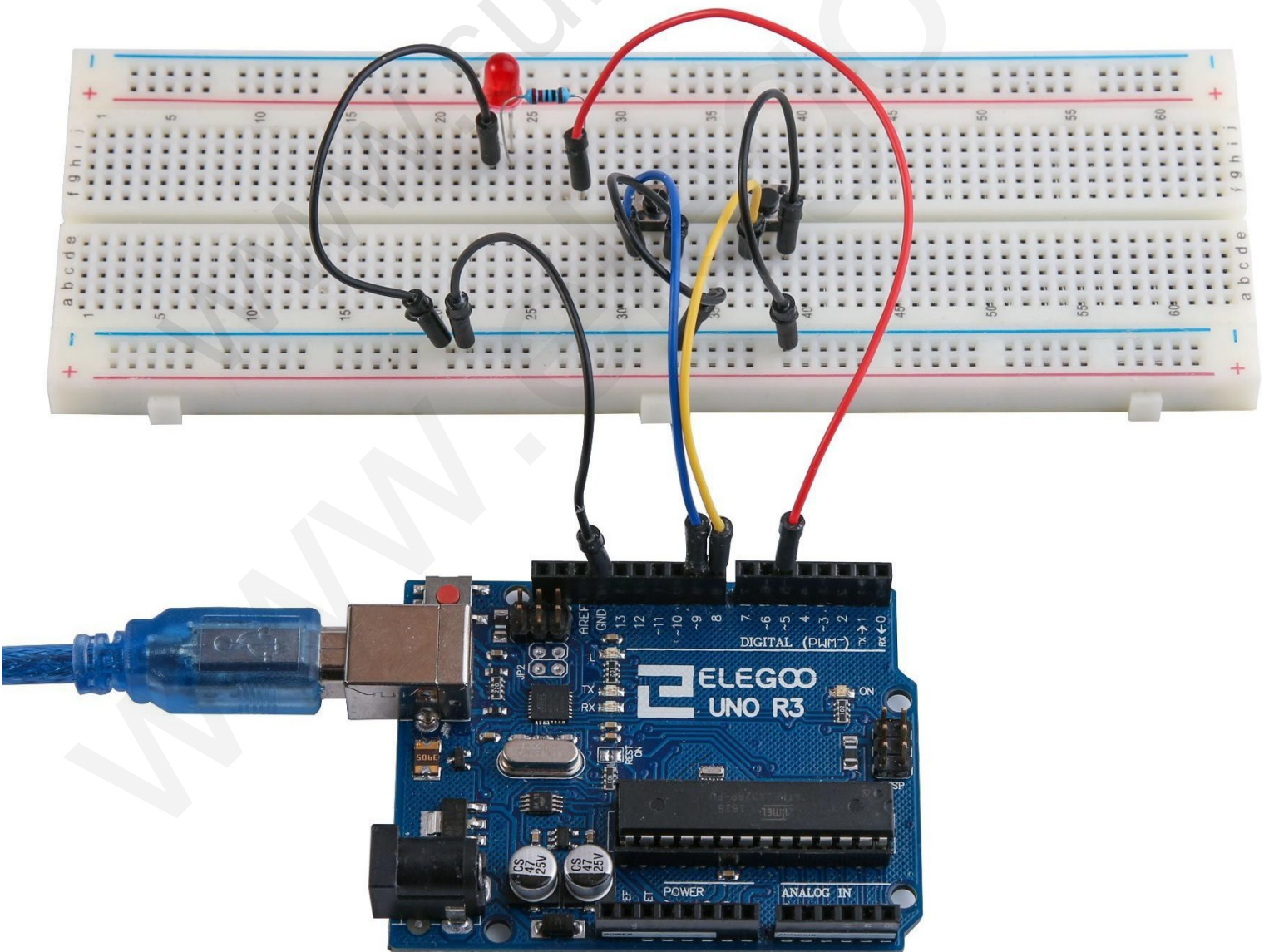
```

Există două declarații "if" în funcția "loop". Una pentru fiecare buton. Fiecare dintre ele execută "digitalRead" pe intrarea corespunzătoare.

Rețineți că dacă butonul este apăsat, intrarea corespunzătoare va fi LOW dacă butonul A scăzut, atunci "digitalWrite" de pe ledPin este activat.

În mod similar, dacă butonul B este apăsat, valoarea LOW este scrisă în pinledPin.

Exemplu de imagine



Lecția 6 Buzzer activ

Prezentare generală

În această lecție, veți învăța cum să generați un sunet folosind un buzzer activ.

Componentă necesară:

- (1) x Elegoo Uno R3
- (1) x Buzzer activ
- (2) x fire F-M (de la femelă la mascul DuPont)

Introducere în dosar

BZUCHAK:

Buzzerii electronici sunt alimentați cu curent continuu și au un circuit integrat. Acestea sunt utilizate pe scară largă în calculatoare, imprimante, copiatoare, alarme, jucării electronice, dispozitive electronice auto, telefoane, cronometre și alte produse electronice vocale. Buzoanele pot fi împărțite în active și pasive. Întoarceți pinii celor două buzzeri în sus. Cel cu PCB verde este buzzerul pasiv, în timp ce celălalt închis

NoReorzndoílumpeázsiknoisimio în sensul că buzzerul activ are un motor oscilant încorporat.

astfel încât emite un sunet atunci când este electrificată. Un buzzer pasiv nu are o astfel de sursă, astfel încât nu va suna atunci când se utilizează semnale de curent continuu; în schimb, trebuie utilizate unde pătrate cu frecvențe între 2K și 5K pentru a-l excita. Un buzzer activ este adesea mai scump decât un buzzer pasiv deoarece are mai multe oscilații circuite.



Conectarea

Diagrama

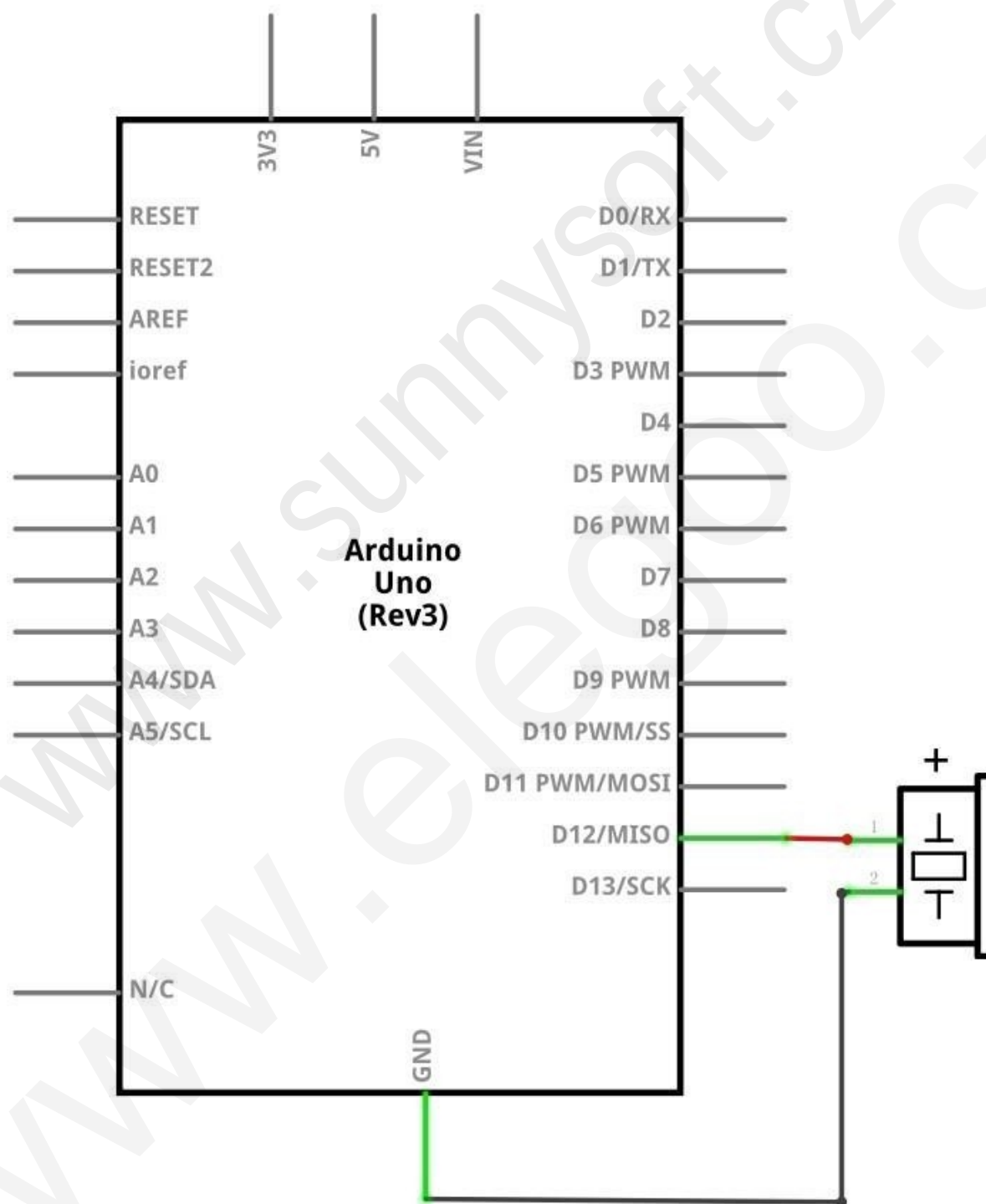
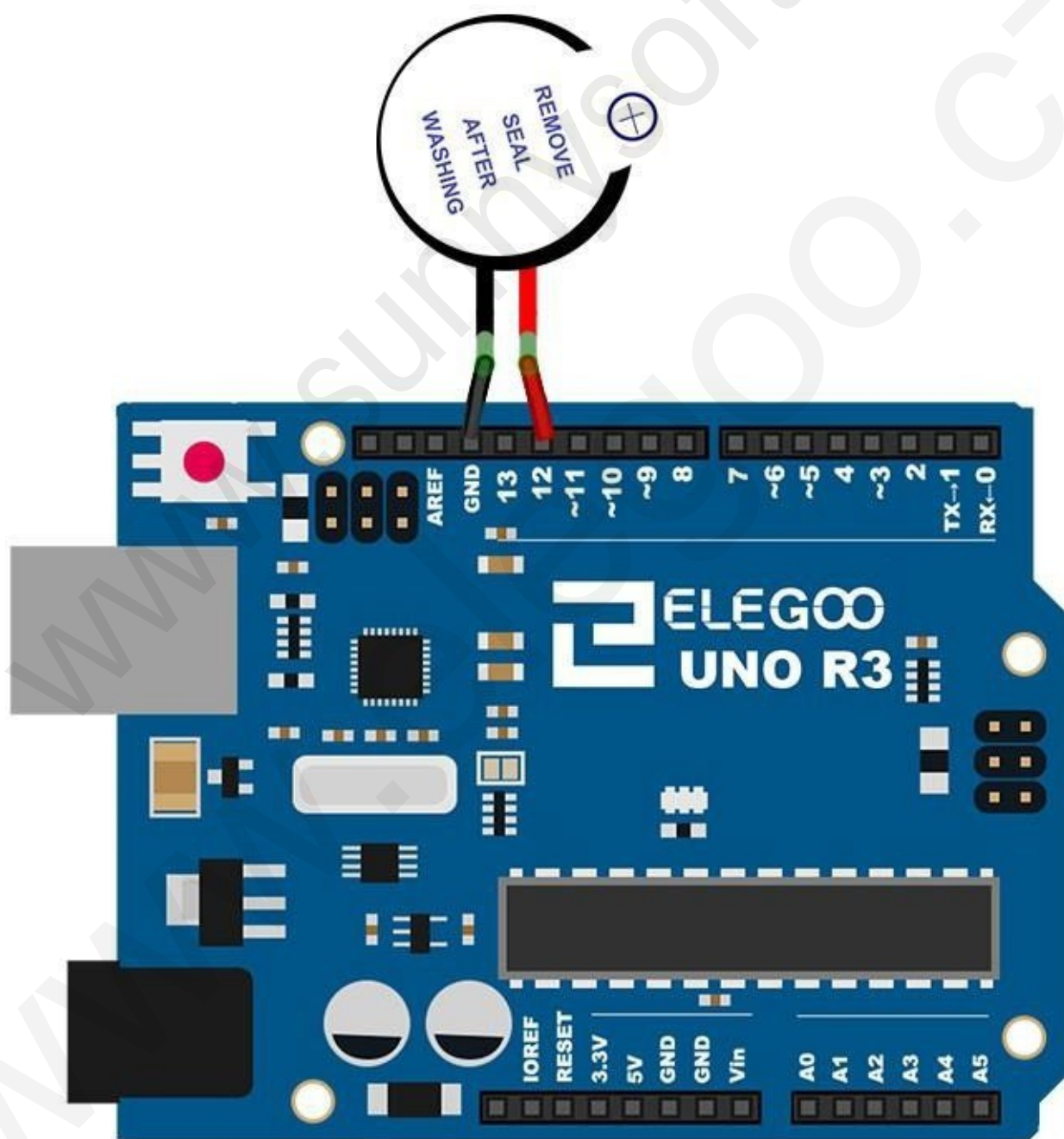


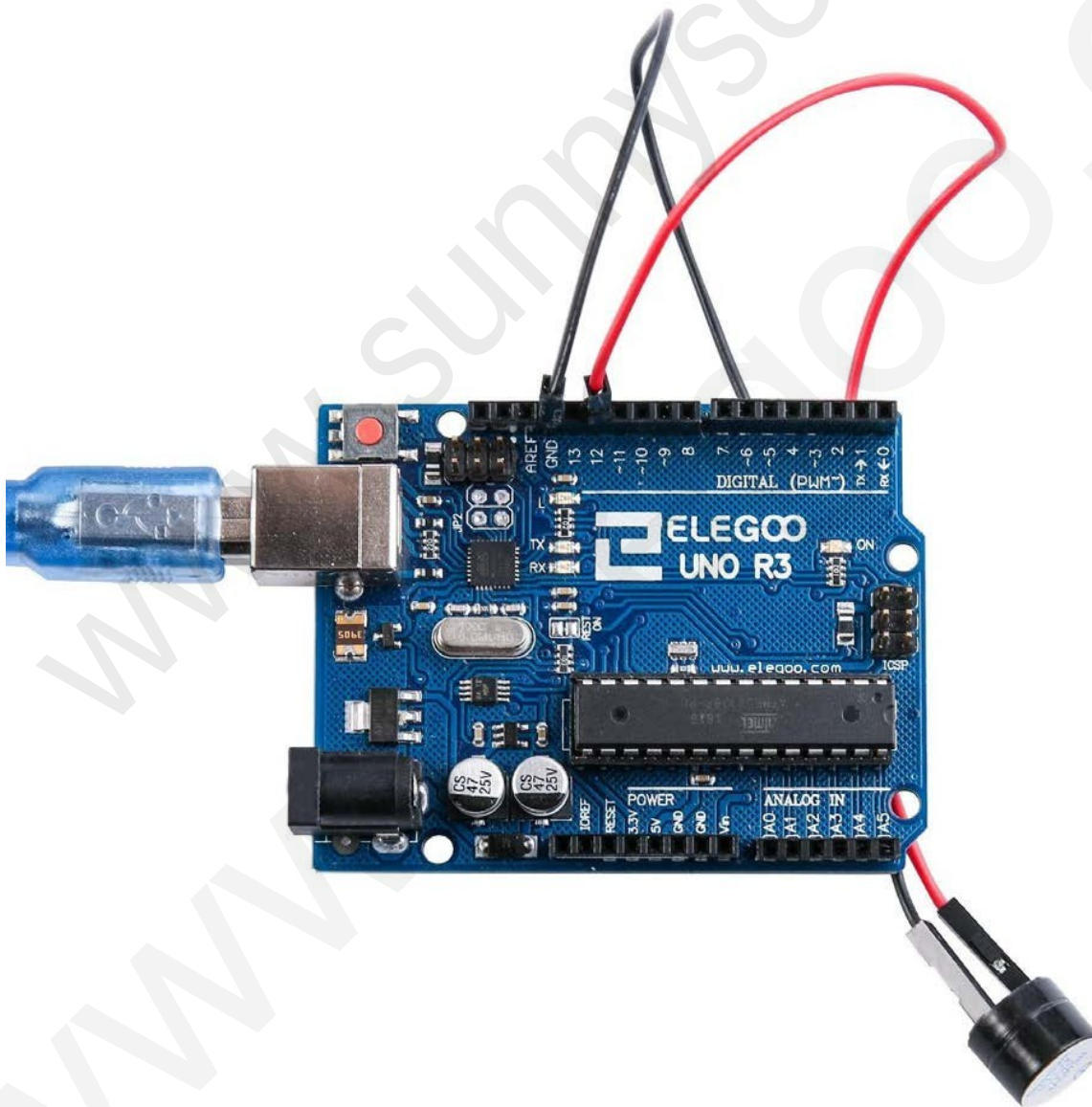
Diagrama de cablare



Cod

După conectare, deschideți programul din folderul de cod - Lecția 6 Realizarea sunetelor și faceți clic pe UPLOAD pentru a încărca programul. Consultați Lecția 2 pentru detalii despre cum să înregistrați programul dacă întâmpinați erori.

Exemplu de imagine



Lecția 7 Buzzer pasiv

Prezentare generală

În această lecție veți învăța cum să utilizați passivebuzzer.

Scopul experimentului este de a crea opt sunete diferite, fiecare având o durată de 0,5 secunde: de la Do alto (523 Hz), Re (587 Hz), Mi (659 Hz), Fa (698 Hz), So (784 Hz), La (880 Hz), Si (988 Hz) la Do înalt (1047 Hz).

Componentă necesară:

- (1) x Elegoo Uno R3
- (1) x buzzer pasiv
- (2) x fire F-M (de la femelă la mascul DuPont)

Introducere în dosar

Buzzer pasiv:

Principiul de funcționare al buzzerului pasiv este că acesta generează un sunet prin intermediul PWM care agită aerul. Prin modificarea corespunzătoare a frecvenței de vibrație, poate genera sunete diferite. De exemplu, prin emiterea unui impuls la o frecvență de 523 Hz poate genera Alto Do, prin emiterea unui impuls la o

Puteți reda o melodie utilizând soneria.

Trebuie să fim atenți să nu folosim funcția de scriere analogică () de pe placa UNO R3 pentru a genera un impuls către sonerie, deoarece impulsul de ieșire de scriere analogică () este fix. (500 Hz).



frecvență de 587 Hz poate genera Medium Re, prin emiterea unui impuls la o frecvență de 659 Hz

www.sunnysoft.cz

Conectarea

Diagrama

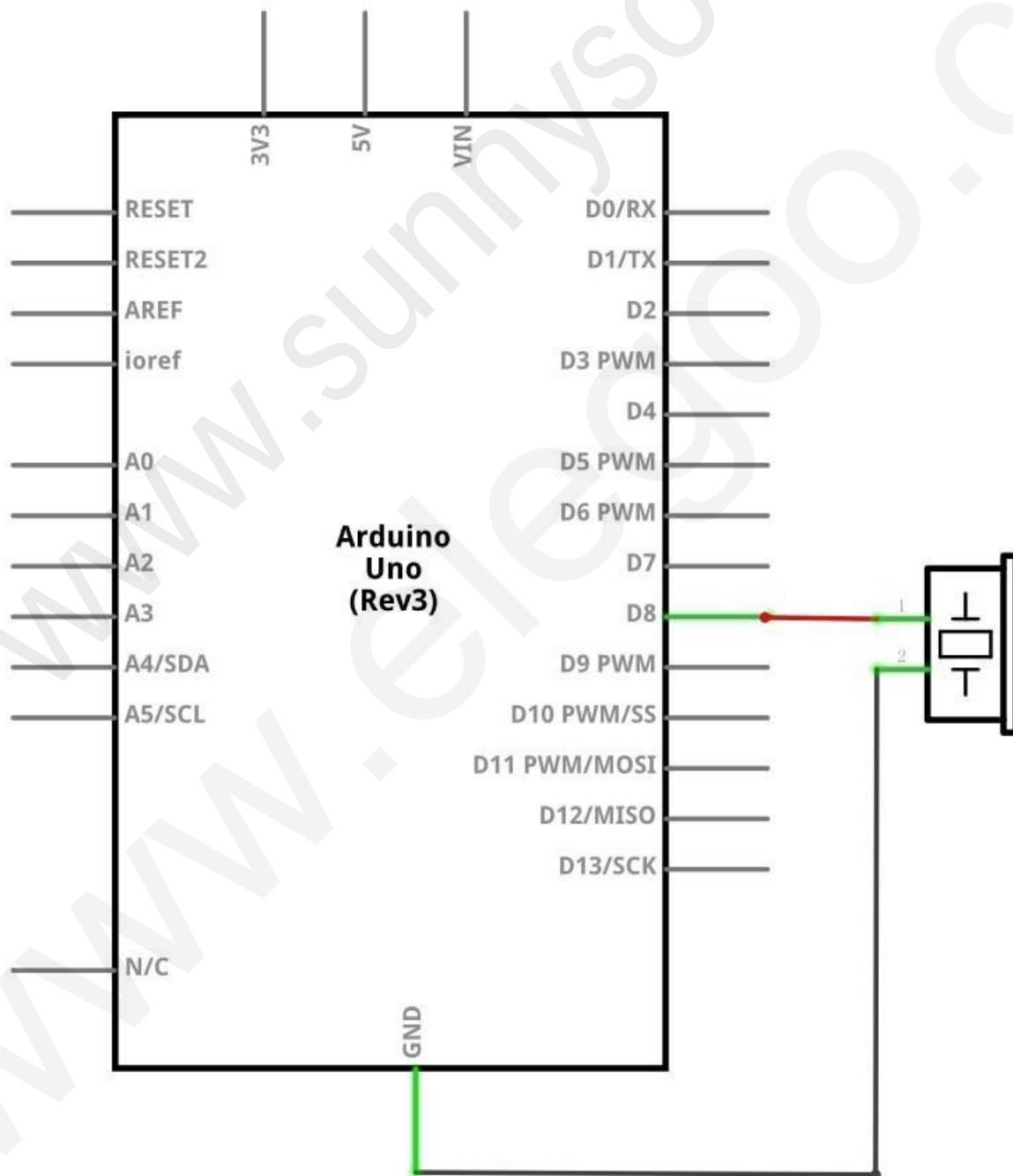


Diagrama de cablare



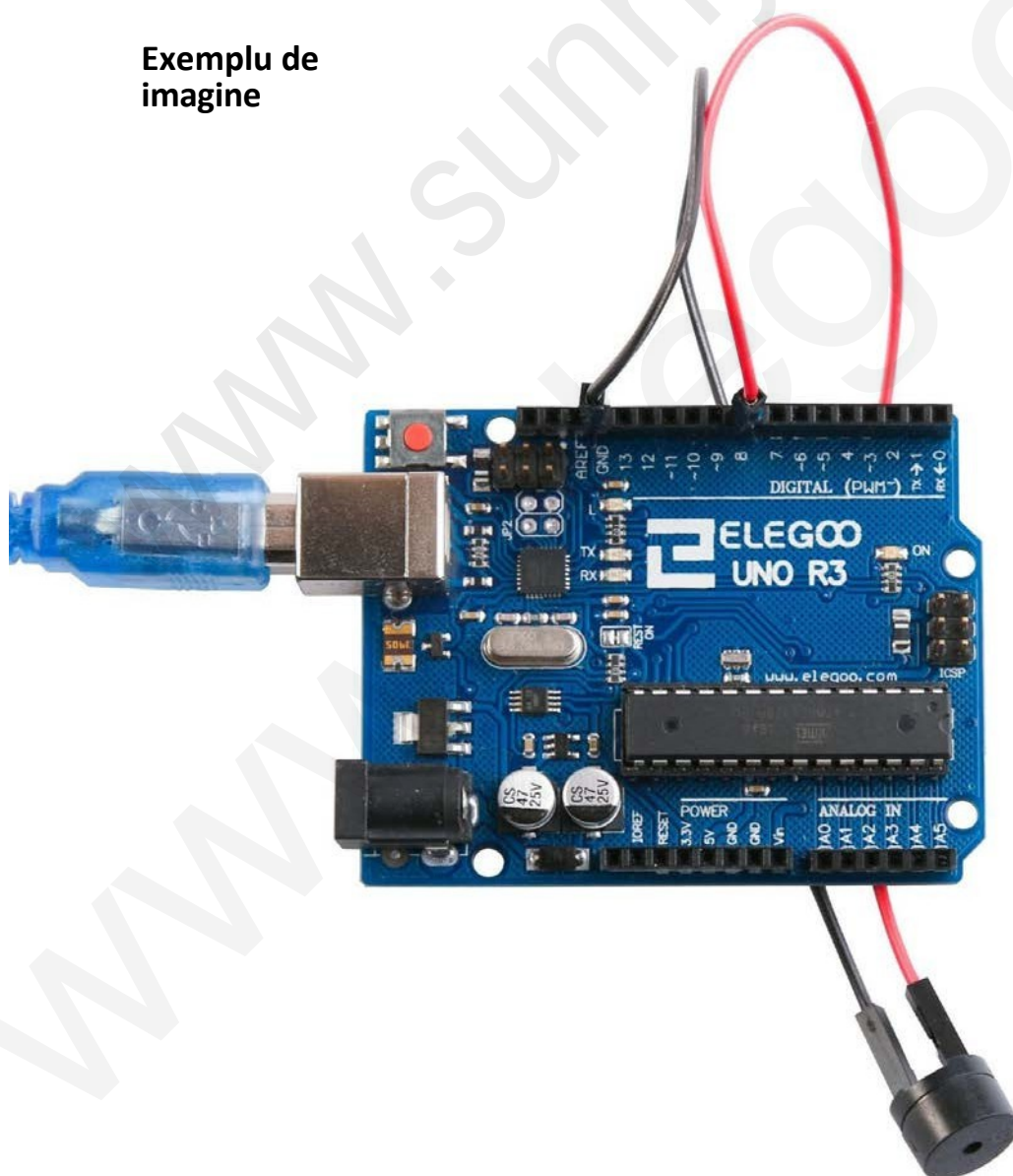
Cablarea soneriei conectată la placa UNO R3, firul roșu (pozitiv) la pinul 8, firul negru (negativ) la GND.

Cod

După conectare, deschideți programul din folderul de cod - Lecția 7 Buzzer pasiv și faceți clic pe UPLOAD pentru a încărca programul. Consultați Lecția 2 pentru detalii privind încărcarea programului dacă întâmpinați erori.

Înainte de a rula această caracteristică, asigurați-vă că aveți biblioteca instalată <pitches>, sau reinstalați-l. În caz contrar, codul dvs. nu va funcționa. Consultați Lecția1 pentru detalii privind încărcarea fișierului de bibliotecă.

Exemplu de imagine



Lecția 8 Comutator cu bilă basculantă

Prezentare generală

În această lecție, veți învăța cum să utilizați comutatorul cu bilă de înclinare pentru a detecta unghiuri mici de înclinare.

Componentă necesară:

- (1) x Elegoo Uno R3
- (1) x comutator cu bilă de înclinare
- (2) x fire F-M (de la femelă la mascul DuPont)



Introducere în dosar

Senzor de înclinare:

Senzorii de înclinare (comutatorul cu bilă de înclinare) vă permit să detectați orientarea sau înclinarea. Sunt mici, necostisitoare, consumă puțină energie și sunt ușor de utilizat. Nu se uzează în cazul utilizării corespunzătoare. Simplitatea lor le face populare pentru jucării, gadgeturi și aparate. Acestea sunt uneori denumite "comutatoare cu mercur", "comutatoare de înclinare" sau "senzori cu bilă rulantă" din motive evidente.

Acestea sunt formate, de obicei, dintr-o cavitate de un anumit tip (cilindrică este populară, deși nu întotdeauna) cu o masă conductoare liberă în interior, cum ar fi o picătură de mercur sau o bilă rulantă. Un capăt al cavității are două elemente conductoare (poli). Atunci când traductorul este orientat astfel încât acest capăt să fie orientat în jos, masa se rostogolește pe poli, scurtcircuitându-i și acționând ca un comutator.

Deși nu sunt la fel de precise și flexibile ca un accelerometru cu drepturi depline, comutatoarele de înclinare pot detecta mișcarea sau orientarea. Un alt avantaj este că cele mari pot comuta singure energia. Accelerometrele, pe de altă parte, trimit tensiuni digitale sau analogice, care trebuie apoi analizate de circuite suplimentare.

Conectarea

Diagrama

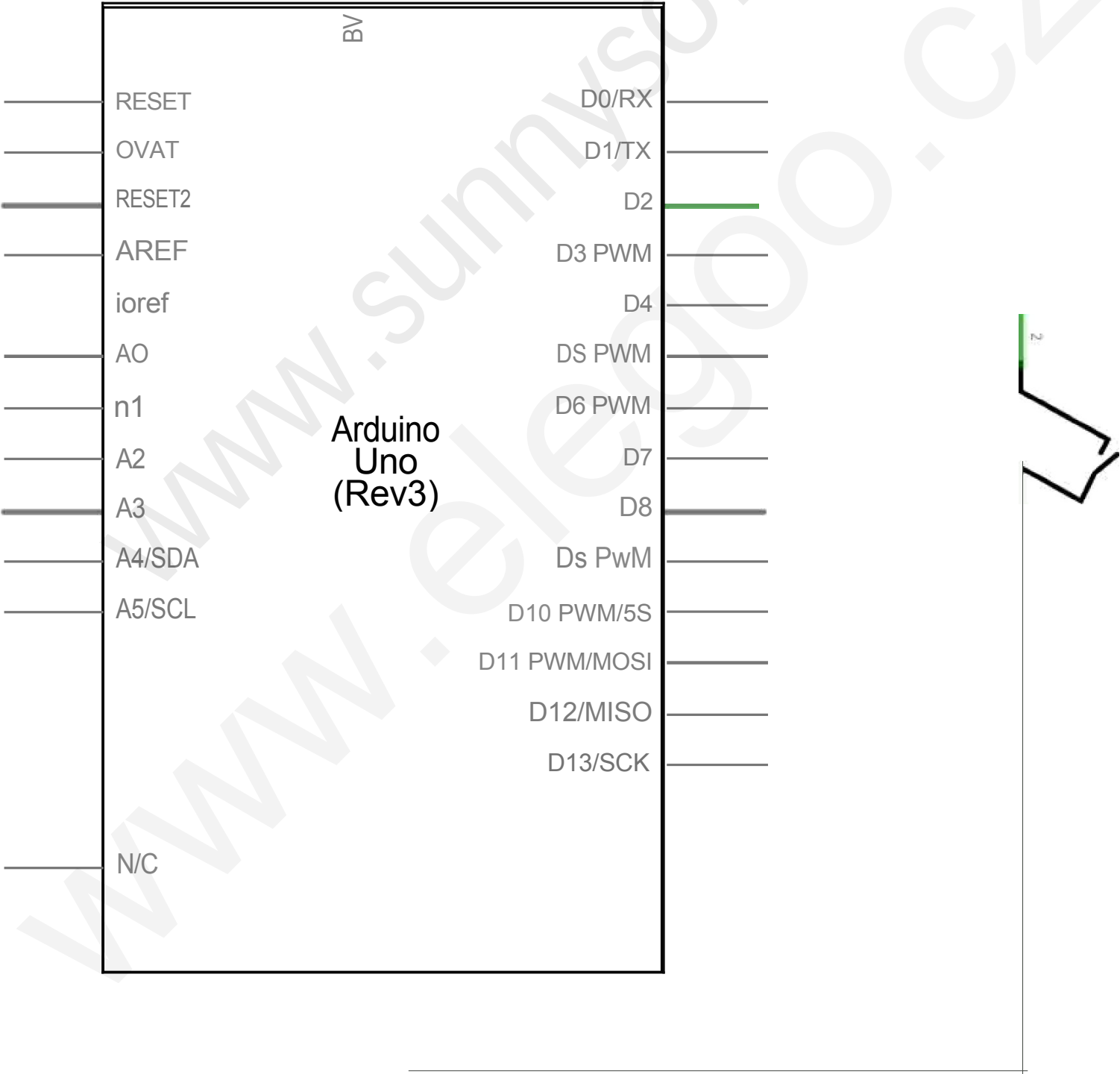
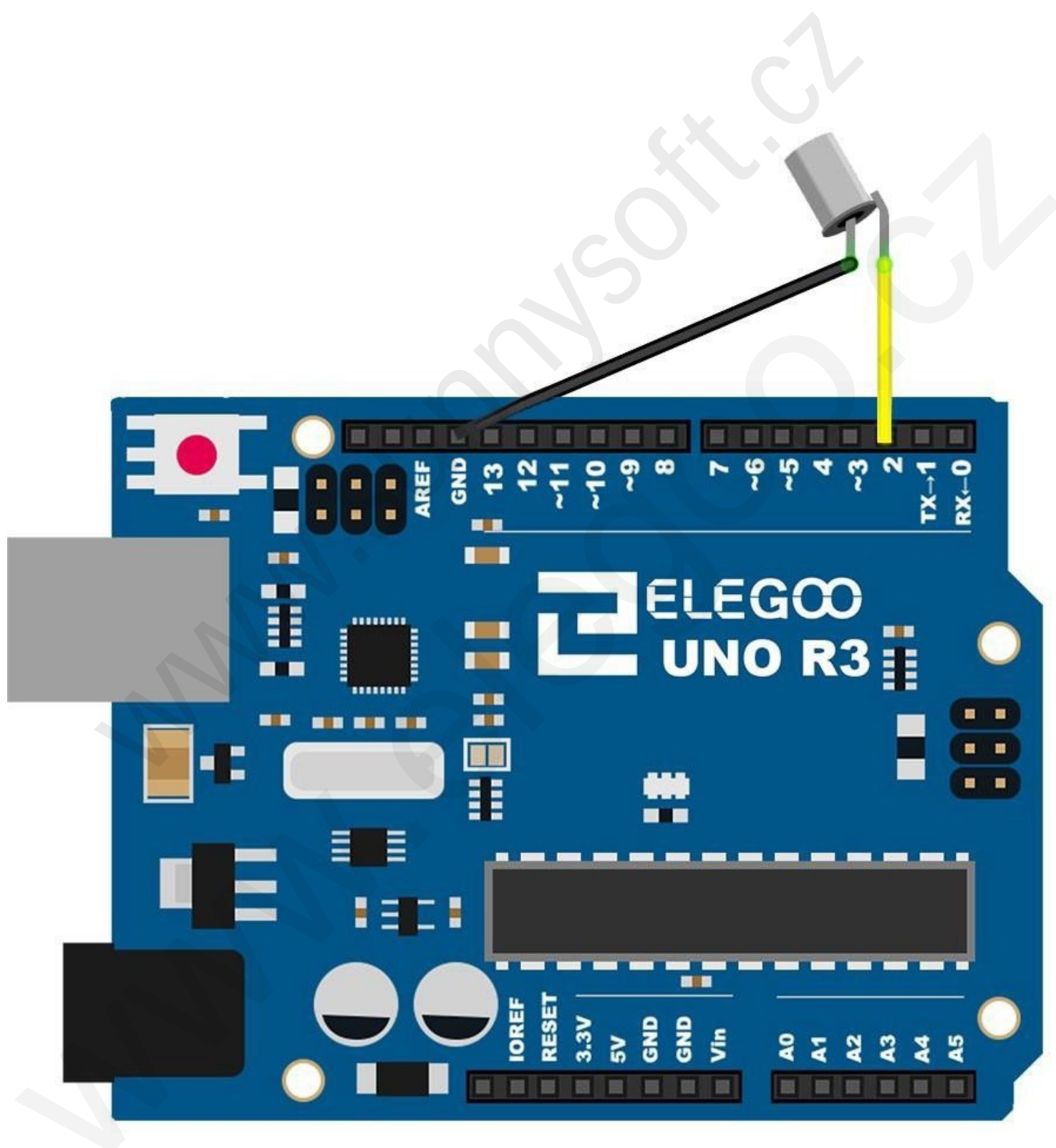


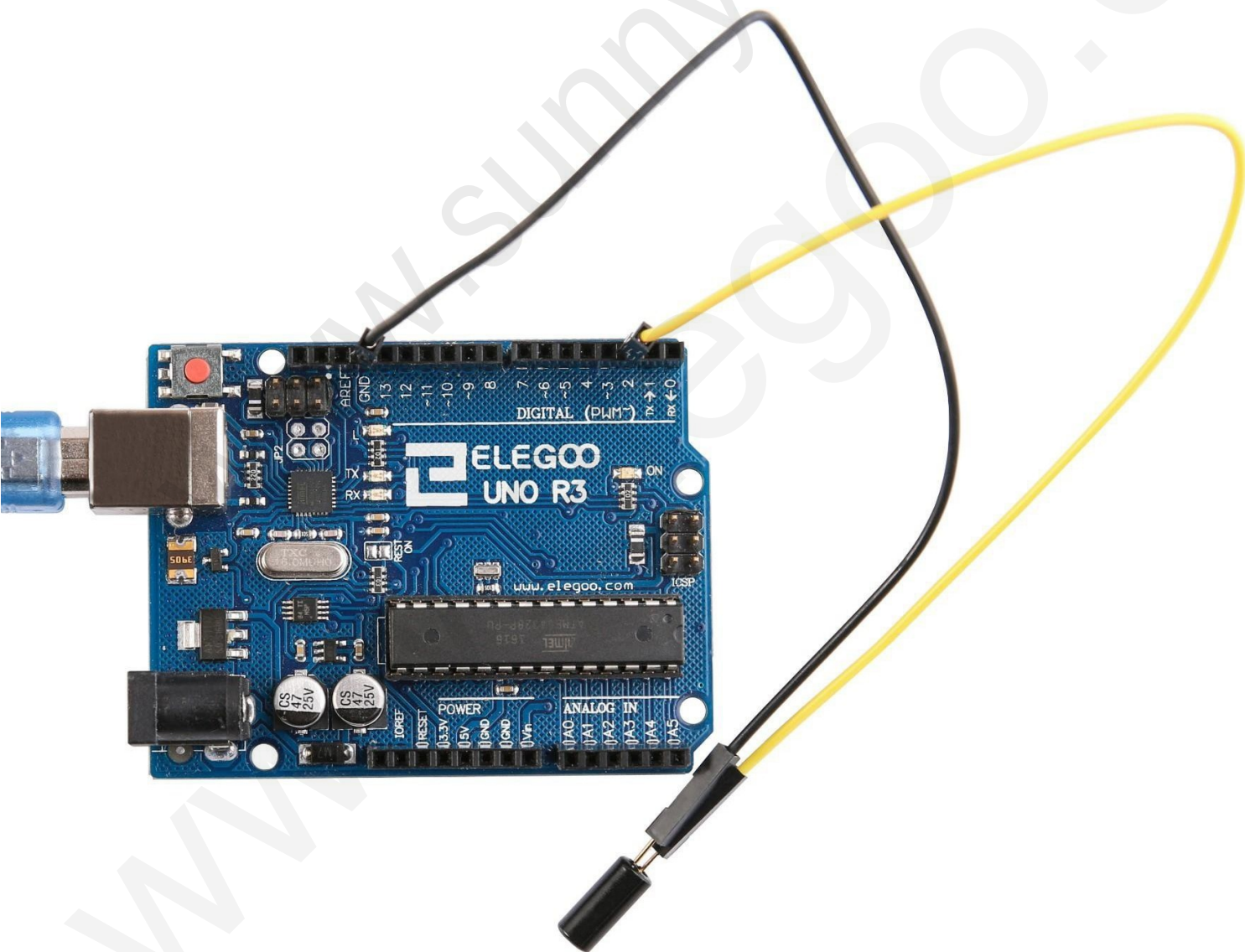
Diagrama de cablare



Cod

După conectare, deschideți programul din folderul cu codul - Lesson 8 Ball Switch și faceți clic pe **UPLOAD** pentru a încărca programul. Consultați Lecția 2 pentru detalii privind încărcarea programului dacă întâmpinați erori.

Exemplu de imagine



Lecția 9 Servo

Prezentare generală

Un servomotor este un tip de motor cu angrenaj care se poate roti doar la 180 de grade. Acesta este controlat prin trimiterea de impulsuri electrice de la placa UNO R3. Aceste impulsuri îi spun servomotorului în ce poziție să se deplaseze. Servo are trei fire, maro este firul de masă și trebuie conectat la portul UNO GND, roșu este firul de alimentare și trebuie conectat la portul 5V, iar portocaliu este firul de semnal și trebuie conectat la portul Dig #9.

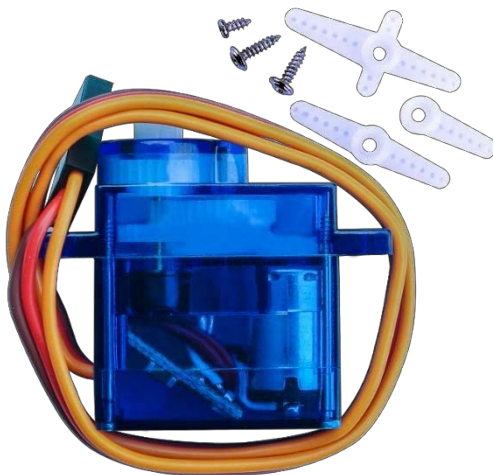
Componentă necesară:

- (1) x Elegoo Uno R3
- (1) x Servo (SG90)
- (3) x fire M-M (fire de conectare mascul la mascul)

Introducere în dosar

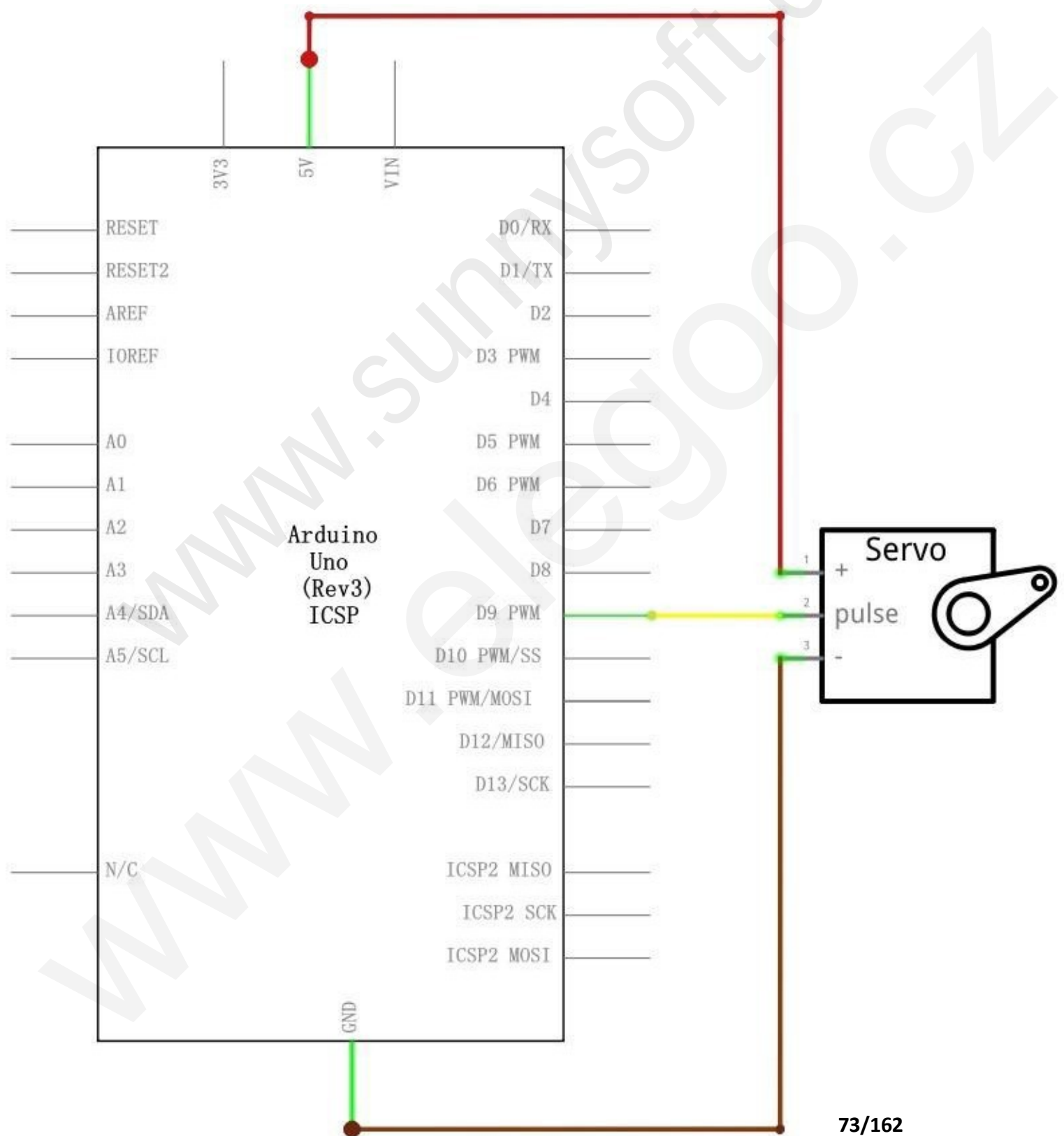
SG90

- Universal pentru conector JR și FP
- Lungimea cablului: 25 cm
- Fără sarcină; Viteza de funcționare: 0,12 s/60 grade (4,8 V), 0,10 s/60 grade (6,0 V)
- Cuplu de oprire (4,8 V): 1,6 kg/cm
- Temperatură: -30~60°C
- Lățime de bandă moartă: 5us
- Tensiune de lucru: 3,5 ~ 6 V
- Dimensiuni: 1,26 x 1,18 x 0,47 inch (3,2 x 3 x 1,2 cm)
- Greutate: 4,73 oz (134 g)



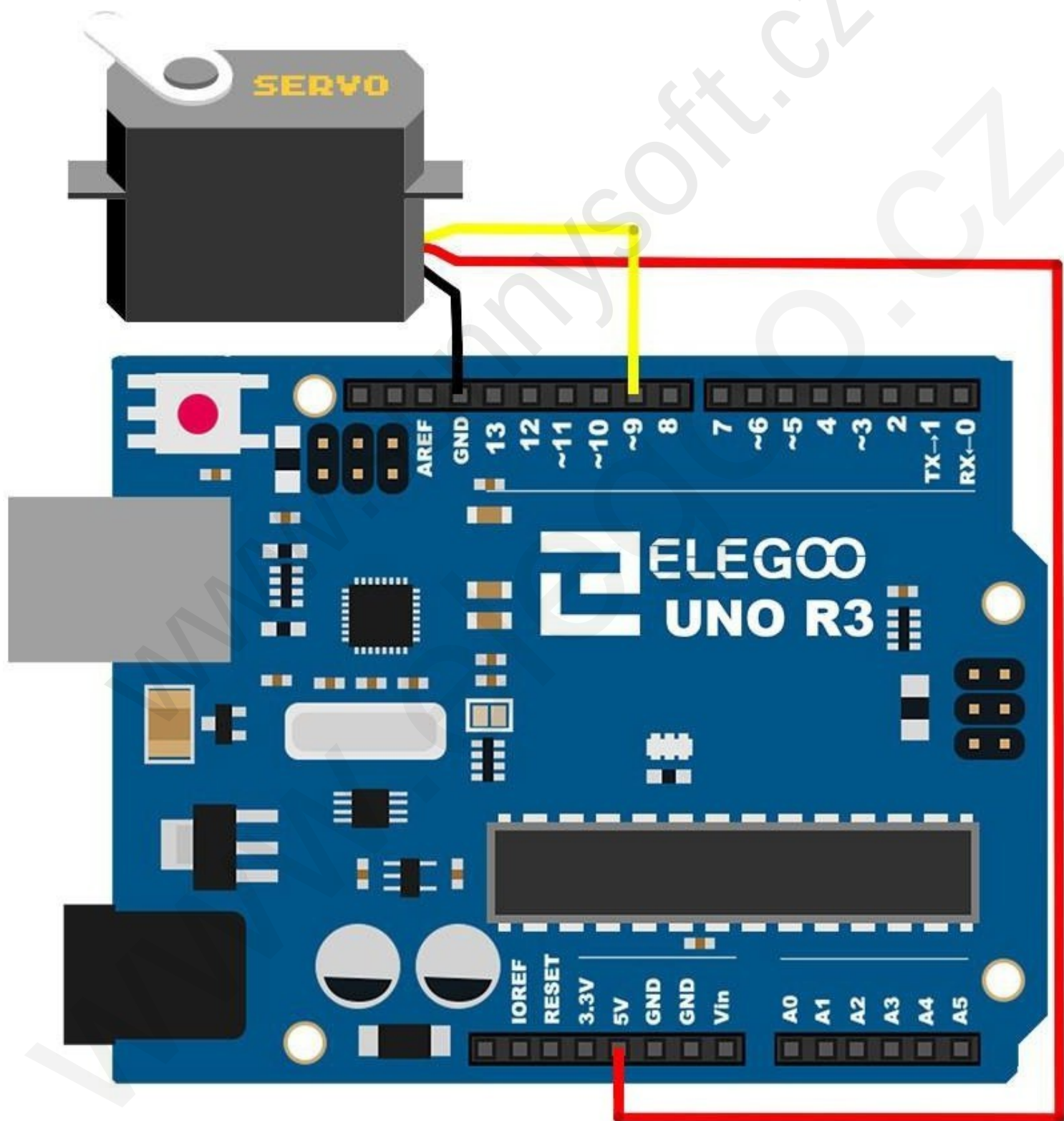
Conectarea

Diagrama



73/162

Diagrama de cablare



74/162

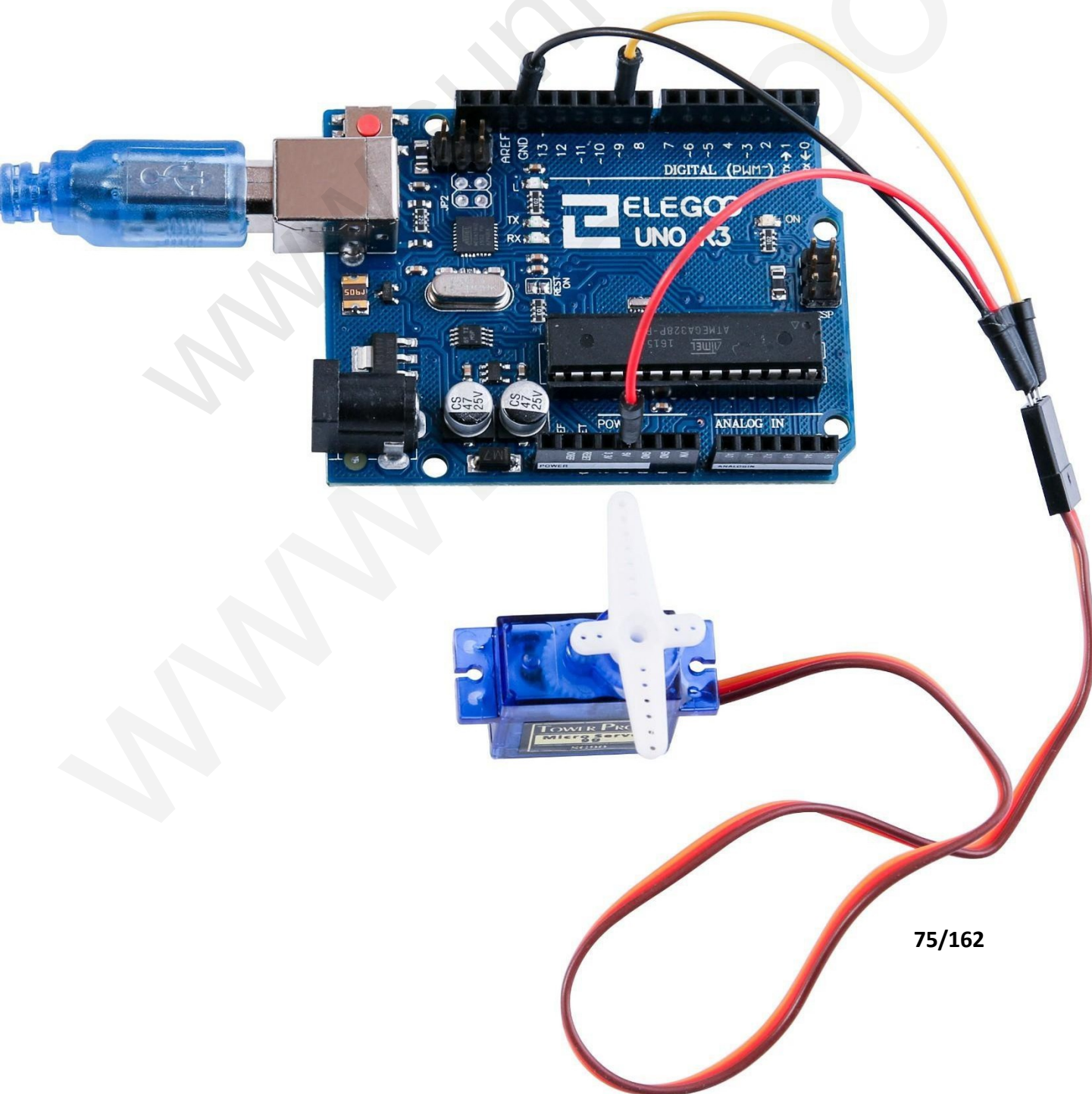
Cod

După conectare, deschideți programul în folderul de cod - Lecția 9 Servo și faceți clic pe UPLOAD pentru a încărca programul. Consultați Lecția 2 pentru detalii privind încărcarea programului dacă apar erori.

Înainte de a începe, asigurați-vă că aveți instalată biblioteca <Servo>, sau reinstalați-l. În caz contrar, codul dvs. nu va funcționa. Consultați Lecția 1 pentru detalii privind încărcarea fișierului de bibliotecă.

Exemplu de imagine

În imagine, firul servo maro este conectat prin firele negre M-M, cel roșu este conectat prin firele M-M roșii, iar cel portocaliu este conectat prin firele M-M galbene.



75/162

Lecția 10 Modul senzor cu ultrasunete

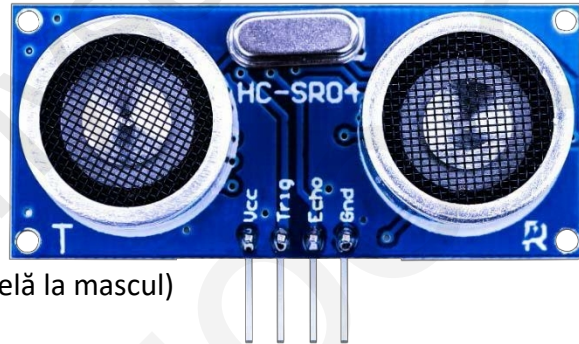
Prezentare generală

Senzorul cu ultrasunete este perfect pentru toate tipurile de proiecte care necesită măsurarea distanței și evitarea obstacolelor.

Senzorul HC-SR04 este ieftin și ușor de utilizat, deoarece vom folosi biblioteca special concepută pentru acest senzor.

Componentă necesară:

- (1) x Elegoo Uno R3
- (1) x modul senzor cu ultrasunete
- (4) x Fire F-M (fire DuPontwires de la femelă la mascul)



Introducere în dosar

Senzor cu ultrasunete

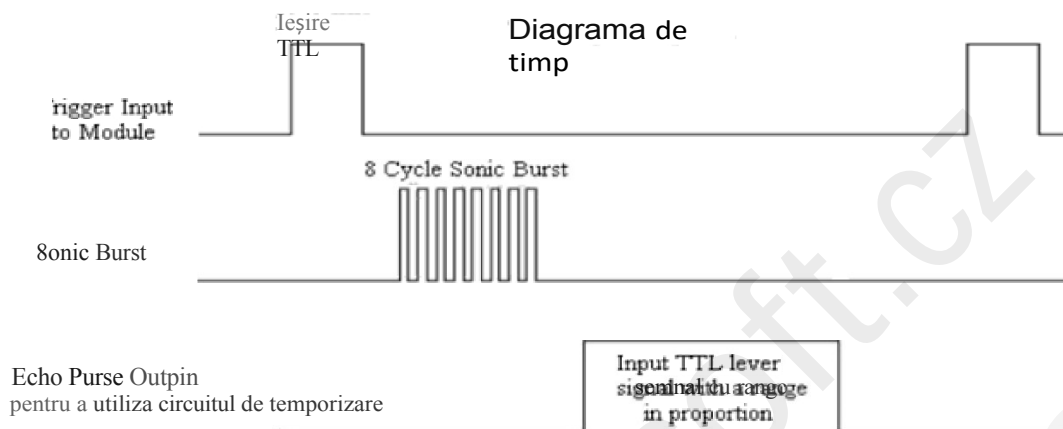
Modulul senzor cu ultrasunete HC-SR04 oferă o funcție de măsurare fără contact de 2- 400 cm, precizia de măsurare poate ajunge până la 3 mm. Modulele conțin transmițător ultrasonic, receptor și circuit de control. Principiul de funcționare de bază:

- (1) Utilizarea declanșatorului IO pentru semnal de nivel înalt de cel puțin 10us,
- (2) Modulul trimite automat opt semnale de 40 kHz și detectează dacă există un semnal de întoarcere a impulsului.
- (3) Semnalul IF înapoi, prin nivelul ridicat, durata de ieșire ridicată IO este timpul de la trimiterea rotației torului cu ultrasunete.

Distanța de testare = (timpul la nivel înalt × viteza sunetului (340 m/s) / 2

Mai jos este prezentată o cronologie. Pentru a începe măsurarea, trebuie doar să aplicați un impuls scurt de 10us la intrarea de declanșare și apoi modulul va trimite un ultrasunete de 8 cicluri la 40 kHz și va crește ecoul. Ecoul este un obiect de distanță care este în raportul dintre lățimea impulsului și intervalul . puteți calcula intervalul prin intervalul de timp dintre trimiterea semnalului de declanșare și primirea semnalului de ecou. Formula: us

/ 58 = centimetri sau noi / 148 = inci; sau: interval = timp nivel înalt * viteză (340M/S) / 2; recomandăm utilizarea unui ciclu de măsurare mai mare de 60 ms pentru a evita declanșarea semnalului electronic.



Conectarea

Diagrama

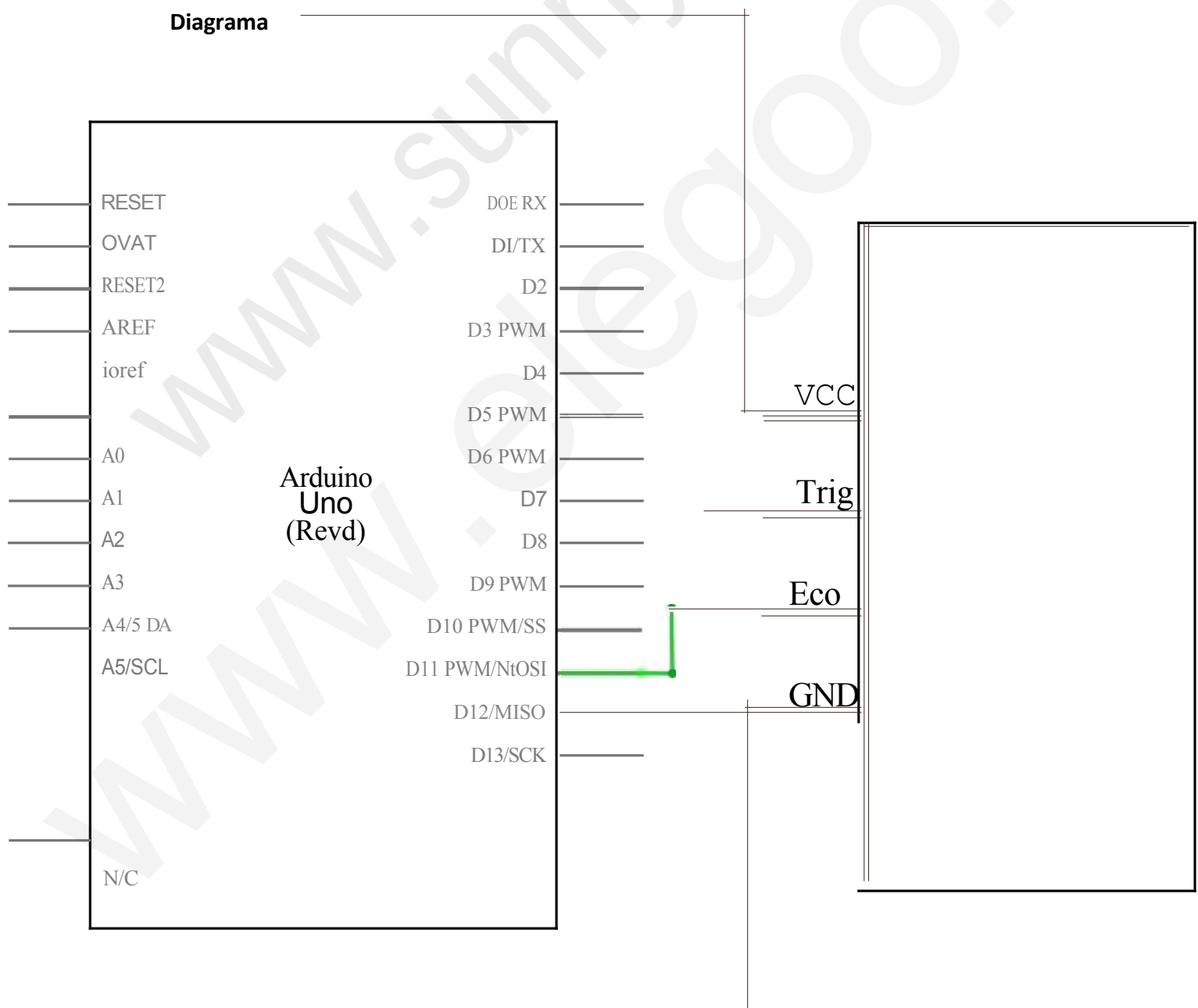
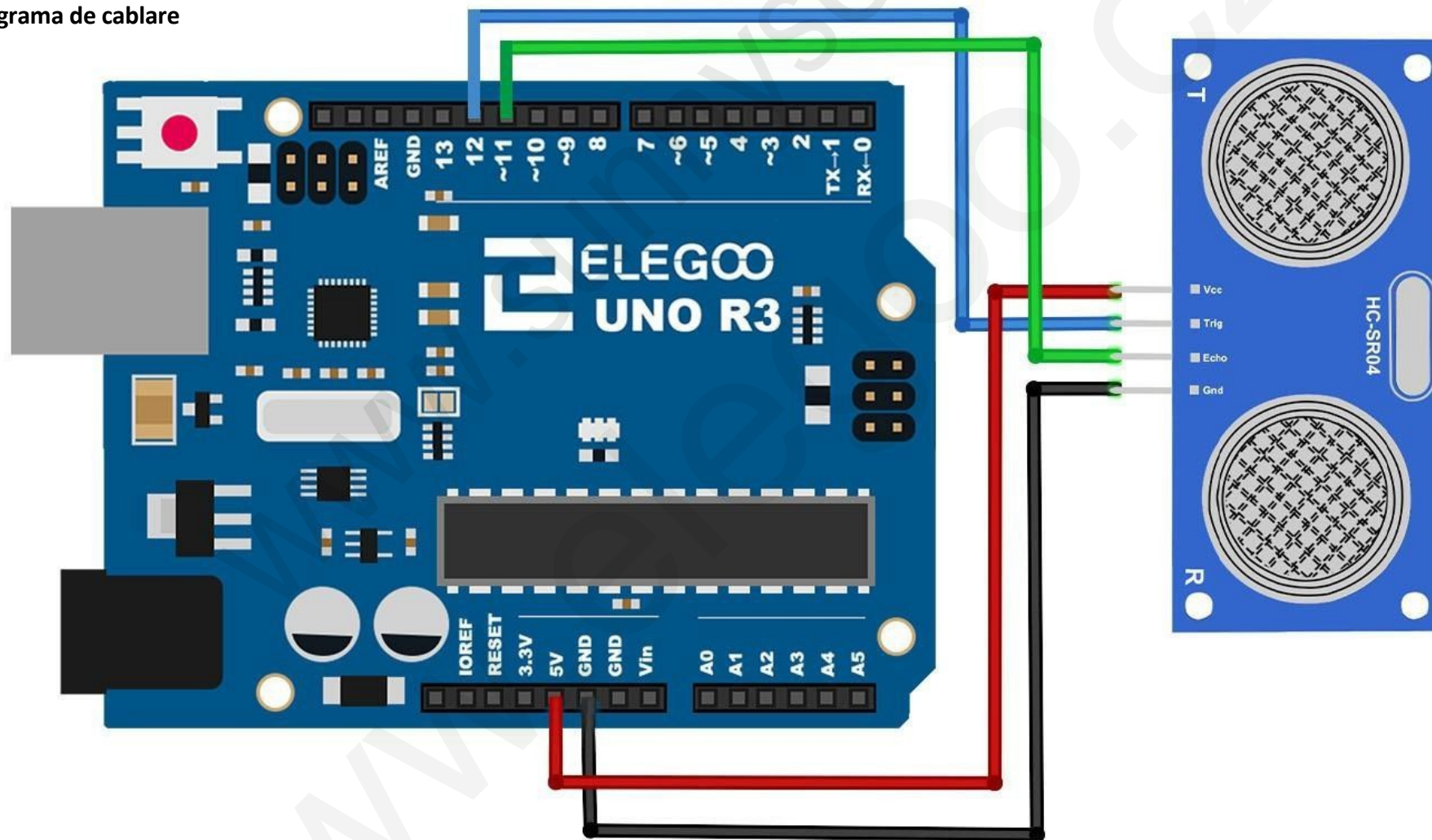


Diagrama de cablare



Cod

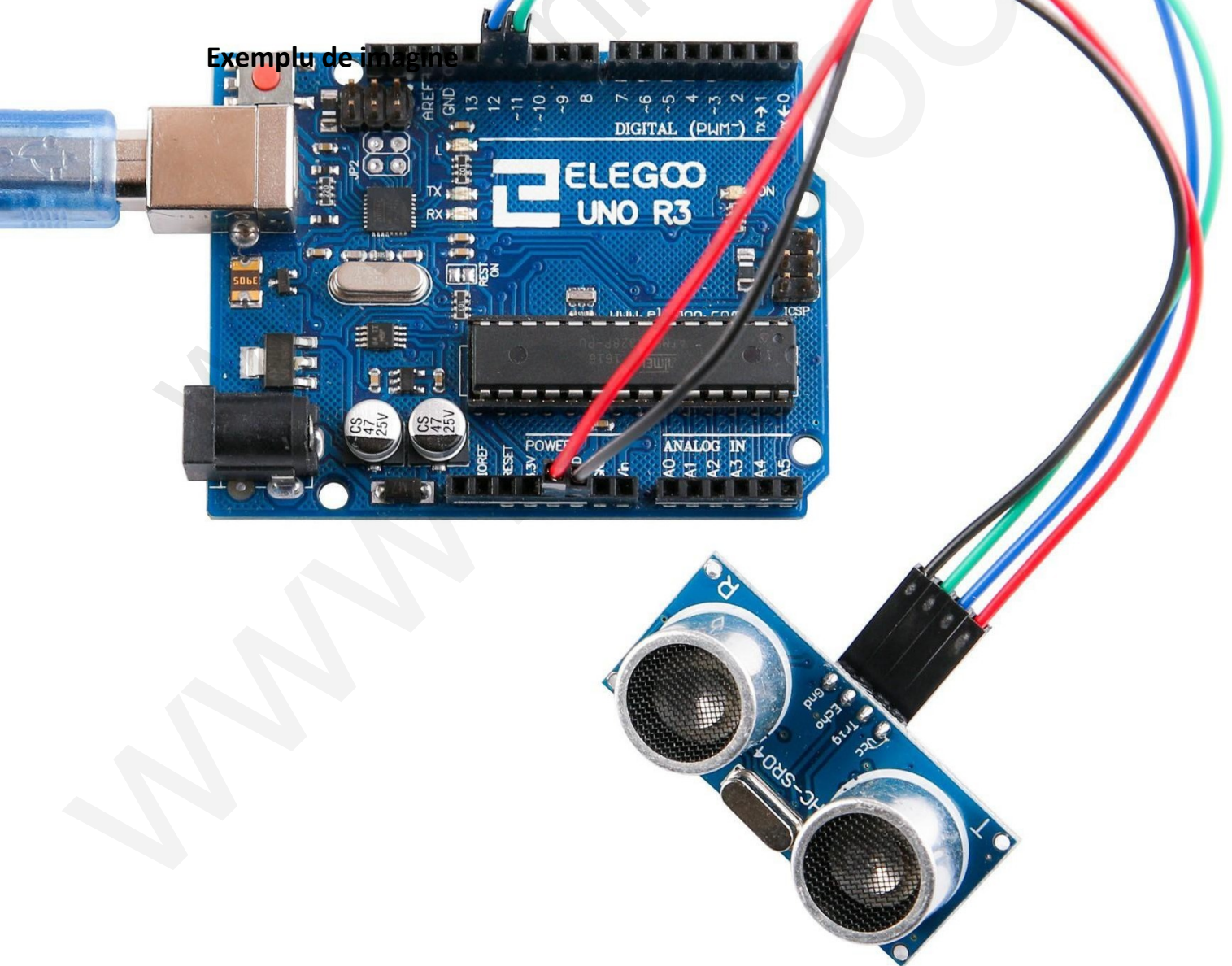
Utilizarea unei biblioteci concepute pentru acești senzori ne permite să scurtăm și să simplificăm codul. Noi includem biblioteca la începutul codului nostru și apoi folosim comenzi simple pentru a controla comportamentul senzorului.

După conectare, deschideți programul din folderul de cod - [Lecția 10 Modul senzor ultrasonic](#) și faceți clic pe butonul **UPLOAD** pentru a încărca programul. Consultați [Lecția 2](#) pentru detalii privind încărcarea programului dacă există erori.

Înainte de a începe, asigurați-vă că aveți instalată biblioteca <HC-SR04>, sau reinstalați-l. În caz contrar, codul dvs. nu va funcționa.

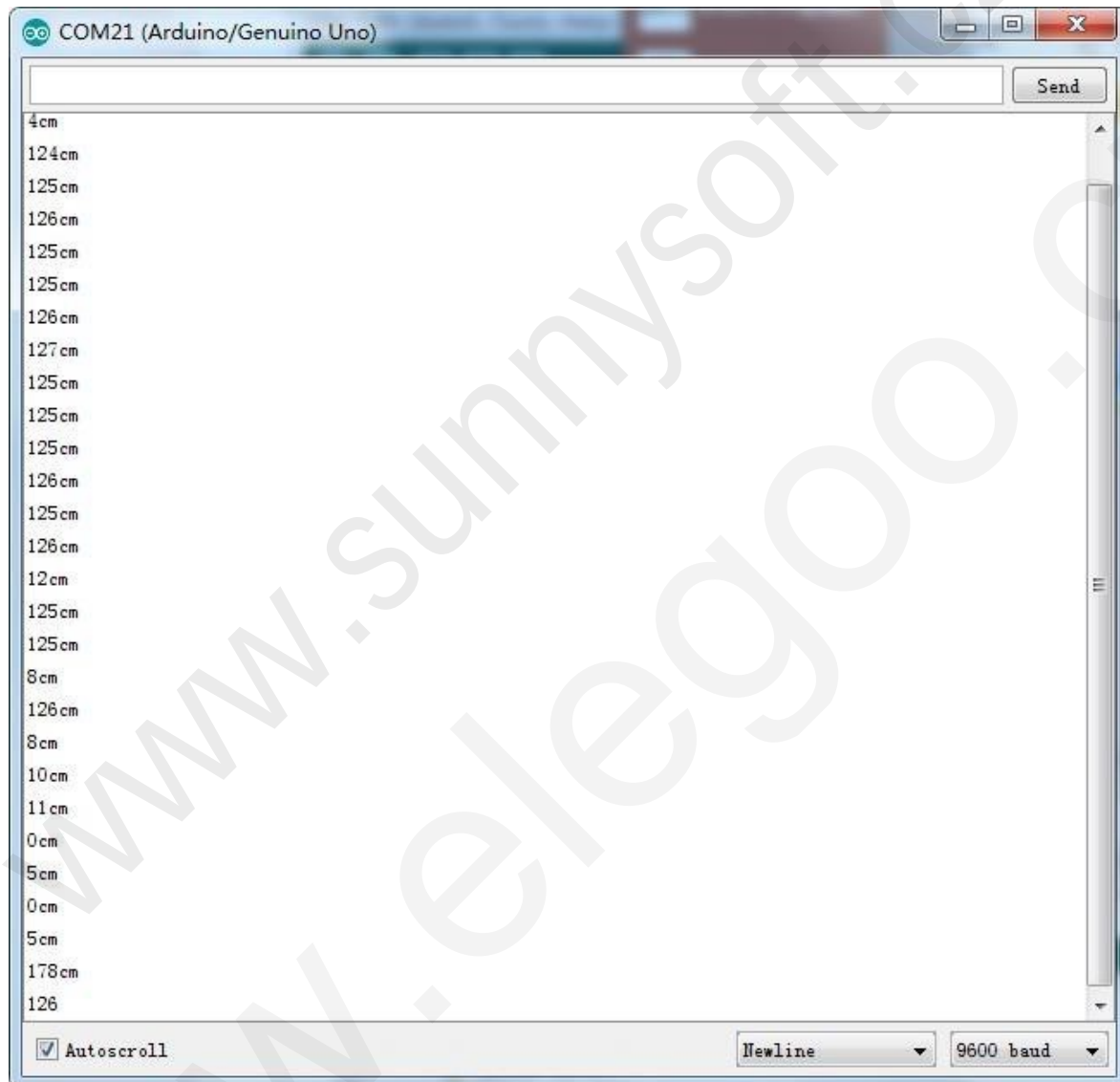
Consultați [Lecția 1](#) pentru detalii privind încărcarea fișierului de bibliotecă.

Exemplu de imagine



Deschideți monitorul și apoi puteți vedea datele pe măsură ce suflă:

Faceți clic pe butonul Monitor serial pentru a porni monitorul serial. Informațiile de bază despre monitorul serial sunt prezentate în detaliu în Lecția 1.



Lecția 11 Senzor de temperatură și umiditate DHT11

Prezentare generală

În acest manual vom învăța cum să utilizăm senzorul de temperatură și umiditate DHT11. Acesta este suficient de precis pentru majoritatea proiectelor care trebuie să monitorizeze umiditatea și temperatura.

Din nou, vom utiliza o bibliotecă special concepută pentru acești senzori, ceea ce va face codul nostru scurt și ușor de scris.

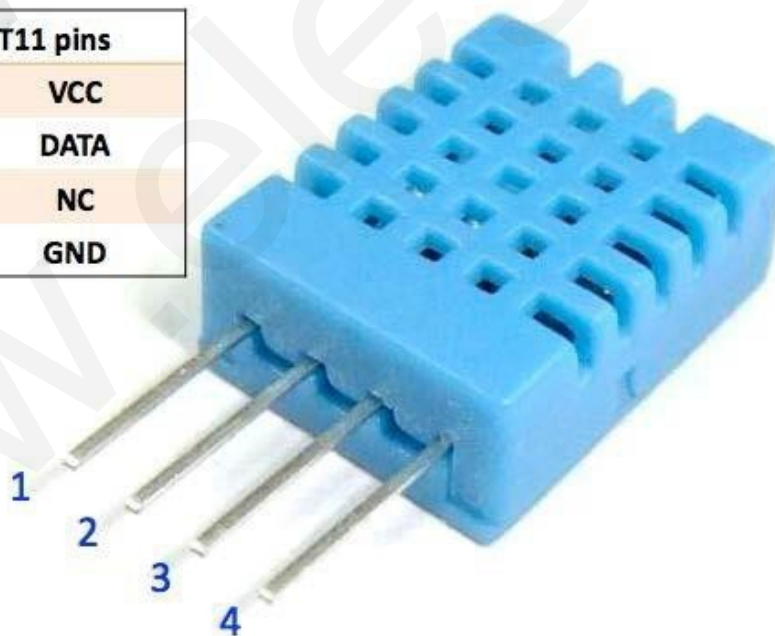
Componentă necesară:

- (1) x Elegoo Uno R3
- (1) x Modul de temperatură și umiditate DHT11
- (3) x fire F-M (fire DuPont de la femelă la mascul)

Introducere în dosar

Senzor de temperatură și umiditate:

DHT11 pins	
1	VCC
2	DATA
3	NC
4	GND



Senzorul digital de temperatură și umiditate DHT11 este un senzor compozit care conține un semnal de ieșire digital calibrat de temperatură și umiditate. Sunt utilizate tehnologii speciale de achiziție a modulelor digitale și de detectare a temperaturii și umidității pentru a asigura fiabilitate ridicată și

stabilitate excelentă pe termen lung. Senzorul încorporează un senzor rezistiv pentru componente umede și un dispozitiv NTC de măsurare a temperaturii și se conectează la un microcontroler puternic pe 8 biți.

Aplicații: tratarea aerului, dehumidificatoare, echipamente de testare și control, bunuri de consum, automobile, control automat, înregistratoare de date, stații meteorologice, aparate de uz casnic, controlere de umiditate, dispozitive medicale și alte dispozitive de măsurare și control al umidității.

Parametrii produsului

Umiditate relativă:

Rezoluție: Precizie: 16bit

Repetabilitate: $\pm 1\%$ RH

Precizie:

La $25\text{ }^{\circ}\text{C} \pm 5\%$ umiditate

relativă

Înlocuibil: complet înlocuibil Timp de răspuns: 1

/e(63%) de $25\text{ }^{\circ}\text{C}$ 6s

1m/s aer 6s Histerezis: $<\pm$

0,3% RH

Stabilitate pe termen lung: $<\pm 0,5\%$ RH/an la temperatură:

Rezoluție: Gamă: 16bit

Repetabilitate: $\pm 0,2\text{ }^{\circ}\text{C}$

Gamă: la $25\text{ }^{\circ}\text{C} \pm 2\text{ }^{\circ}\text{C}$

Timp de răspuns: 1/e (63%)

10S Caracteristici electrice

Alimentare: DC 3.5~5.5V

Curent de alimentare: 0,3 mA măsurare în așteptare 60 μA

Timp de eșantionare: mai mult de 2

secunde Descriere pin:

1. sursă de alimentare VDD 3,5~5,5V DC
2. DATE date seriale, un autobuz
3. NC, pin gol

4. GND masă, sursă de alimentare negativă

Conectarea

Diagrama

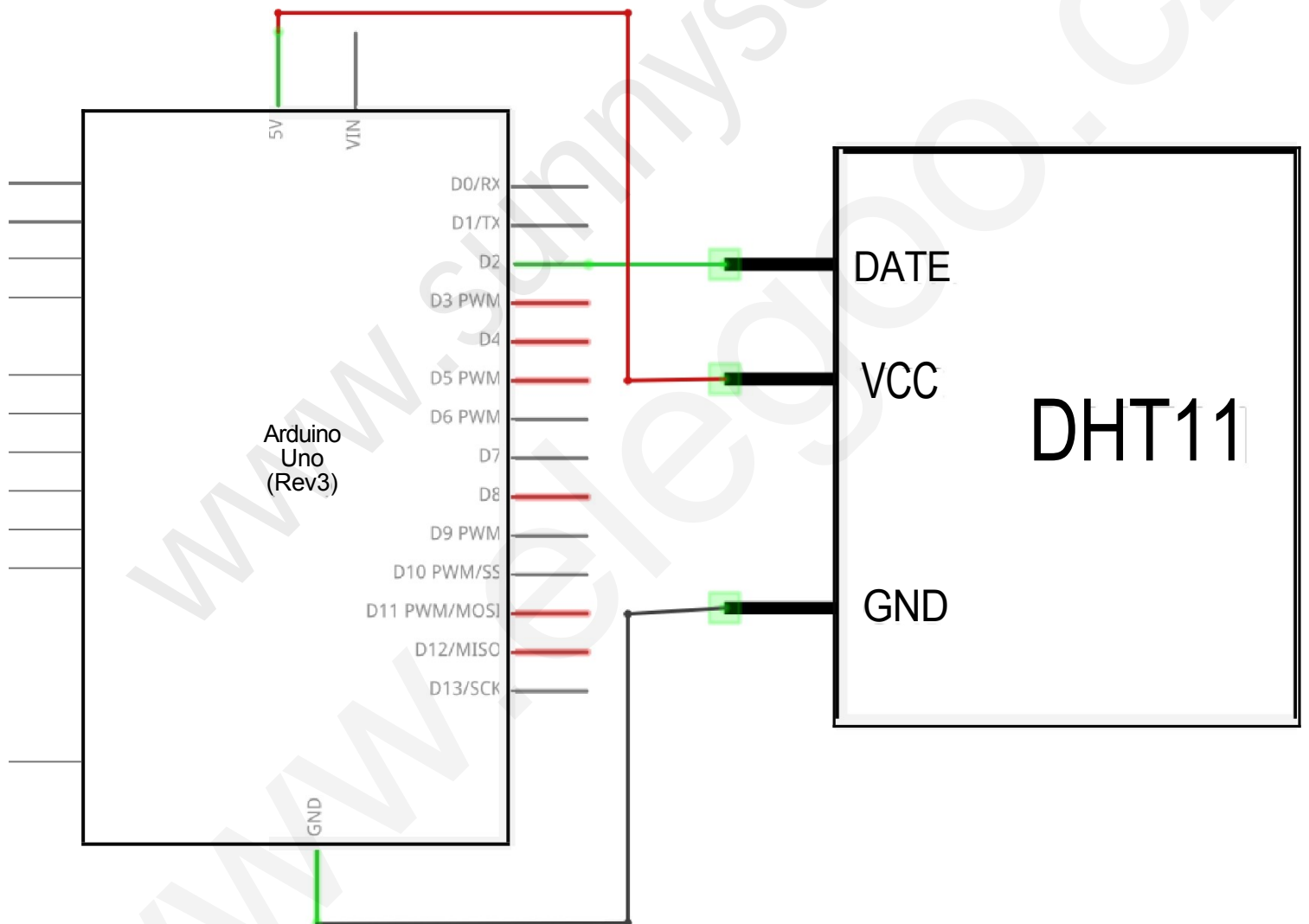
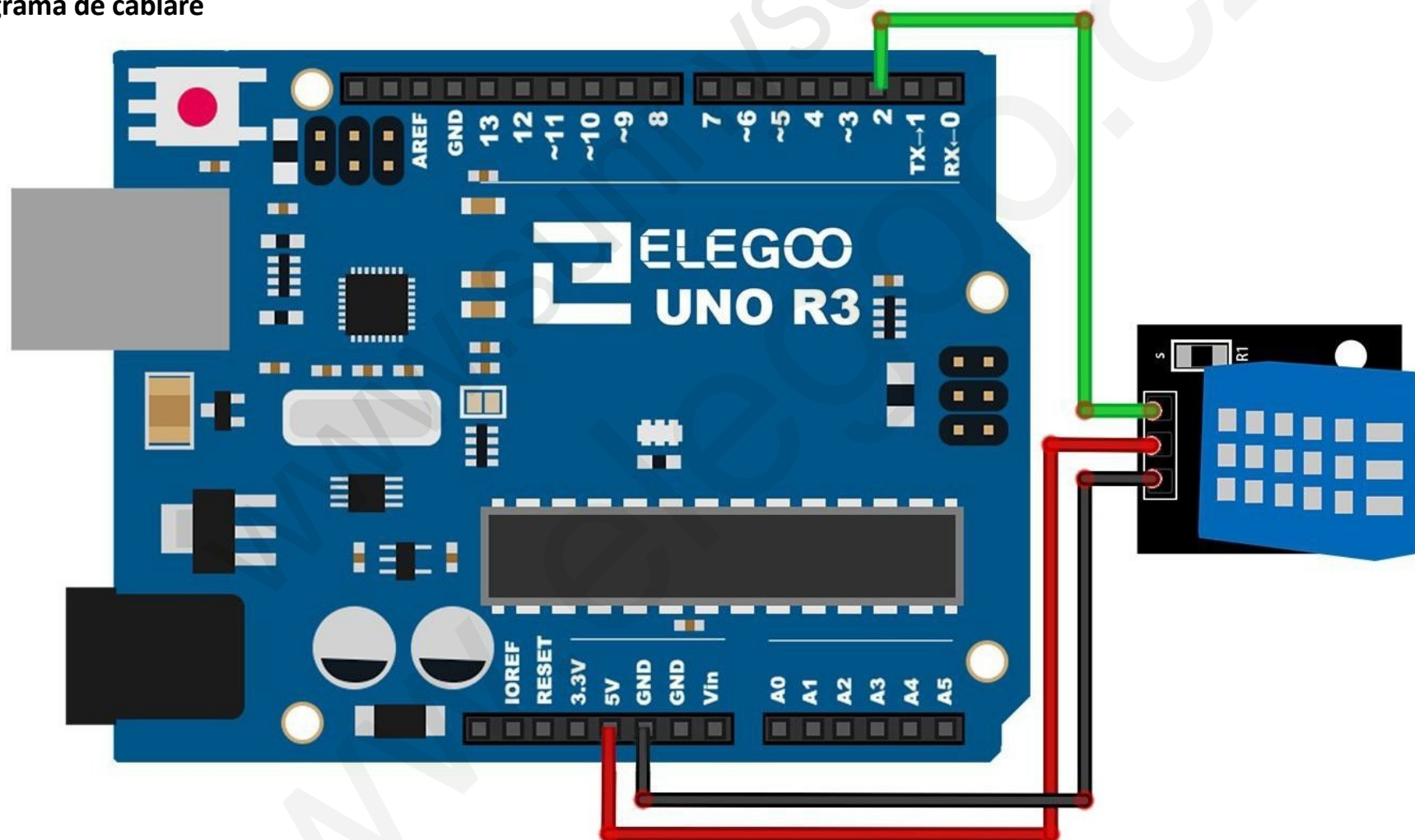


Diagrama de cablare



După cum puteți vedea, avem nevoie doar de 3 conexiuni la senzor, deoarece unul dintre pini nu este utilizat.

Acestea sunt următoarele conexiuni: tensiune, masă și semnal, care pot fi conectate la orice pin de pe UNO.

Cod

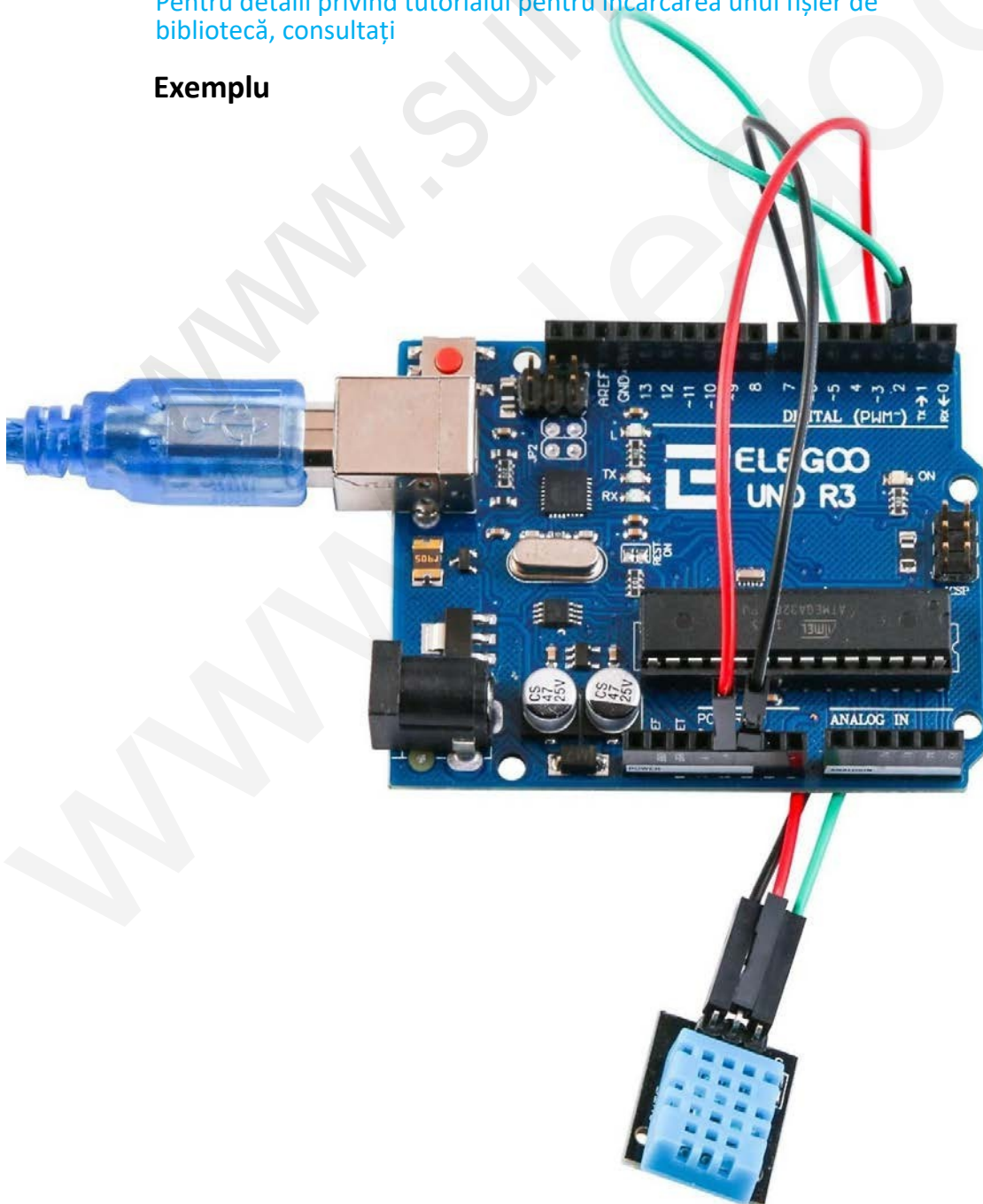
Odată conectat, deschideți programul din folderul cu codul - Lesson 11 Temperature and Humidity Sensor DHT11 și faceți clic pe butonul UPLOAD pentru a încărca programul.

Consultați Lecția 2 pentru detalii privind înregistrarea programului în caz de erori.

Înainte de execuție, asigurați-vă că aveți biblioteca <DHT> instalată sau reinstalați-o.

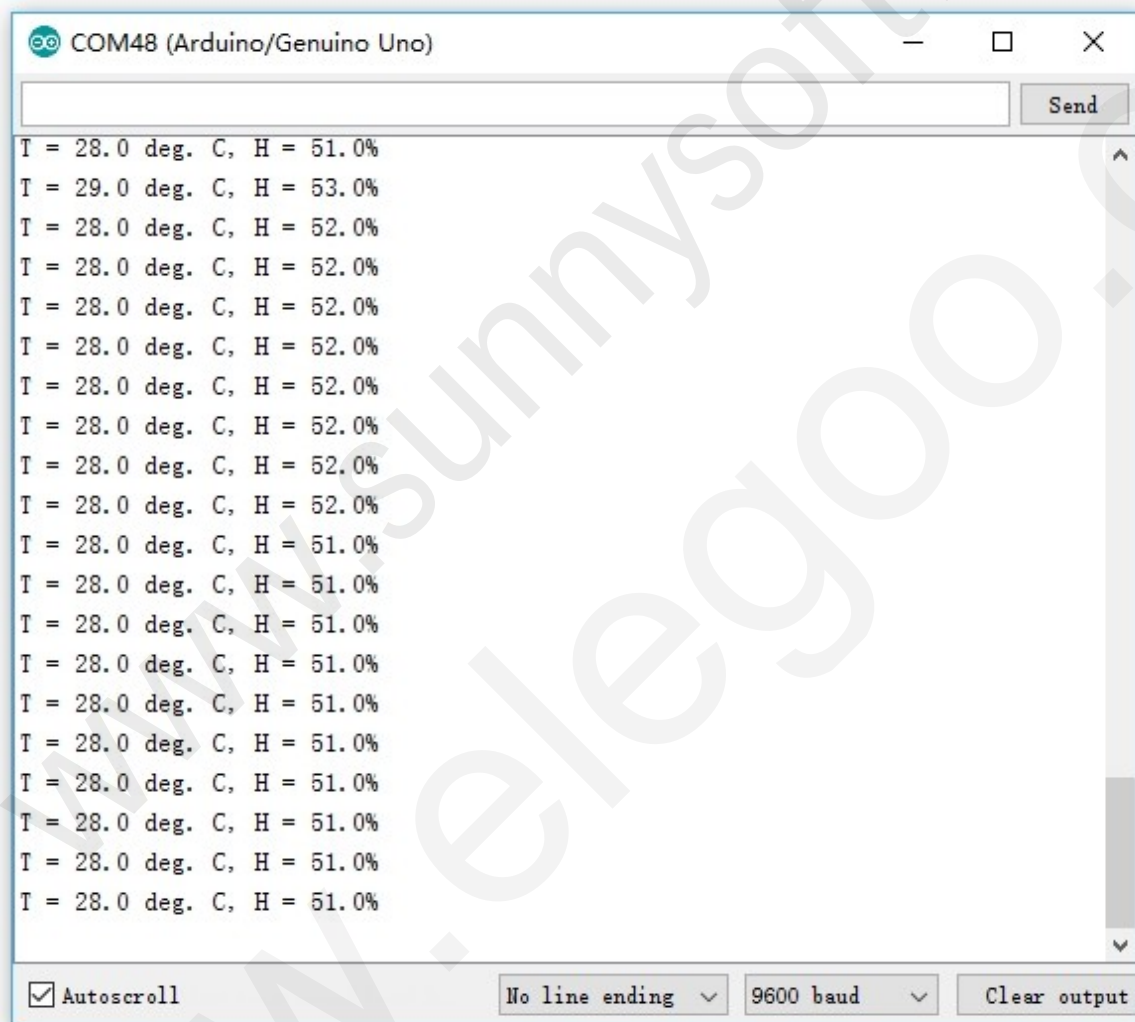
Pentru detalii privind tutorialul pentru încărcarea unui fișier de bibliotecă, consultați

Exemplu



Încărcați programul și apoi deschideți monitorul pentru a vedea datele de mai jos:
(Acesta arată temperatura ambientală, putem vedea că este de 28 de grade).

Faceți clic pe butonul [Monitor serial](#) pentru a porni monitorul serial. Informațiile de bază despre monitorul serial sunt prezentate în detaliu în [Lecția 1](#).



Lecția 12 Modul Joystick analogic

Prezentare generală

Joystick-urile analogice sunt o modalitate excelentă de a adăuga control proiectelor dvs. În acest tutorial vom învăța cum să utilizăm modulul joystick analogic.

Componentă necesară:

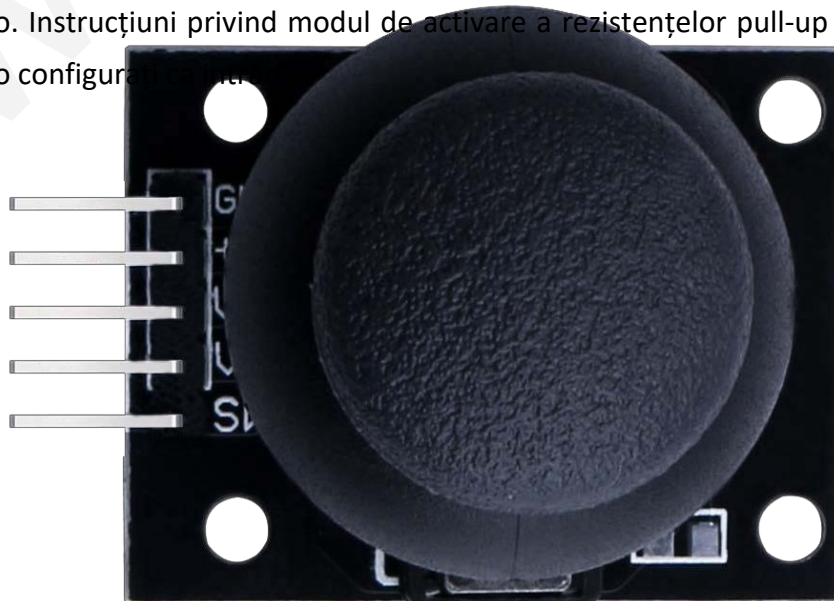
- (1) x Elegoo Uno R3
- (1) x modul joystick
- (5) x fire F-M (fire DuPontwires de la femelă la mascul)

Introducere în dosar

Joystick

Modulul are 5 pini: VCC, masă, X, Y, cheie. Rețineți că marcajele de pe modulul dvs. pot varia ușor în funcție de locul de unde ați obținut modulul. Pârghia este analogică și ar trebui să ofere citiri mai precise decât simpla tatură "direcțională" a joystick-ului folosind unele forme de butoane sau comutatoare mecanice. În plus, puteți apăsa joystick-ul în jos (destul de tare pe al meu) pentru a activa butonul "apăsați pentru a selecta".

Pentru a citi datele de la pinii X/Y trebuie să folosim pinii analogici ai Arduino, iar pentru a citi butonul trebuie să folosim pinul digital. Pinul Key este conectat la masă atunci când joystick-ul este apăsat, iar în caz contrar este flotant. Pentru a obține date stabile de la pinul Key/Select, trebuie să îl conectăm la VCC folosind o rezistență pull-up. Pot fi utilizate rezistențele încorporate pe pinii digitali ai Arduino. Instrucțiuni privind modul de activare a rezistențelor pull-up pentru pinii Arduino configurați ca intrări digitale.



Conectarea

Diagrama

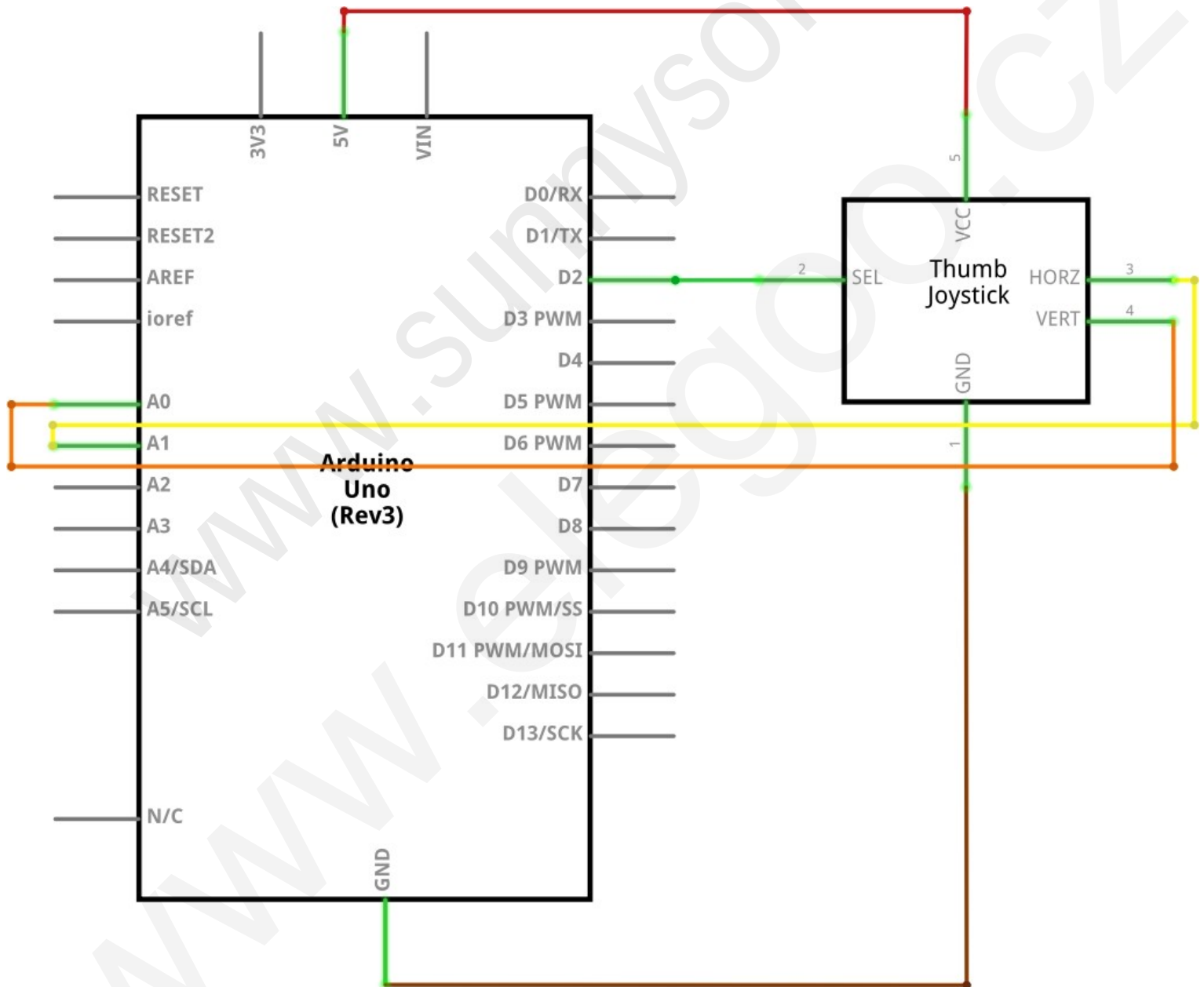
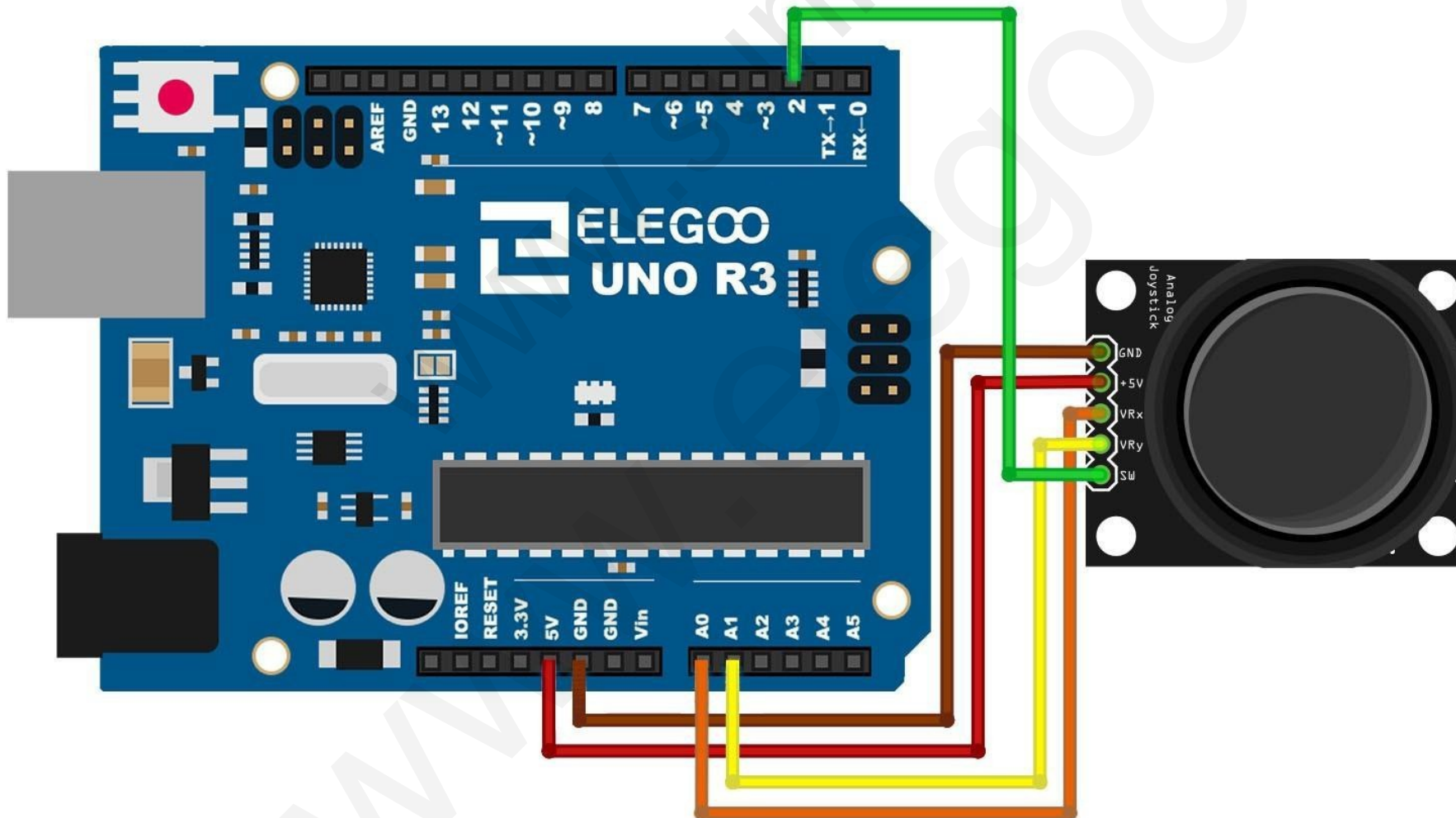


Diagrama de cablare



Avem nevoie de 5 conexiuni la joystick.

Aceste conexiuni sunt cheie, Y, X, tensiune și masă.

"Y și X" sunt analogice, iar "Key" este digitală. Dacă nu aveți nevoie de un comutator, puteți utilizați numai 4 pini.

Cod

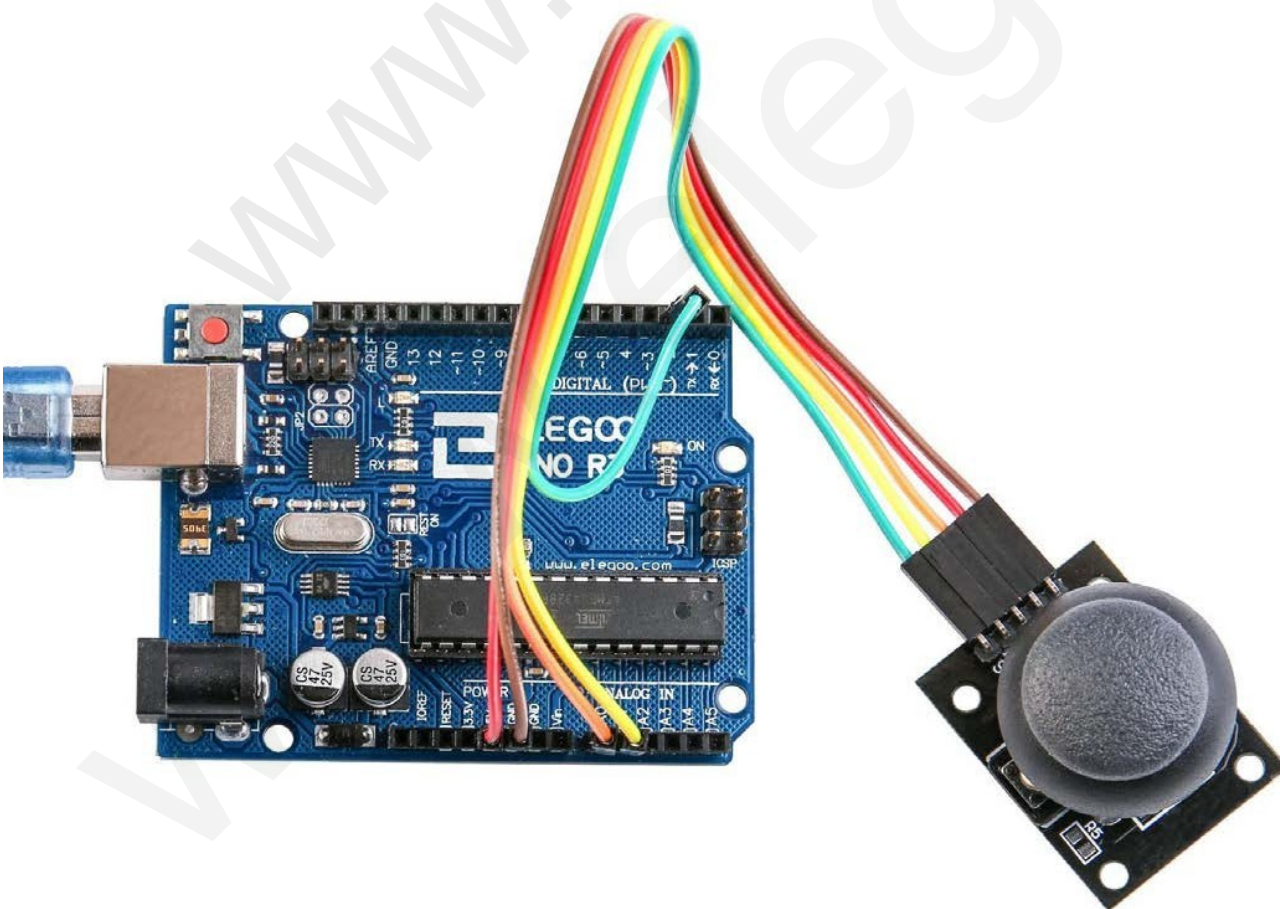
După conectare, deschideți programul din folderul de cod - [Lecția 12 Modul analogic Joystick](#) și faceți clic pe butonul **UPLOAD** pentru a încărca programul. Consultați [Lecția 2](#) pentru detalii privind încărcarea programului dacă apar erori.

Joystick-urile analogice sunt practic potențiometre, deci returnează valori analogice.

Atunci când joystick-ul este în repaus sau la mijloc, acesta ar trebui să returneze o valoare de aproximativ 512.

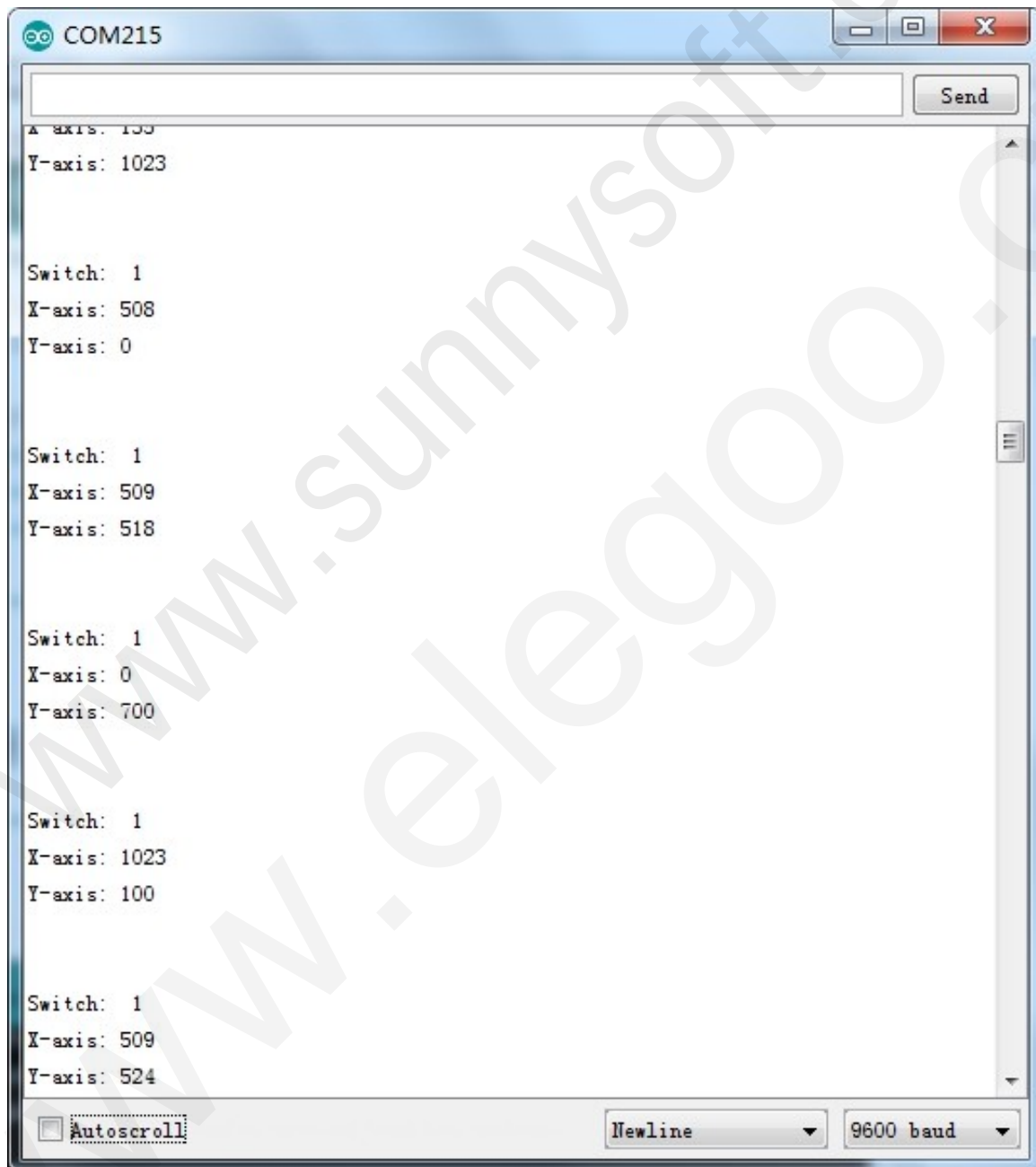
Intervalul de valori este de la 0 la 1024.

Exemplu de imagine



Deschideți monitorul și apoi puteți vedea datele pe măsură ce suflă:

Faceți clic pe butonul Monitor serial pentru a porni monitorul serial. Informațiile de bază despre monitorul serial sunt prezentate în detaliu în Lecția 1.



Lecția 13 Modul receptor IR

Prezentare generală

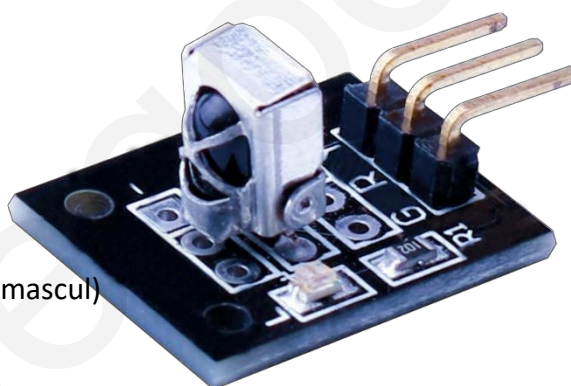
Utilizarea unei telecomenzi cu infraroșu este o modalitate excelentă de a vă controla proiectul fără fir.

Telecomanda cu infraroșu este simplă și ușor de utilizat. În acest tutorial, vom conecta un receptor cu infraroșu la UNO și apoi vom utiliza o bibliotecă care a fost proiectată pentru acel senzor special.

În schița noastră, vom avea toate codurile IR hexazecimale disponibile pe această telecomandă și vom vedea, de asemenea, dacă codul a fost recunoscut și dacă ținem tasta apăsată.

Componentă necesară:

- (1) x Elegoo Uno R3
- (1) x modul receptor IR
- (1) x telecomandă IR
- (3) x fire F-M (fire DuPont de la femelă la mascul)



Introducere în dosar

RECEPTOR SENZOR IR:

Detectoarele în infraroșu sunt microcipuri mici cu o celulă fotoelectrică care sunt adaptate la lumina infraroșie. Acestea sunt utilizate aproape întotdeauna pentru detectarea telecomenzii - fiecare televizor și DVD player are unul dintre acestea în partea frontală pentru a asculta semnalul infraroșu de la clicker. În interiorul telecomenzii se află un LED IR corespunzător care trimite impulsuri IR pentru a informa televizorul când acesta este pornit, oprit sau când schimbă canalul. Lumina IR nu este vizibilă pentru ochiul uman, ceea ce înseamnă că testarea setărilor necesită un pic mai multă muncă.

Există câteva diferențe între aceste fotocupluri și, de exemplu, fotocuplurile CdS:

Detectoarele IR sunt filtrate special pentru lumina IR, nu sunt potrivite pentru detectarea luminii vizibile. Pe de altă parte, fotocelulele sunt bune pentru detectarea luminii vizibile galbene/verzi și nu sunt bune pentru lumina IR.

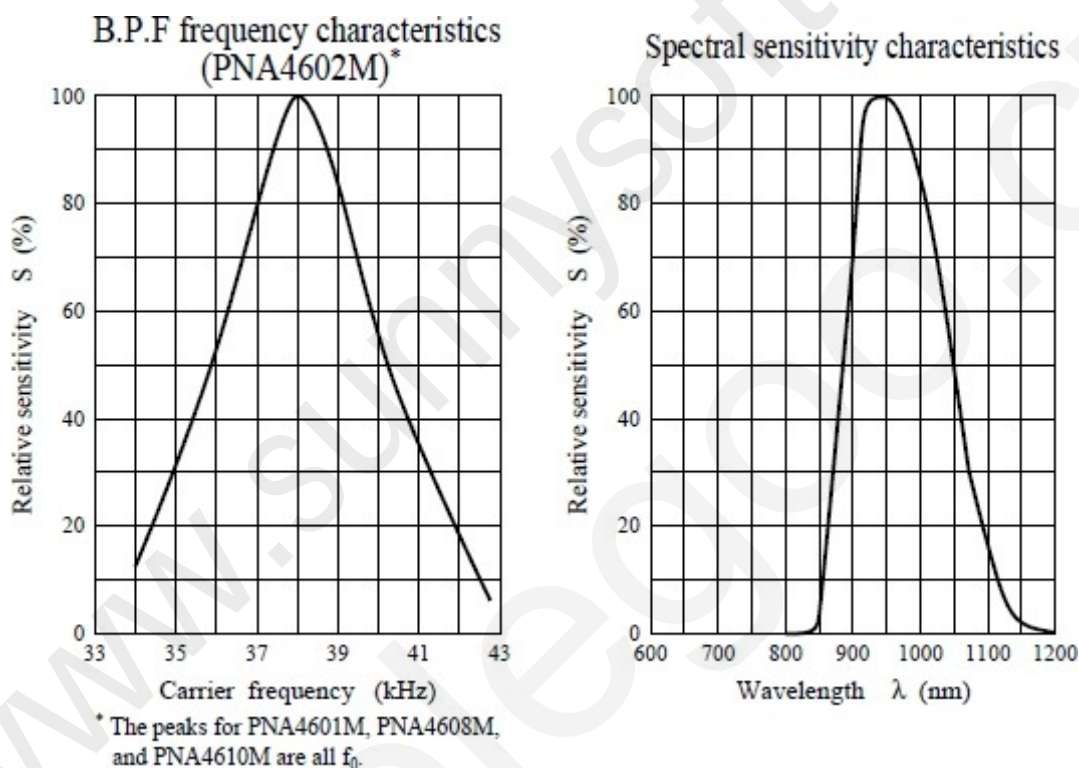
Detectoarele infraroșii au în interior un demodulator care caută radiații infraroșii

modulate cu o frecvență de 38 KHz. Prin simpla iluminare a LED-ului IR

www.sunnysoft.cz

nu va fi detectată, trebuie să fie o pâlpâire PWM la 38 KHz. Fotocelulele nu au demodulator și pot detecta orice frecvență (inclusiv DC) în limita vitezei de răspuns a fotocelulei (care este de aproximativ 1 KHz).

Detectoarele IR au o ieșire digitală - fie detectează un semnal IR de 38KHz și au o ieșire scăzută (0V), fie nu detectează niciun semnal și au o ieșire ridicată (5V). Fotocelulele se comportă ca niște rezistențe, rezistența modificându-se în funcție de cantitatea de lumină la care sunt expuse. **Ce puteți măsura**



După cum puteți vedea din aceste grafice din fișa tehnică, frecvența maximă de detecție este de 38 KHz, iar culoarea maximă a LED-ului este de 940 nm. Puteți utiliza o frecvență de la aproximativ 35 KHz la 41 KHz, dar sensibilitatea va scădea, astfel încât nu va detecta la fel de bine la distanță. În mod similar, puteți utiliza LED-uri cu o lungime de undă între 850 și 1100 nm, dar acestea nu vor funcționa la fel de bine ca LED-urile cu o lungime de undă între 900 și 1000 nm, așa că asigurați-vă că obțineți LED-uri potrivite! Consultați fișa tehnică a LED-ului infraroșu pentru lungimea de undă.

Încercați să obțineți 940nm - rețineți că 940nm nu este lumină vizibilă!

**Conectarea
Diagrama**

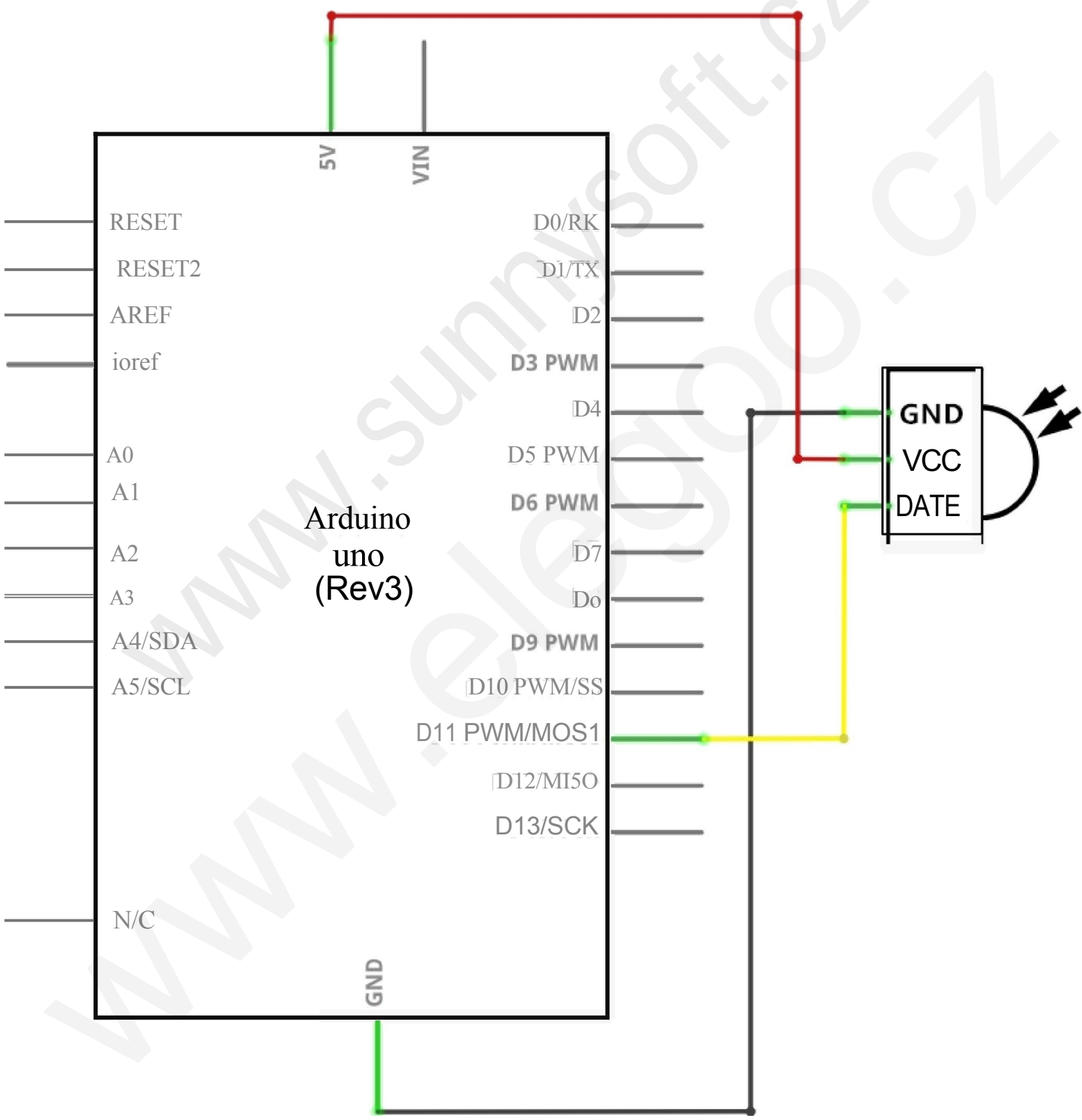
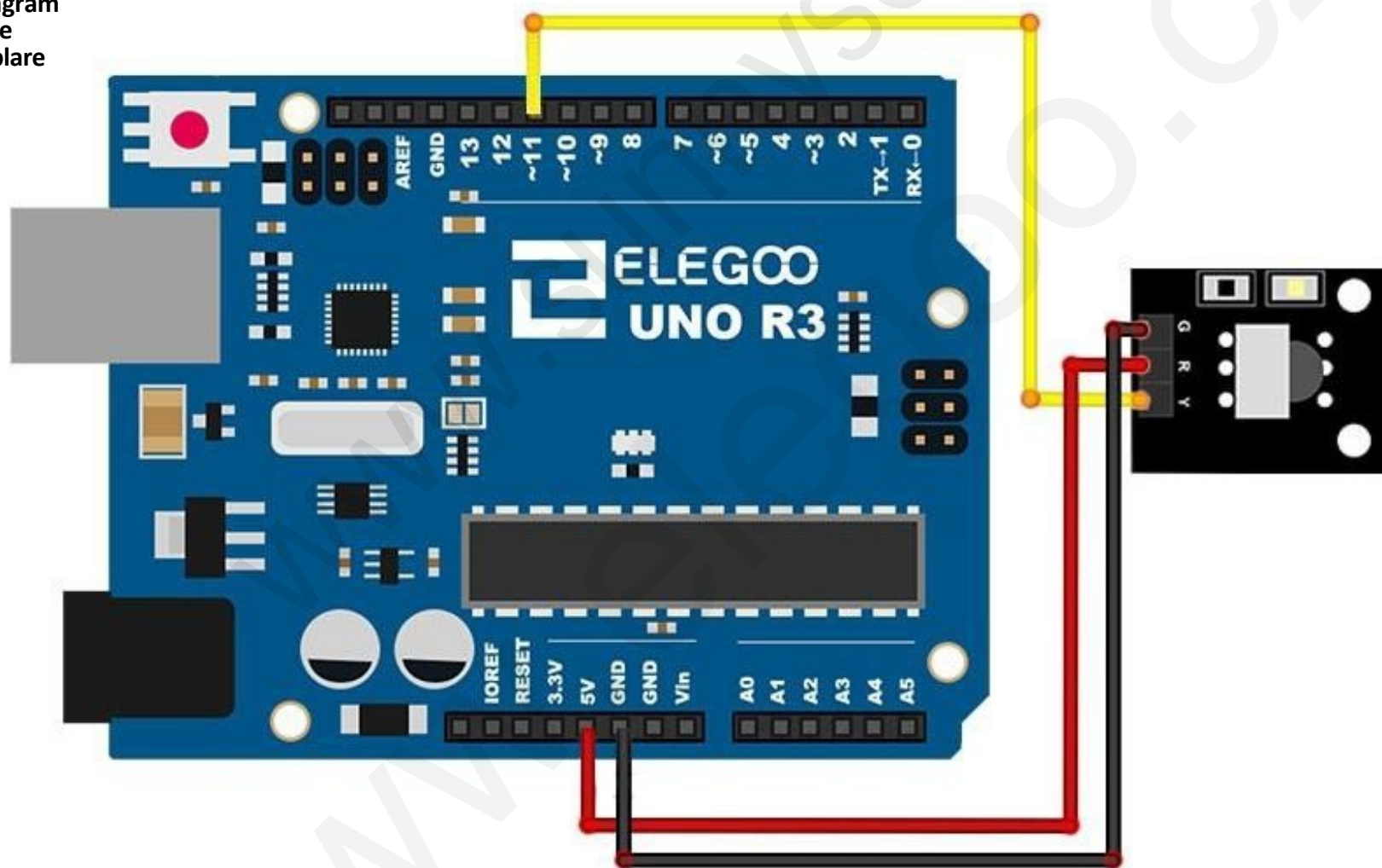


Diagram
a de
cablare



Există 3 conexiuni la receptorul IR.

Aceste conexiuni sunt semnal, tensiune și masă.

"-" este masa, "S" este semnalul, iar pinul din mijloc este tensiunea de 5 V.

Cod

Odată conectat, deschideți programul din folderul de cod - Lecția 13 Modul receptor IR și faceți clic pe butonul UPLOAD pentru a încărca programul. Consultați Lecția 2 pentru detalii privind încărcarea programului dacă apar erori.

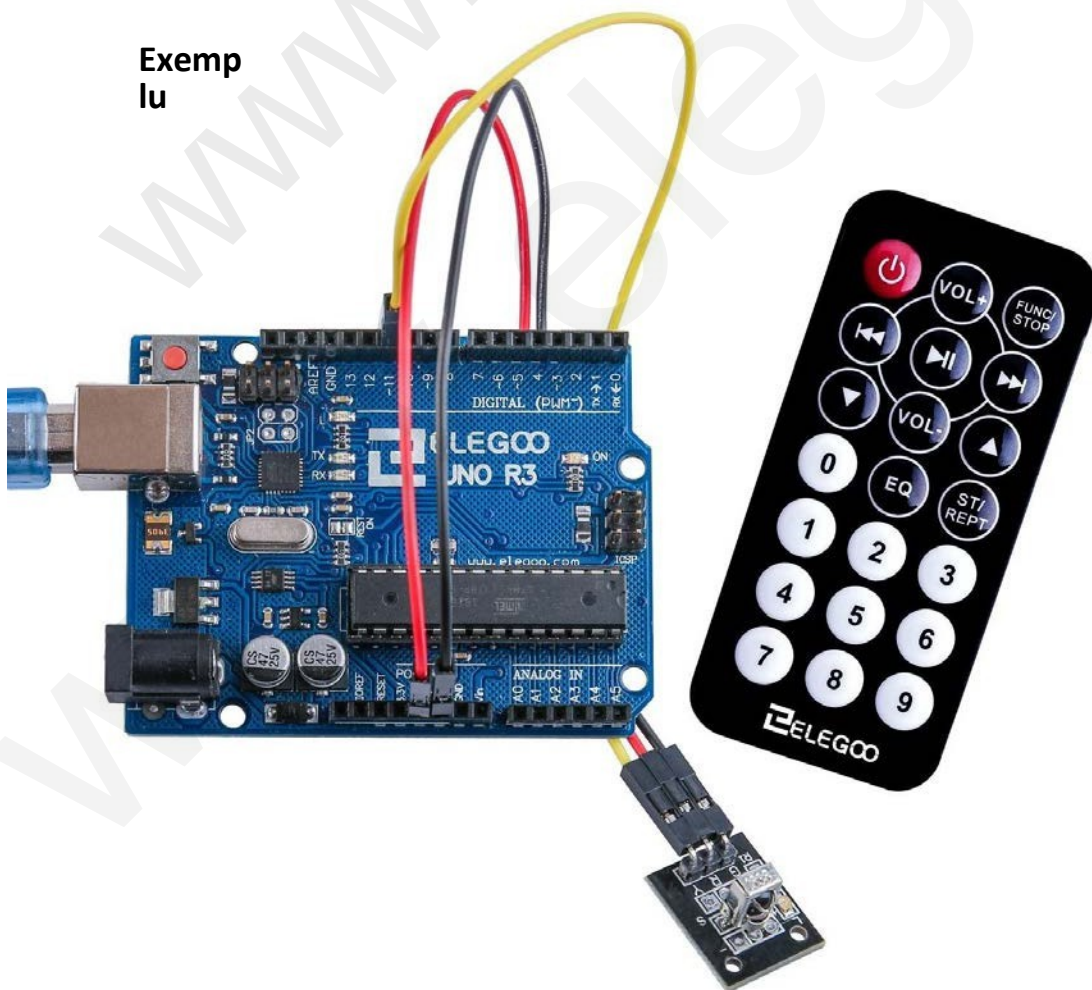
Înainte de a rula această caracteristică, asigurați-vă că aveți instalată biblioteca <IRremote> sau reinstalați-o. În caz contrar, codul dvs. nu va funcționa.

Pentru detalii privind încărcarea unui fișier de bibliotecă, consultați Lecția 1.

Apoi, mutăm <RobotIRremote> din folderul Library deoarece această bibliotecă intră în conflict cu biblioteca pe care o vom utiliza. După ce ați terminat de programat microcontrolerul, trageți-l înapoi în folderul Library.

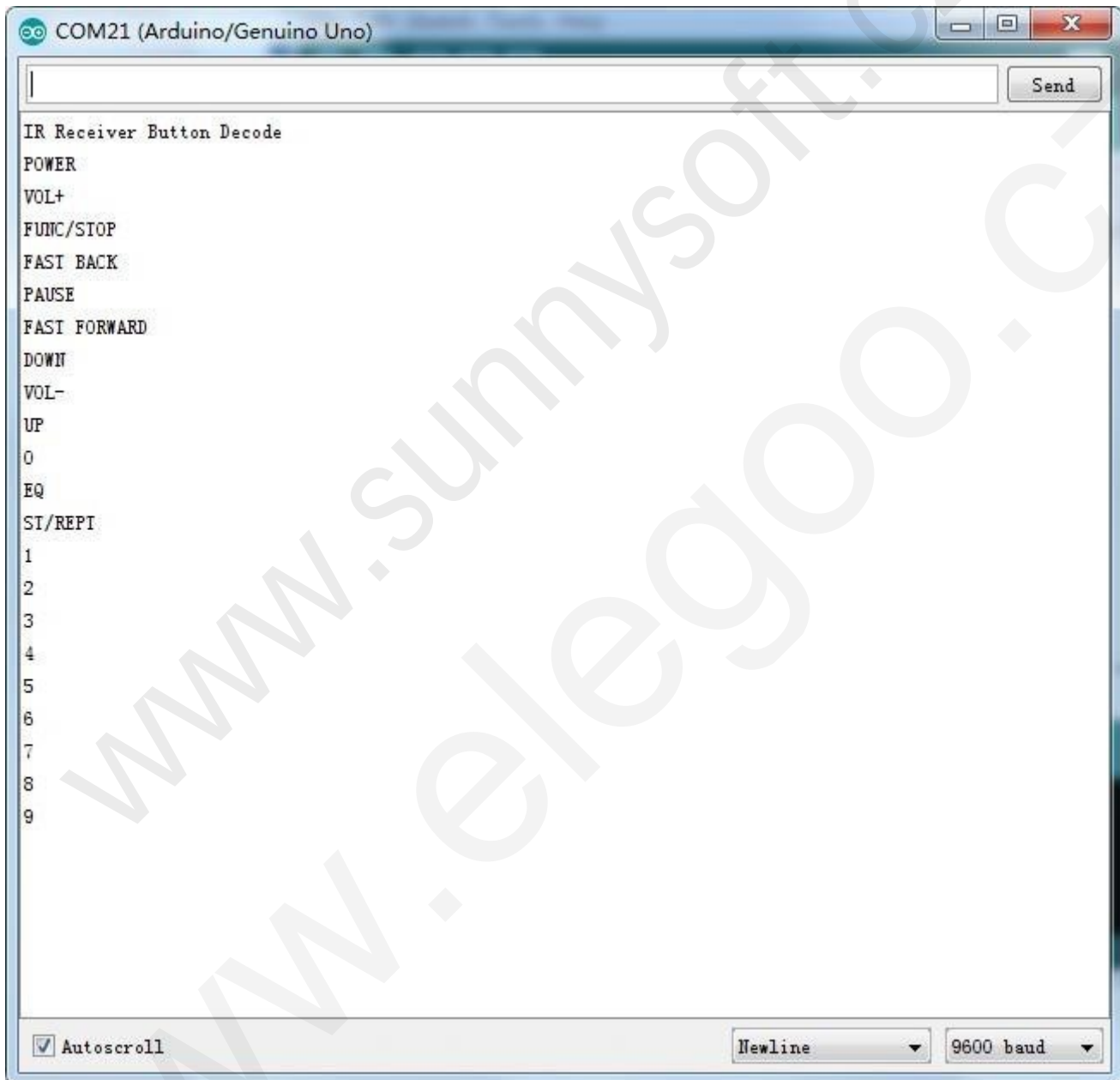
După instalarea bibliotecii, reporniți computerul.

Exemplu



Deschideți monitorul și apoi puteți vedea datele pe măsură ce suflă:

Faceți clic pe butonul Monitor serial pentru a porni monitorul serial. Informațiile de bază despre monitorul serial sunt prezentate în detaliu în Lecția 1.

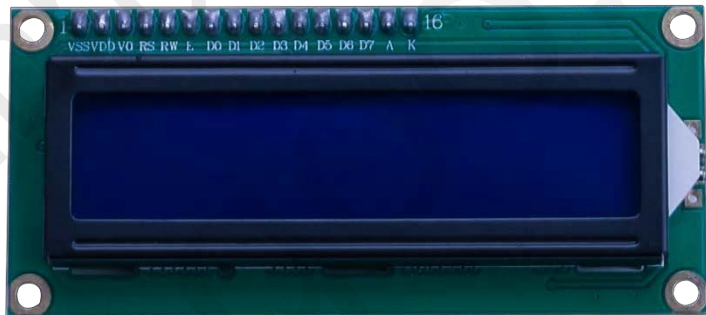


Lecția 14 Afișaj LCD

Prezentare generală

În această lecție veți învăța cum să conectați și să utilizați afișajul LCD alfanumeric. Afișajul are o lumină de fundal LED și poate afișa două linii cu până la 16 caractere pe fiecare linie. Pe afișaj, puteți vedea dreptunghiurile pentru fiecare caracter și pixelii care alcătuiesc fiecare caracter. Afișajul este alb doar pe un fundal albastru și este conceput pentru a afișa text.

În această lecție vom rula programul de exemplu Arduino pentru biblioteca LCD, dar în lecția următoare vom utiliza senzori pentru a face ca afișajul nostru să indice temperatura.



Componentă necesară:

- (1) x Elegoo Uno R3
- (1) x modul LCD1602
- (1) x potențiomtru (10k)
- (1) x 830 puncte de legare Placă de sârmă
- (16) x fire M-M (fire de conectare mascul la mascul)

Introducere în dosar

LCD1602

Familiaritate cu pinii LCD1602:

VSS: priză care se conectează la masă

VDD: Pin care se conectează la sursa de alimentare de +5 V.

VO: LCD1602 Priză de reglare a contrastului

RS: RS: Pin de selectare a registrului care controlează care parte a memoriei LCD este utilizată pentru scrierea datelor. Puteți selecta fie registrul de date, care scrie ceea ce apare pe ecran, fie registrul de instrucțiuni, în care controlerul LCD caută instrucțiuni cu privire la ce să facă în continuare.

R/W: Pin de citire/ scriere care selectează modul de citire sau scriere.

E: Pin de activare care determină modulul LDC să execute instrucțiunile corespunzătoare atunci când este furnizată alimentare la nivel scăzut.

D0-D7: Pinii de citire și scriere a datelor

A și K: Pini pentru controlul luminii de fundal LED

www.sunnysoft.cz

Conectarea Diagrama

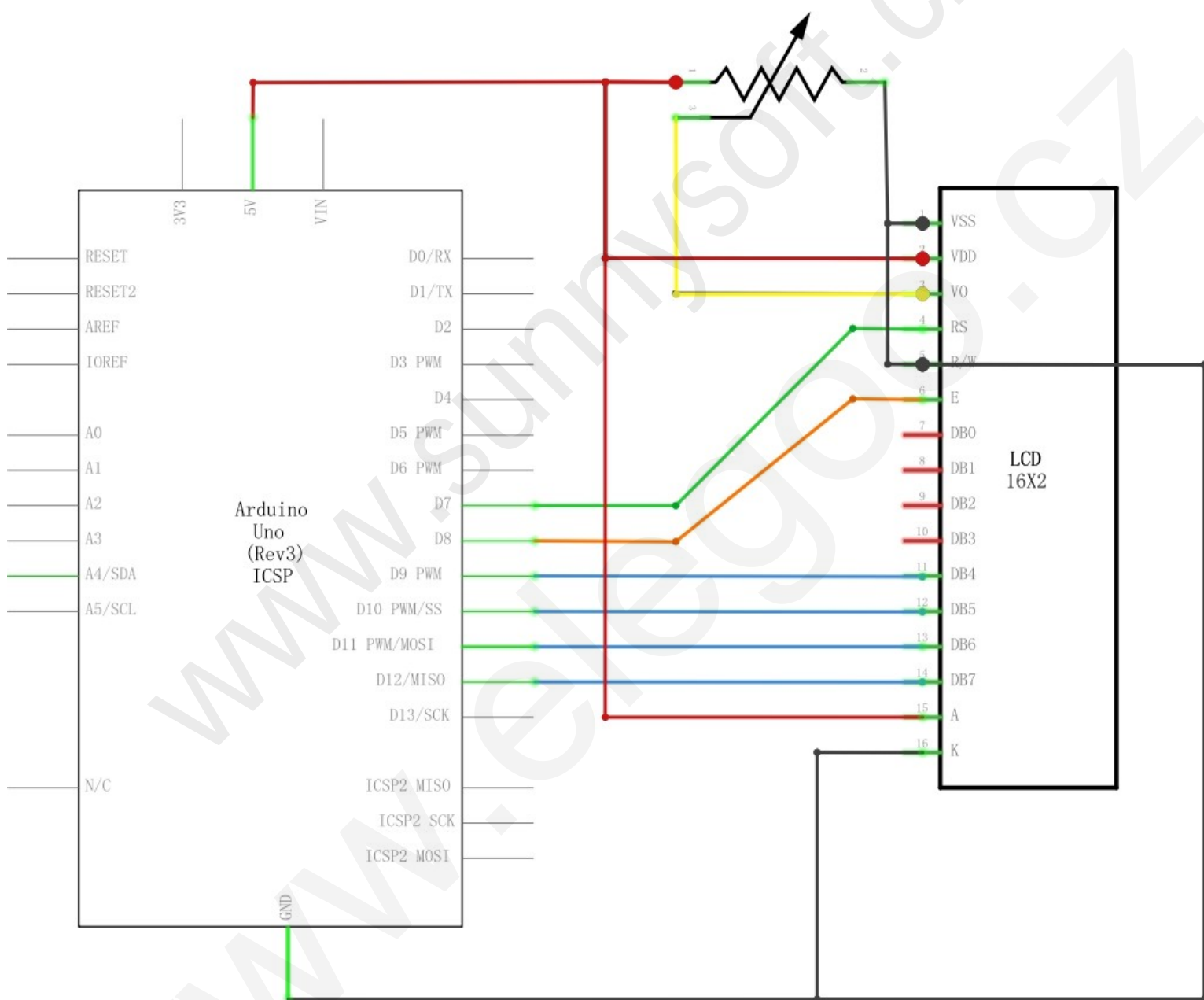
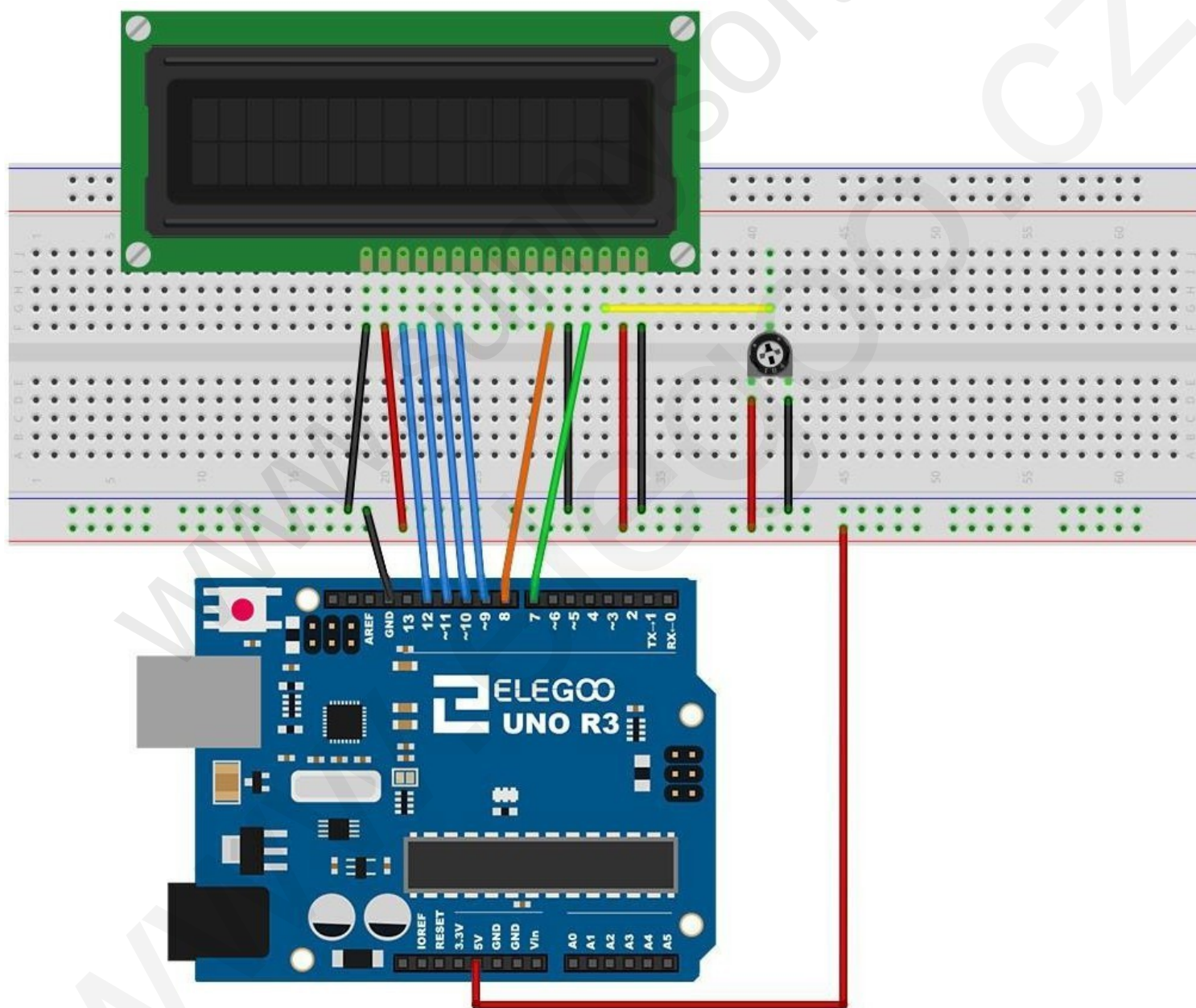


Diagrama de cablare



Afișajul LCD are nevoie de șase pini Arduino care sunt setați ca ieșiri digitale. De asemenea, are nevoie de conexiuni 5V și GND.

Există multe conexiuni. Alinierea afișajului cu partea superioară a plăcii ajută la identificarea pinilor săi fără o numărătoare lungă, mai ales dacă placa este numerotată pe rânduri, rândul 1 fiind rândul superior al plăcii. Nu uitați firul galben lung care conectează cursorul potențiometrului la pinul 3 al afișajului. "Potențialul" este utilizat pentru a controla contrastul afișajului.

Este posibil ca afișajul dvs. să fie livrat fără pini de antet conectați. În acest caz, urmați instrucțiunile din secțiunea următoare.

Cod

După conectare, deschideți programul din folderul de cod - Lecția 14 Afișaj LCD și faceți clic pe **UPLOAD** pentru a încărca programul. Consultați Lecția 2 pentru detalii despre cum să înregistrați programul dacă apar erori.

Înainte de execuție, asigurați-vă că aveți instalată biblioteca < LiquidCrystal >, sau reinstalați-l. În caz contrar, codul dvs. nu va funcționa.

Consultați Lecția 1 pentru detalii privind încărcarea fișierului de bibliotecă.

Încărcați codul pe placa Arduino și ar trebui să vedeți mesajul "hello, world" urmat de un număr care numără de la zero în jos.

Pe schiță, linia trebuie mai întâi să fie evidențiată:

```
#include <LiquidCrystal.h>
```

Acest lucru îi spune lui Arduino că doriți să utilizați biblioteca de cristale lichide.

În continuare avem o linie pe care a trebuit să o modificăm. Aceasta definește care pini Arduino ar trebui conectați la care pini de afișare.

```
Cristal lichid lcd(7, 8, 9, 10, 11, 12;
```

După încărcarea acestui cod, asigurați-vă că lumina de fundal este aprinsă și reglați potențiometrul până la capăt până când apare mesajul text.

În funcția "setup" avem două comenzi:

```
lcd.begin(16, 2;
```

```
lcd.print("Hello, World!
```

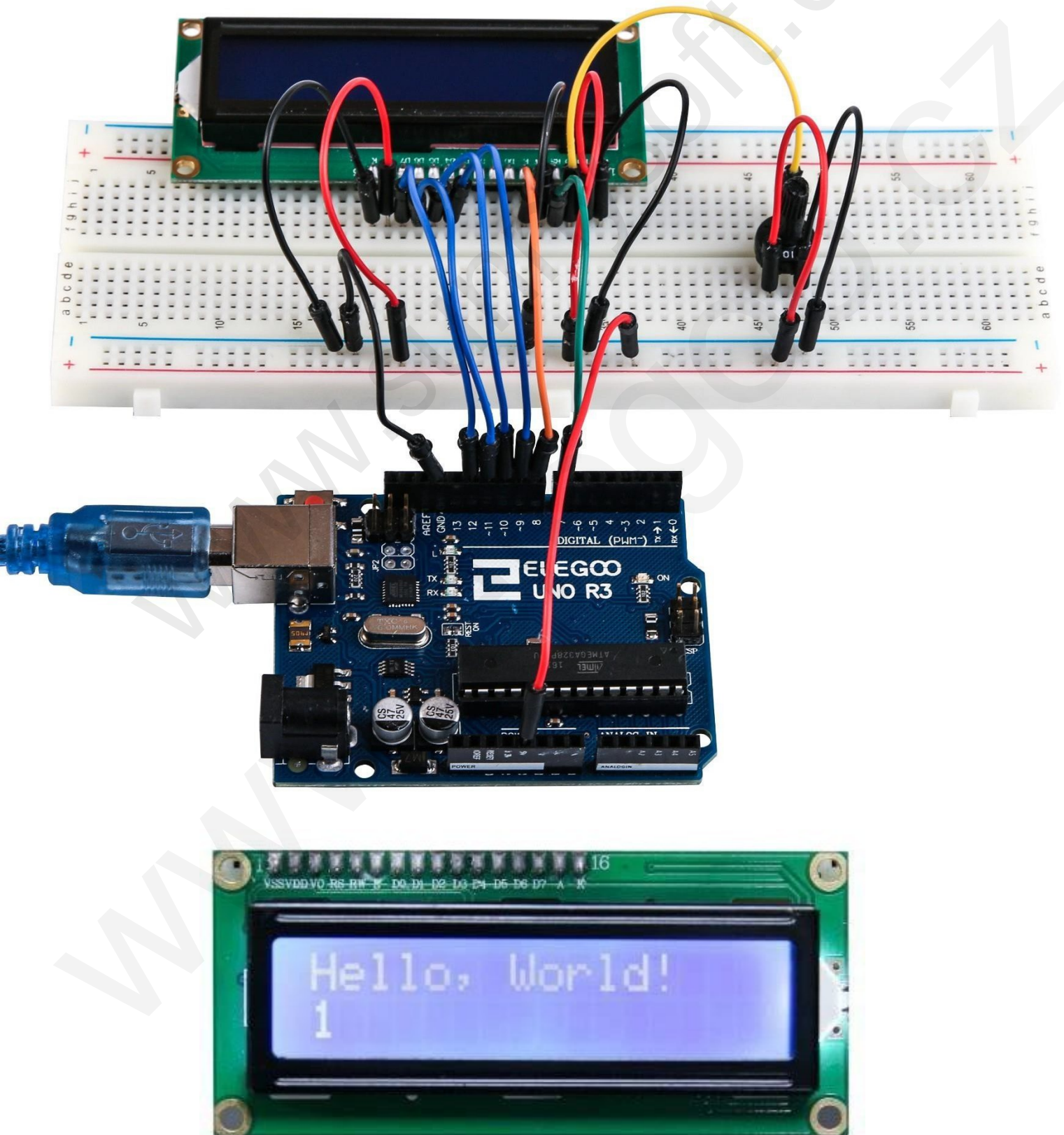
Primul indică bibliotecii cu cristale lichide câte coloane și câte rânduri are afișajul. Al doilea linie afișează mesajul pe care îl vedem pe prima linie a ecranului.

În funcția "loop" avem și două comenzi: `lcd.setCursor(0, 1;`

```
lcd.print(millis()/1000;
```


Primul stabilește poziția cursorului (unde va fi afișat următorul text) la coloana 0 și rândul 1. Numerele coloanelor și rândurilor încep de la 0, nu de la 1. A doua linie arată numărul de milisecunde de la resetarea Arduino.

Exemplu de imagine



Lecția 15 Termometru

Prezentare generală

În această lecție veți utiliza afișajul LCD pentru a afișa temperatura.

Componentă necesară:

- (1) x Elegoo Uno R3
- (1) x modul LCD1602
- (1) rezistor x 10k ohm
- (1) x termistor
- (1) x potențiometrul
- (1) x 830 puncte de legare Placă de sârmă
- (18) x fire M-M (fire de conectare)

Introducere în componentele termistorului

Un termistor este un rezistor termic - un rezistor care își modifică rezistența în funcție de temperatură. Din punct de vedere tehnic, toate rezistențele sunt termistoare - rezistența lor se modifică ușor cu temperatura - dar această modificare este de obicei foarte mică și dificil de măsurat. Termistorii sunt făcuți să își schimbe rezistența drastic în funcție de temperatură, astfel încât poate fi o schimbare de 100 ohmi sau mai mult pe grad!

Există două tipuri de termistoare: NTC (coeficient de temperatură negativ) și PTC (coeficient de temperatură pozitiv). În general, senzorii NTC sunt utilizați pentru măsurarea temperaturii. PTC-urile sunt adesea folosite ca siguranțe resetabile - creșterea temperaturii le va crește rezistența, ceea ce înseamnă că se vor încălzi și vor "strangula" curentul atunci când trece mai mult curent prin ele, ceea ce este destul de util pentru protecția circuitelor!

Conectarea Diagrama

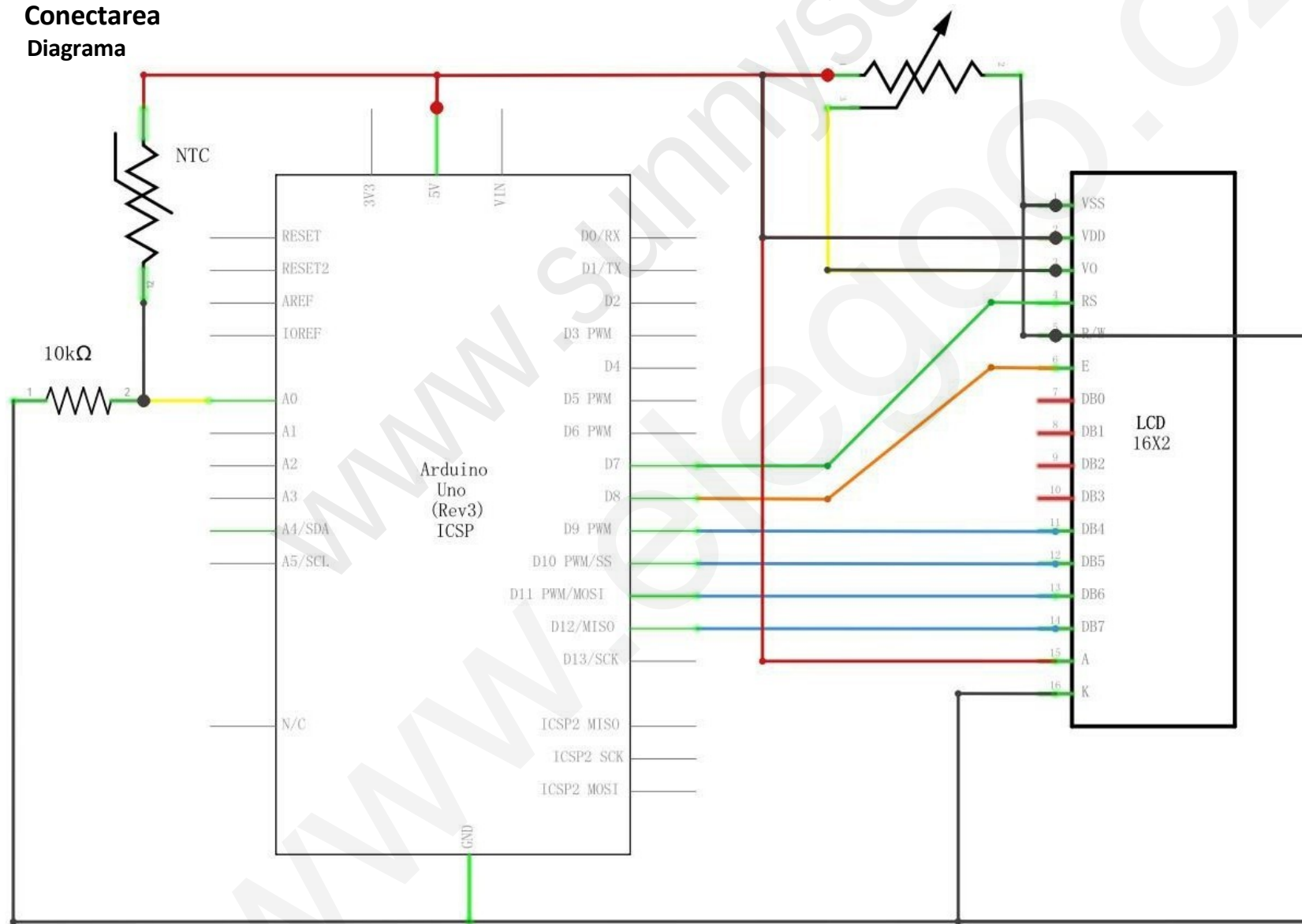
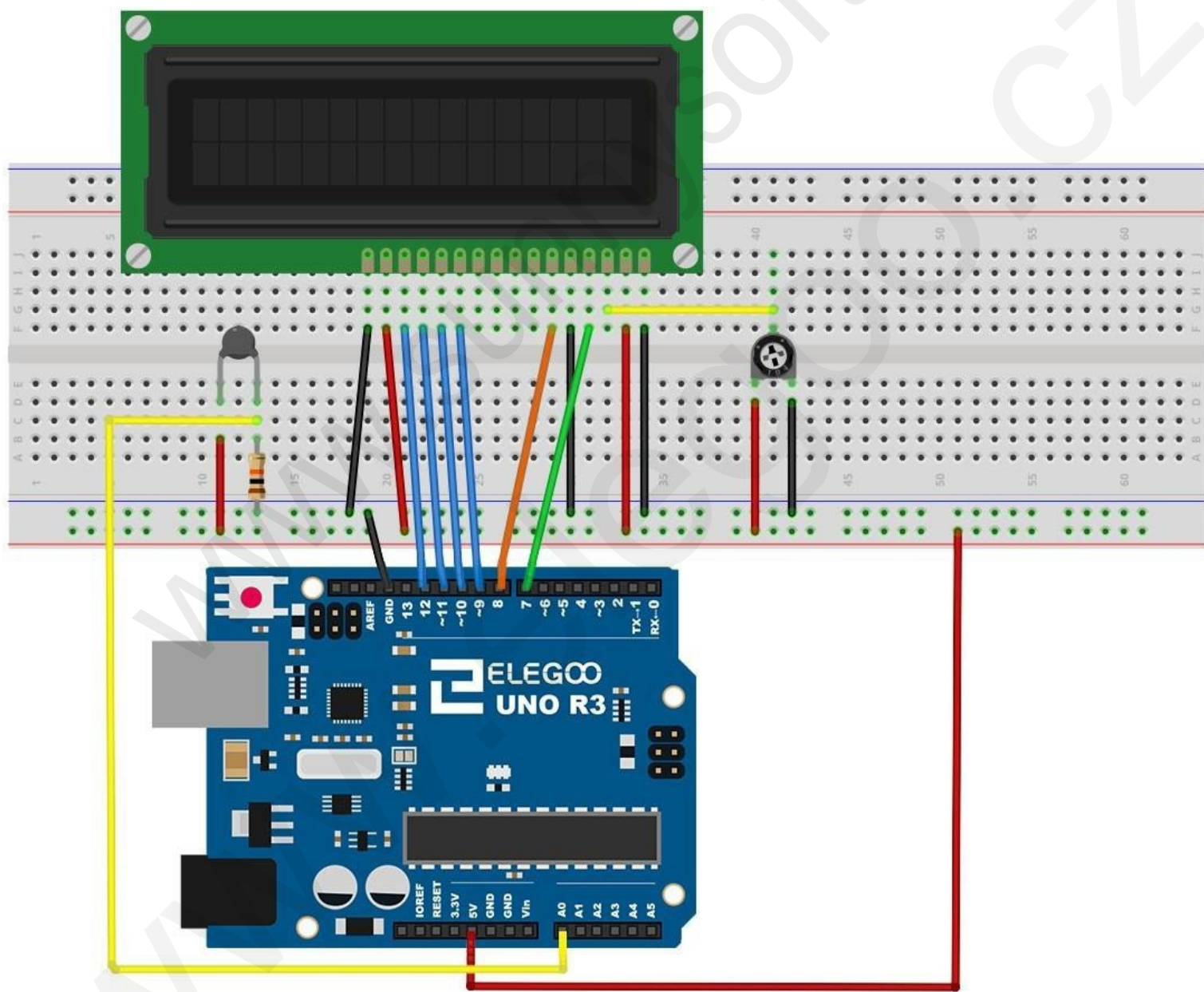


Diagrama de cablare



Disponerea de pe tablă se bazează pe disponerea din Lecția 22, deci se va simplifica dacă o aveți încă pe tablă.

Există mai multe fire jumper lângă potențiometru care au fost în acest aranjament s-a deplasat ușor.

Rezistorul de 10 kΩ și termistorul sunt noi pe placă.

Cod

După conectare, deschideți programul din folderul cu codul - Lecția 15 Termometru și faceți clic pe UPLOAD pentru a încărca programul. Consultați Lecția 2 pentru detalii privind încărcarea programului dacă apar erori.

Înainte de a utiliza această funcție, asigurați-vă că aveți instalată biblioteca < Liquid Crystal >, sau reinstalați-l. În caz contrar, codul dvs. nu va funcționa. Consultați Lecția 1 pentru detalii privind încărcarea fișierului de bibliotecă.

Schița se bazează pe lesso14. Încărcați-o pe Arduino și ar trebui să vedeți că se încălzește a senzorului de temperatură prin atașarea unui inel va crește
ishotepplot.

Mi se pare util să adaug o linie de comentariu deasupra comenzii "Icddc".

```
//      BSED4 D5 D6 D7  
Cristal lichid lcd(7, 8, 9, 10, 11,12);
```

Acest lucru vă va ușura munca dacă decideți să schimbați știfturile utilizate.

Acum se întâmplă două lucruri interesante în funcția "loop". În primul rând, trebuie să convertim datele analogice din

senzorilor de temperatură la temperatura reală și, în al doilea rând, trebuie să ne dăm seama cum să le afișăm.

Mai întâi să analizăm calculul temperaturii. `int`

```
tempReading =analogRead(tempPin);  
double tempK = log(10000.0 * ((1024.0 / tempReading -1)));  
tempK = 1 / (0,001129148 + (0,000234125 + (0,0000000876741 * tempK ))  
* tempK );  
float tempC = tempK - 273.15;  
float tempF = (tempC * 9.0)/ 5.0 + 32.0;
```

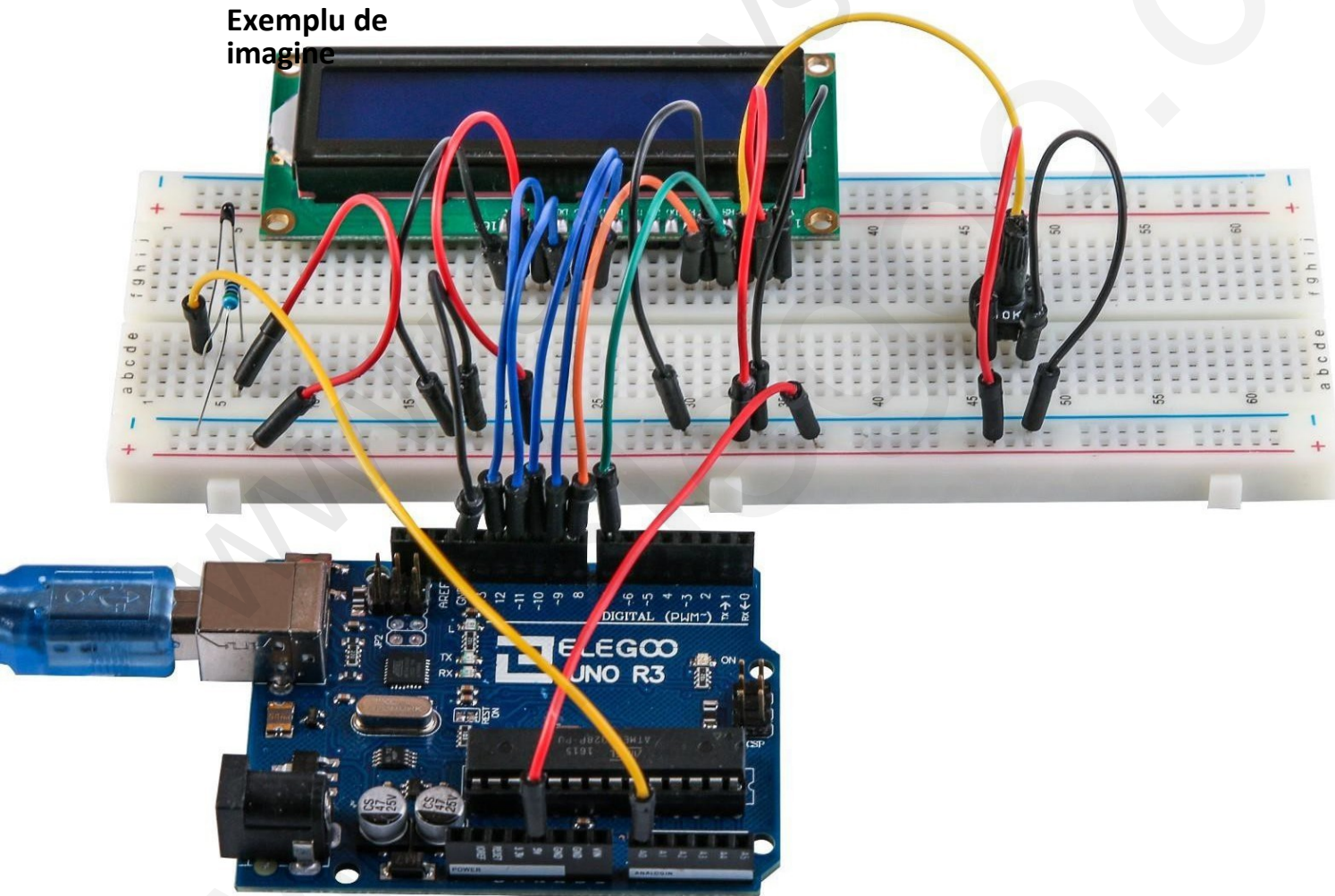
Afișarea valorilor variabile pe afișajul LCD poate fi complicată. Principala problemă este că citirea poate să nu aibă întotdeauna același număr de cifre. Astfel, dacă temperatura s-a schimbat de la 101,50 la 99,00, există riscul ca o cifră în plus de la citirea veche să rămână pe afișaj.

Pentru a evita acest lucru, scrieți de fiecare dată întreaga linie LCD în jurul buclei.

```
lcd.setCursor(0, 0);  
lcd.print("Temp      C  ");  
lcd.setCursor(6, 0);  
lcd.print(tempF);
```

Un comentariu destul de ciudat are rolul de a ne reaminti de cele 16 coloane ale afișajului. Puteți tasta apoi un șir de caractere de această lungime, cu spații unde va fi luată citirea efectivă. Pentru a completa spațiile libere, poziționați cursorul acolo unde doriți să apară citirea, apoi imprimați-o.

Exemplu de imagine



Lecția 16 Opt LED-uri cu 74HC595

Prezentare generală

În această lecție, veți învăța cum să utilizați opt LED-uri roșii mari cu UNO fără a renunța la 8 pini de ieșire!

Deși ați putea conecta opt LED-uri, fiecare cu o rezistență pe un pin UNO, ați începe rapid să rămâneți fără pini pe UNO. Cu excepția cazului în care aveți o mulțime de lucruri conectate la UNO. Asta e bine - dar se întâmplă adesea să vrem să avem butoane, senzori, servomotoare etc. și, înainte să vă dați seama, nu mai aveți niciun pin. Deci, în schimb, utilizați un cip numit 74HC595 Serial to Parallel Converter. Acest cip are opt ieșiri (perfecte) și trei intrări pe care le alimentați cu date în biți și bucăți.

Acest cip face controlul LED-urilor un pic mai lent (puteți schimba LED-urile doar de aproximativ 500.000 de ori pe secundă în loc de 8.000.000 pe secundă), dar este încă foarte rapid, mult mai rapid decât oamenii pot detecta, deci merită!

Componentă necesară:

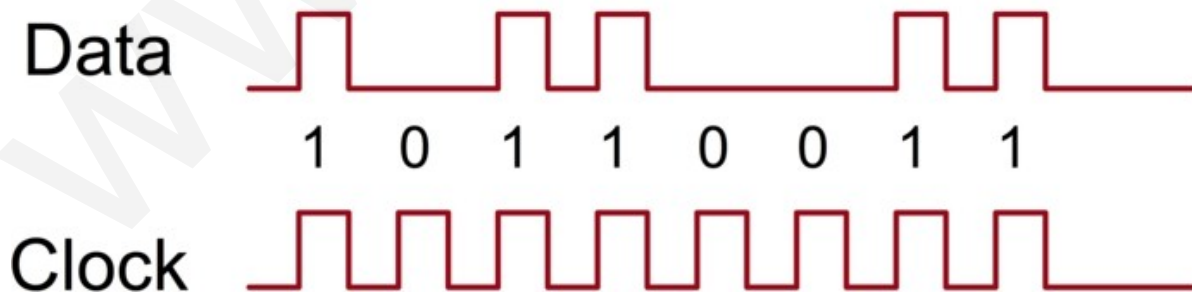
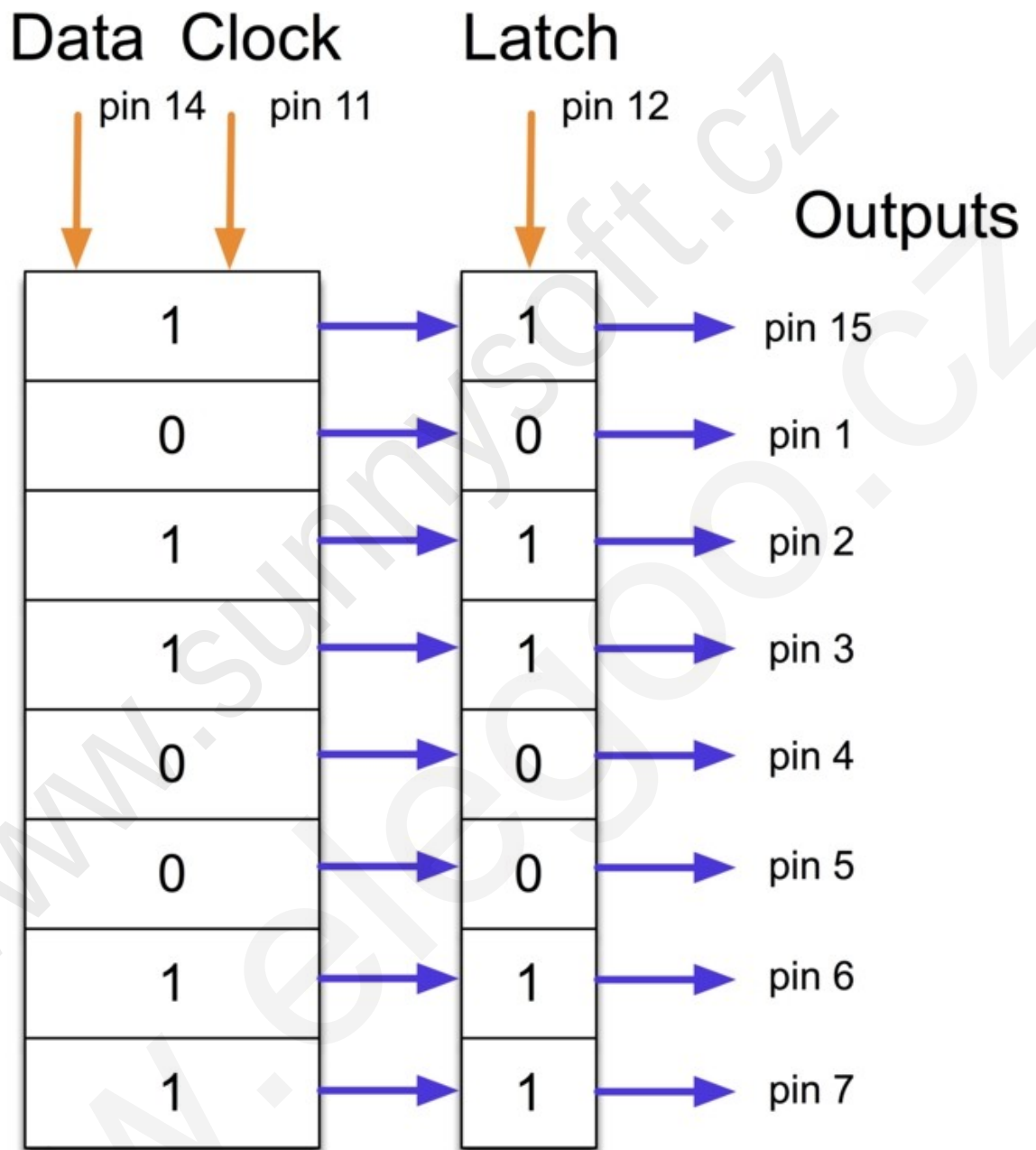
- (1) x Elegoo Uno R3
- (1) x 830 puncte de legare placă de pâine
- (8) x LED-uri
- (8) rezistențe x 220 ohm
- (1) x 74hc595 IC
- (14) x fire M-M (fire de conectare mascul la mascul)



Introducere în dosar

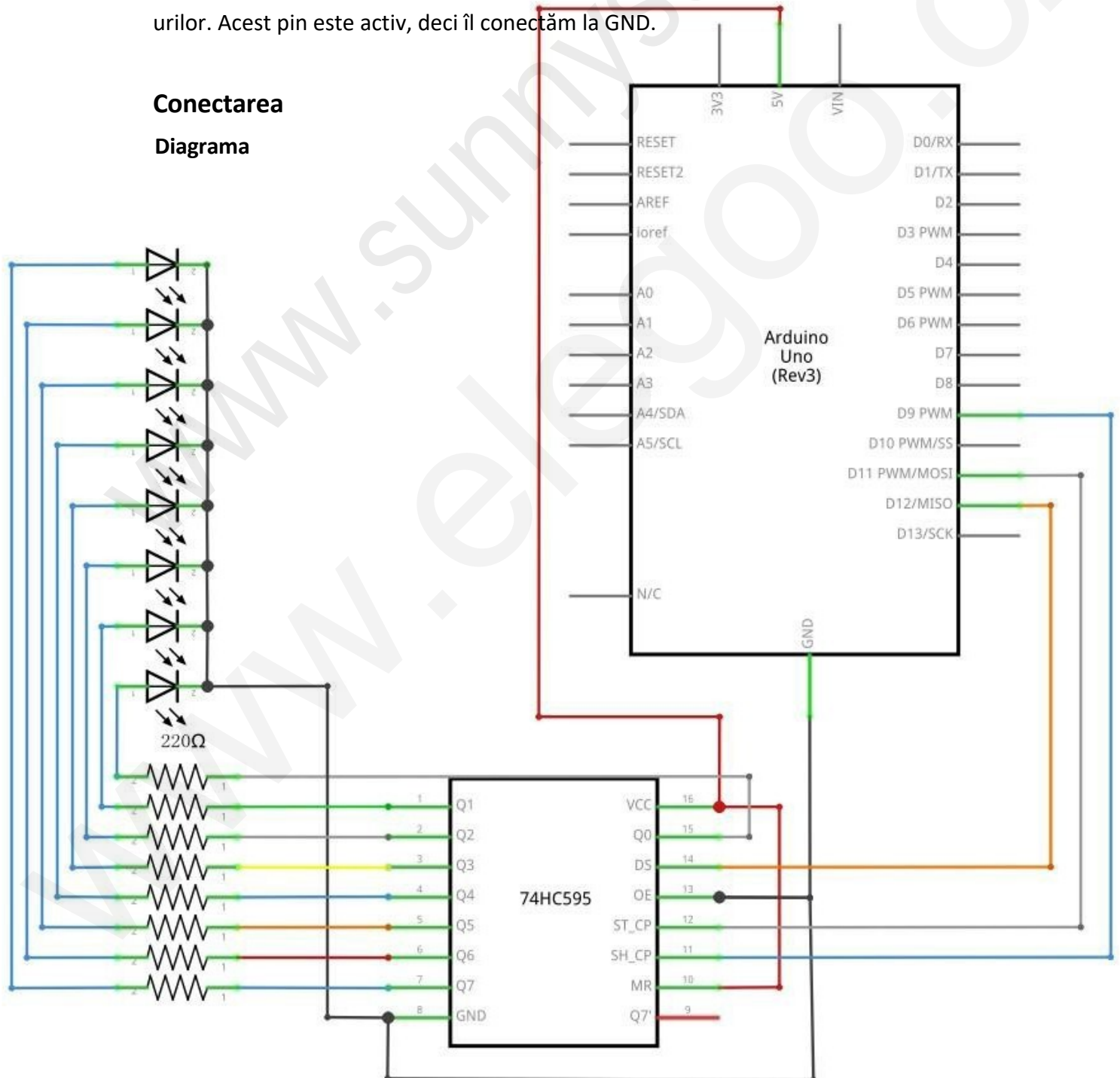
Registru de deplasare 74HC595:

Un registru de deplasare este un tip de cip care conține opt locații de memorie, fiecare dintre acestea putând avea o valoare de 1 sau 0. Pentru a activa sau dezactiva fiecare dintre aceste valori, introducem date prin pinii "Data" și "Clock" ai cipului.



Cipul are, de asemenea, un pin de activare a ieșirilor (OE), care este utilizat pentru a activa sau dezactiva toate ieșirile simultan. Îl puteți conecta la pinul UNO activat pentru PWM și puteți utiliza "analogWrite" pentru a controla luminozitatea LED-urilor. Acest pin este activ, deci îl conectăm la GND.

Diagrama



Deoarece avem opt LED-uri și opt rezistențe de conectat, este destul de multe conexiuni de făcut.

Este probabil cel mai ușor să introduceți mai întâi cipul 74HC595, deoarece acesta se conectează la aproape toate celelalte. Plasați-l astfel încât mica creștătură în formă de U să fie orientată spre partea superioară a plăcii. Pinul 1 al cipului este la stânga acestei creștături.

Digital 12 de la UNO merge la pinul #14 al registrului de schimbare Digital 11 de la UNO merge la pinul #12 al registrului de schimbare Digital 9 de la UNO merge la pinul #11 al registrului de schimbare.

Toate ieșirile circuitului integrat, cu excepția uneia, sunt pe partea stângă a cipului.

Acesta este motivul pentru care există și LED-uri pentru o cablare mai ușoară.

Plasați rezistențele în spatele cipului. Trebuie să aveți grijă ca firele rezistențelor să nu se atingă între ele. Trebuie să verificați de două ori acest lucru înainte de a conecta alimentarea la UNO. Dacă vi se pare dificil să aranjați rezistențele astfel încât firele lor să nu se atingă, va fi util să scurtați firele astfel încât acestea să se afle mai aproape de suprafața plăcii.

Apoi așezați LED-urile pe placă. Toate firele pozitive mai lungi ale LED-urilor trebuie la cip, indiferent pe ce parte a plăcii se află.

Conectați firele de legătură după cum se arată mai sus. Nu uitați de cel care duce de la pinul 8 al circuitului integrat la coloana GND de pe placă.

Încărcați schița puțin mai târziu și încercați-o. Fiecare LED ar trebui să se aprindă pe rând până când toate LED-urile sunt aprinse, apoi toate se sting și ciclul se repetă.

Cod

După conectare, deschideți programul din folderul de cod - Lecția 16 Opt LED-uri cu

74HC595 și faceți clic pe butonul UPLOAD pentru a încărca programul.

Consultați Lecția 2 pentru detalii privind încărcarea programului în caz de erori.

Mai întâi definim cei trei pini pe care îi vom utiliza. Acestea sunt ieșirile digitale UNO care vor fi conectate la pinii de blocare, ceas și date ai circuitului 74HC595.

```
int latchPin = 11;
```

```
int clockPin = 9;
```

```
int dataPin = 12;
```

În continuare, este definită variabila "leds". Această variabilă va conține modelul LED-urilor care sunt în prezent aprinse sau stinse. Tipul de date "byte" reprezintă numere folosind opt biți. Fiecare bit poate fi pornit sau oprit, deci este perfect pentru urmărirea

care dintre cele opt LED-uri ale noastre sunt aprinse sau stinse.

```
byte leds = 0;
```

Funcția "setup" setează numai cei trei pini pe care îi folosim ca ieșiri digitale.

```
void setup()
{
    pinMode(latchPin, OUTPUT);
    pinMode(dataPin, OUTPUT);
    pinMode(clockPin, OUTPUT);
}
```

Funcția "loop" stinge mai întâi toate LED-urile atribuind variabilei "leds" o valoare de 0. Apoi apelează funcția "updateShiftRegister", care trimite modelul "leds" către registrul de deplasare, astfel încât toate LED-urile să se stingă. Vom vedea mai târziu cum funcționează funcția "updateShiftRegister".

Funcția buclei se oprește timp de o jumătate de secundă și apoi începe numărătoarea de la 0 la 7 folosind bucla "for" și variabila "i". De fiecare dată, funcția utilizează funcția Arduino "bitSet" pentru a seta bitul care controlează LED-ul respectiv în variabila "leds". Apoi apelează și funcția "updateShiftRegister" pentru a actualiza LED-urile în funcție de ceea ce se află în variabila "leds". Există apoi o întârziere de o jumătate de secundă înainte ca valoarea "i" să fie incrementată și următorul LED să fie aprins.

```
void loop()
{
    leds = 0;
    updateShiftRegister();
    delay(500);
    pentru (int i = 0; i < 8; i++)
    {
        bitSet(leds, i);
        updateShiftRegister();
        delay(500);
    }
}
```

Funcția "updateShiftRegister" stabilește mai întâi latchPin la o valoare mică, apoi apelează funcția UNO "shiftOut" și abia apoi stabilește din nou latchPin la o valoare mare. Această funcție acceptă patru parametri, primii doi fiind pinii care urmează să fie utilizați pentru date și, respectiv, pentru ceas.

Al treilea parametru specifică de la ce capăt al datelor doriți să începeți. Începem cu bitul cel mai din dreapta, care este denumit "bitul cel mai puțin semnificativ".

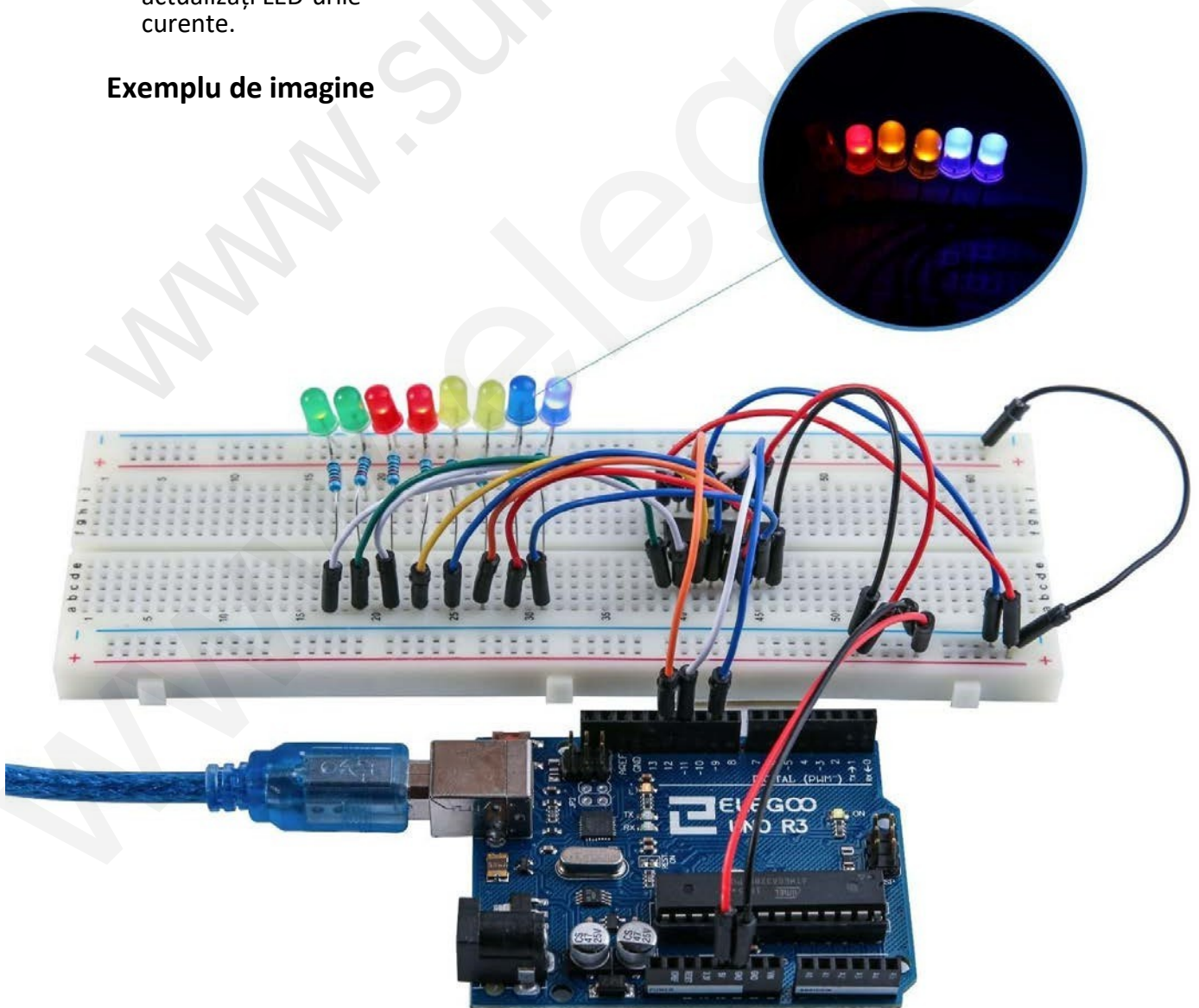
Bit" (LSB).

Ultimul parametru reprezintă datele reale care urmează să fie deplasate în registrul de deplasare, care în acest caz este "leds".

```
void updateShiftRegister()  
{  
    digitalWrite(latchPin, LOW);  
    shiftOut(dataPin, clockPin, LSBFIRST, leds);  
    digitalWrite(latchPin, HIGH);  
}
```

Dacă doriți să opriți unul dintre LED-uri în loc să îl aprindeți, apăsați o funcție Arduino similară (bitClear) cu variabila "leds". Aceasta ar seta bitul "leds" la 0 și apoi ați apela doar funcția "updateShiftRegister" pentru a actualizați LED-urile curente.

Exemplu de imagine



Lecția 17 Monitor serial

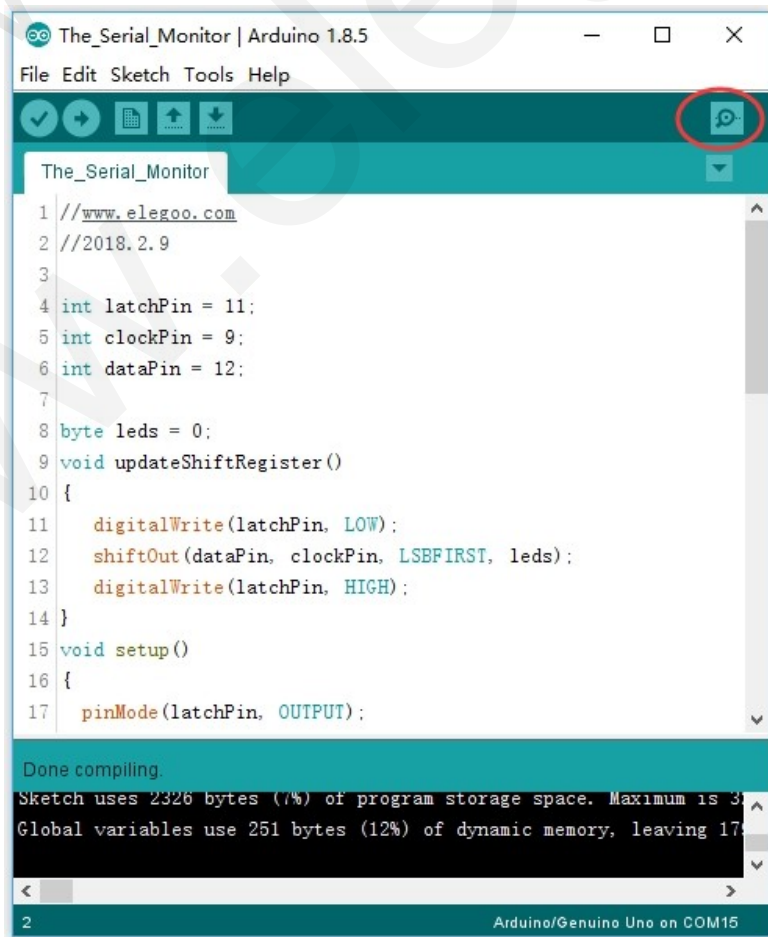
Prezentare generală

În această lecție vă veți baza pe lecția 16 și veți adăuga capacitatea de a controla LED-urile de pe computer utilizând un monitor serial Arduino. Monitorul serial este "legătura" dintre computerul dvs. și UNO. Acesta vă permite să trimiteți și să primiți mesaje text, ceea ce este util la depanare și, de asemenea, atunci când controlați UNO de la tastatură! De exemplu, veți putea trimite comenzi de la computer pentru a aprinde LED-urile. În această lecție, veți utiliza exact aceleași componente și o schemă de placă similară celei din Lecția 16. Dacă nu ați făcut-o deja, urmați lecția 16 acum.

Măsurile luate

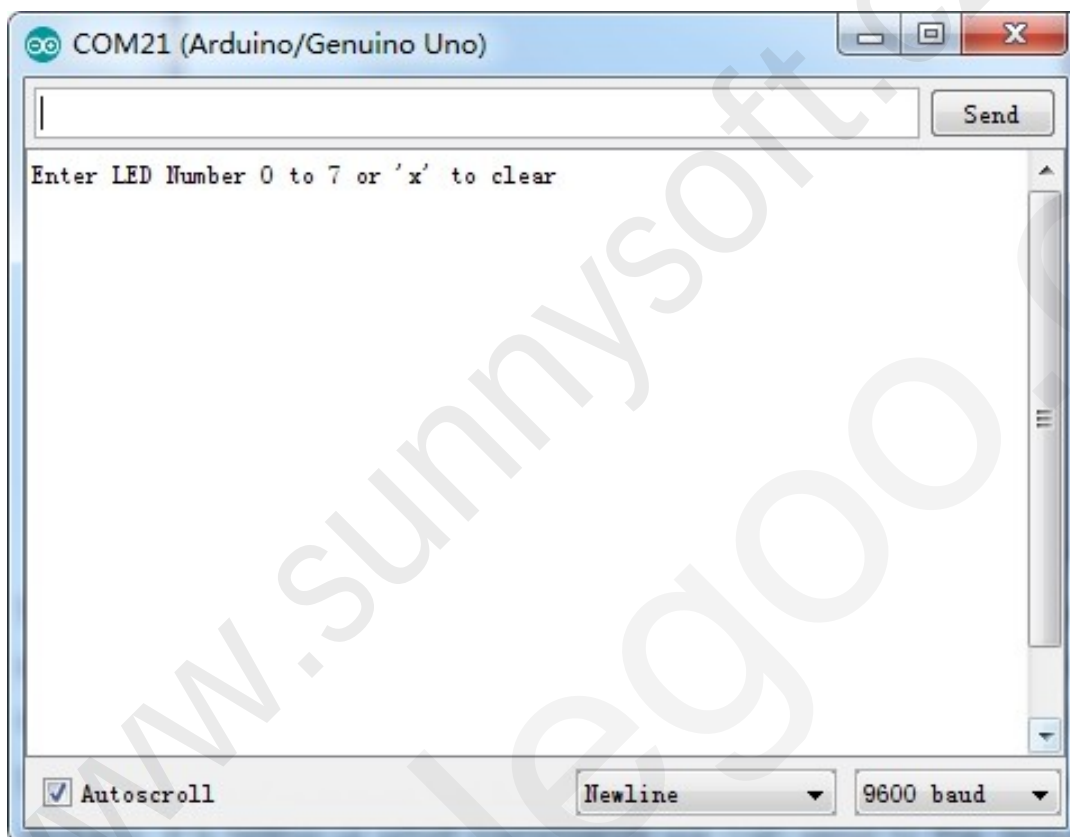
După încărcarea acestei schițe în UNO, faceți clic pe butonul din dreapta de pe bara de instrumente

în Arduino IDE. Butonul este încercuit mai jos.



Se va deschide următoarea fereastră.

Faceți clic pe butonul Monitor serial pentru a porni monitorul serial. Informațiile de bază despre monitorul serial sunt prezentate în detaliu în Lecția 1.

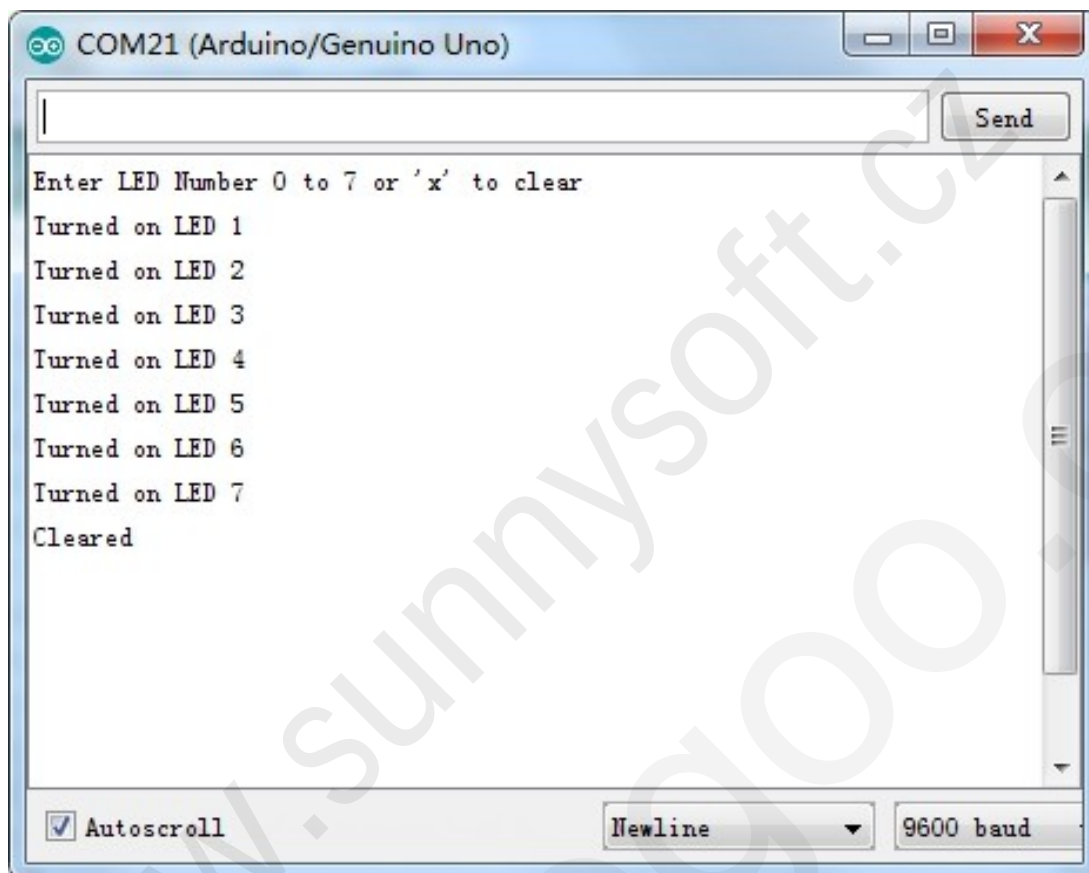


Această fereastră se numește Serial Monitor și face parte din software-ul Arduino IDE. Scopul său este de a permite trimiterea de mesaje de la computer la placa UNO (prin USB) și, de asemenea, de a primi mesaje de la placa UNO.

Arduino a trimis mesajul "Enter LED number 0 to 7 or 'x' to clear". Acesta ne spune ce comenzi putem trimite către Arduino: fie trimiteți "x" (pentru a opri toate LED-urile), fie numărul LED-ului pe care doriți să îl aprindeți (unde 0 este LED-ul de jos, 1 este următorul de sus, până la 7 pentru LED-ul de sus).

Încercați să introduceți următoarele comenzi în zona superioară a monitorului serial, care este la nivelul butonului "Send". După introducerea fiecăruia dintre aceste caractere, apăsați butonul "Send": x 0 3 5

Introducerea x nu va avea niciun efect dacă toate LED-urile sunt deja stinse, dar după introducerea fiecărui număr, LED-ul corespunzător ar trebui să se aprindă și un mesaj de confirmare va fi afișat de la placa UNO. Monitorul serial va fi afișat așa cum se arată în figura următoare.



Introduceți din nou x și apăsați butonul "Submit" pentru a opri toate LED-urile.

Cod

După conectare, deschideți programul din folderul de cod - [Lecția 17 Monitor serial](#) și faceți clic pe **UPLOAD** pentru a încărca programul. Consultați [Lecția 2](#) pentru detalii privind încărcarea programului dacă apar erori.

După cum v-ați putea aștepta, schița se bazează pe schița utilizată în [Lecția 16](#). Prin urmare, doar piesele noi vor fi discutate aici. Veți găsi util link-ul către schița completă din Arduino IDE.

Există trei linii noi la sfârșitul funcției "setup": **void setup()**

```
{  
  pinMode(latchPin, OUTPUT);  
  pinMode(dataPin, OUTPUT);  
  pinMode(clockPin, OUTPUT);  
  updateShiftRegister();  
  Serial.begin(9600);
```

117/162

```
while (! Serial); // Așteptați până când Serial este gata -  
Leonardo Serial.println("Introduceți numărul de LED-uri de la  
0 la 7 sau 'x' pentru a șterge");  
}
```

Mai întâi avem comanda "Serial.begin(9600)". Aceasta pornește comunicarea serială astfel încât UNO să poată trimite comenzi prin conexiunea USB. Valoarea 9600 se numește "rata de baud" a conexiunii. Aceasta este viteza la care trebuie trimise datele. Puteți modifica această valoare la o valoare mai mare, dar va trebui să modificați și monitorul Arduino Serial la aceeași valoare. Vom vorbi despre asta mai târziu; pentru moment, lăsați-o la 9600.

Linia care începe cu "while" asigură că există ceva la celălalt capăt al conexiunii USB cu care Arduino poate comunica înainte de a începe să trimită mesaje. În caz contrar, un mesaj ar putea fi trimis, dar nu și afișat. Această linie este într-adevăr necesară numai dacă utilizați un Arduino Leonardo, deoarece Arduino UNO resetează automat placa Arduino atunci când deschideți Serial Monitor, în timp ce acest lucru nu se întâmplă cu Leonardo.

Ultima dintre liniile noi din "setup" trimite mesajul pe care îl vedem în partea de sus a monitorului serial.

Funcția "loop" este cea în care are loc întreaga

acțiune: `void loop()`

```
{  
  dacă (Serial.available())  
  {  
    char ch = Serial.read();  
    if (ch >= '0' && ch <= '7')  
    {  
      int led = ch - '0';  
      bitSet(leds, led);  
      updateShiftRegister();  
      Serial.print("LED pornit");  
      Serial.println(led);  
    }  
    dacă (ch == 'x')  
    {  
      leds = 0;  
      updateShiftRegister();  
    }  
  }  
}
```

```

        Serial.println("Curățat");
    }
}
}

```

Tot ceea ce se întâmplă în interiorul buclei este conținut în declarația "if". Deci, dacă apelul Funcția încorporată în Arduino "Serial.available()" nu este "true", nu se întâmplă nimic altceva. Serial.available() va returna "true" dacă datele au fost trimise către UNO și sunt gata să fie procesate. Mesajele primite sunt păstrate într-un așa-numit buffer, iar Serial.available() returnează true dacă bufferul nu este gol.

Dacă mesajul a fost primit, se trece la următoarea linie de cod: **char ch = Serial.read();**

Aceasta încarcă următorul caracter din buffer și îl elimină din buffer. De asemenea, îl atribuie variabilei "ch". Variabila "ch" este de tip "char", ceea ce înseamnă "caracter", și, după cum sugerează și numele, conține un singur caracter.

Dacă ați urmat instrucțiunile din promptul din partea de sus a monitorului serial, acest caracter va fi fie un număr cu o singură cifră între 0 și 7, fie litera "x".

Instrucțiunea "if" de pe linia următoare verifică dacă este o singură cifră verificând dacă "ch" este mai mare sau egal cu "0" și mai mic sau egal cu "7". Acest tip de potrivire a caracterelor arată un pic ciudat, dar este perfect acceptabil.

Fiecare caracter este reprezentat de un număr unic numit valoare ASCII. Aceasta înseamnă că atunci când comparăm caractere folosind <= și >=, comparăm de fapt valori ASCII.

Dacă testul are succes, trecem la linia următoare:

```
int led = ch - "0";
```

Acum facem aritmetică pe semne! Scădeți cifra "0" din cifra dată. Deci, dacă ați introdus "0", atunci "0" - "0" va fi egal cu 0. Dacă ați introdus "7", atunci "7" - "0" va fi egal cu 7, deoarece scăderea utilizează de fapt valori ASCII.

Deoarece știm numărul LED-ului pe care dorim să îl aprindem, trebuie doar să setăm acest bit în

variabilele "leds" și actualizează registrul de

deplasare. **bitSet(leds, led);**

```
updateShiftRegister();
```


Următoarele două linii trimit înapoi un mesaj de confirmare către monitorul serial. `Serial.print("LED aprins ");`

`Serial.println(led);`

Prima linie utilizează `Serial.print` în loc de `Serial.println`. Diferența dintre cele două este că `Serial.print` nu începe o linie nouă după ce imprimă ceea ce se află în parametrul său. Folosim acest lucru în prima linie deoarece imprimăm mesajul în două părți. Mai întâi partea generală: apoi este afișat numărul LED-ului.

Numărul LED-ului este stocat în variabila "int", nu într-un șir de text. Fișierul `Serial.print` poate accepta fie un șir de text citat, fie o variabilă "int", sau practic orice tip de variabilă.

Instrucțiunea "if", care tratează cazul în care a fost procesată o singură cifră, este urmată de o a doua instrucțiune "if" care determină dacă "ch" este litera "x".

```
dacă (ch == 'x')
{
    leds = 0; updateShiftRegister();
    Serial.println("Cleared");
}
```

Dacă da, se șterg toate LED-urile și se trimite un mesaj de confirmare.

Lecția 18 Photobooth

Prezentare generală

În această lecție veți învăța cum să măsurați intensitatea luminii utilizând o intrare analogică. Vă veți baza pe lecția 16 și veți utiliza nivelul de lumină pentru a controla numărul de LED-uri care se aprind.

Celula foto se află pe partea inferioară a plăcii, unde vasul era deasupra.

Componentă necesară:

- (1) x Elegoo Uno R3
- (1) x 830 puncte de legare placă de pâine
- (8) x LED-uri
- (8) rezistențe x 220 ohm
- (1) x 1k ohm rezistență
- (1) x 74hc595 IC
- (1) x fotorezistor (fotocelulă)
- (16) x fire M-M (fire de conectare mascul la mascul)



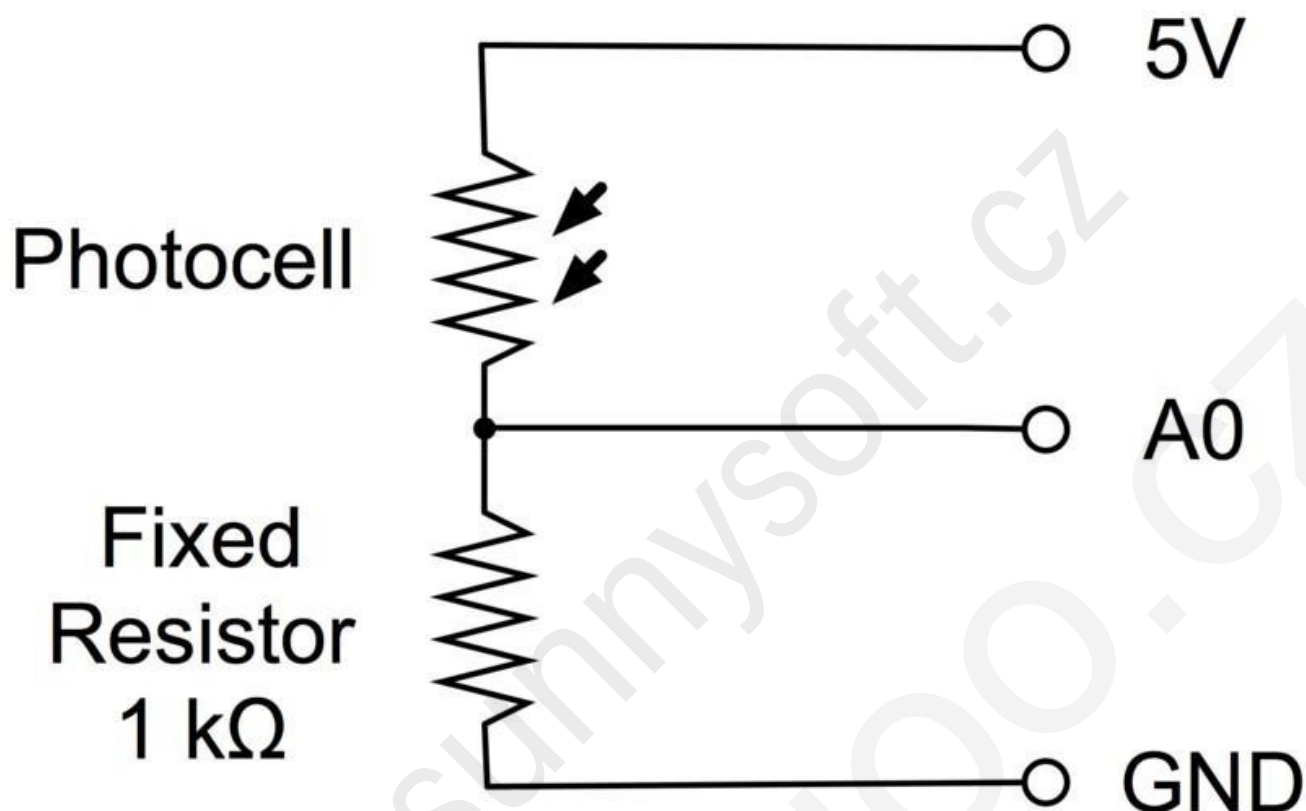
Introducere în dosar

FOTOCCELL:

Fotocelula utilizată este un tip numit rezistență dependentă de lumină, denumită uneori LDR. După cum sugerează și numele, aceste componente se comportă la fel ca un rezistor, dar rezistența lor se modifică în funcție de cantitatea de lumină care le atinge.

Aceasta are o rezistență de aproximativ 50 k Ω în întuneric și 500 Ω în lumină puternică. Pentru a converti această valoare variabilă a rezistenței în ceva ce putem măsura pe intrarea analogică a plăcii UNO R3, trebuie să o convertim într-o tensiune.

Cel mai simplu mod de a realiza acest lucru este de a-l combina cu un rezistor fix.

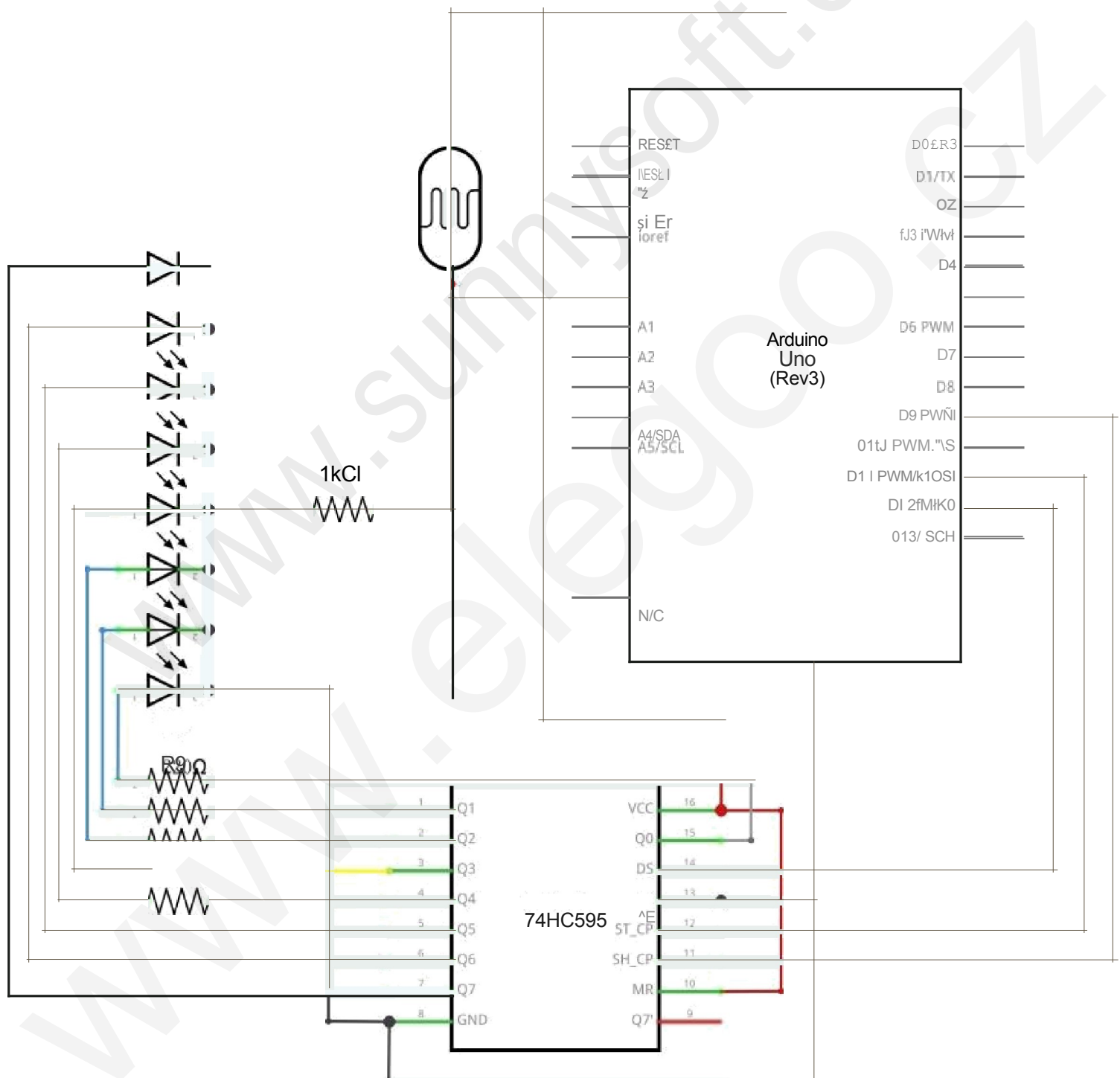


Rezistorul și fotocuplorul acționează împreună ca un potențiometrul. Atunci când lumina este foarte puternică, rezistența fotocelulei este foarte scăzută în comparație cu o rezistență cu valoare fixă, astfel încât este ca și cum potențiometrul ar fi ridicat la maximum.

Atunci când fotocelula este în lumină slabă, rezistența este mai mare decât rezistența fixă de 1 kΩ și este ca și cum potențiometrul ar fi îndreptat spre GND.

Citiți schița din secțiunea următoare și încercați să acoperiți fotocelula cu degetul și apoi să o țineți lângă sursa de lumină.

Conectarea Diagrama



Cod

După conectare, deschideți programul din folderul cu codul - Lecția 18 Photobooth și faceți clic pe butonul UPLOAD pentru a încărca programul. Consultați Lecția 2 pentru detalii privind încărcarea programului în cazul în care apar erori.

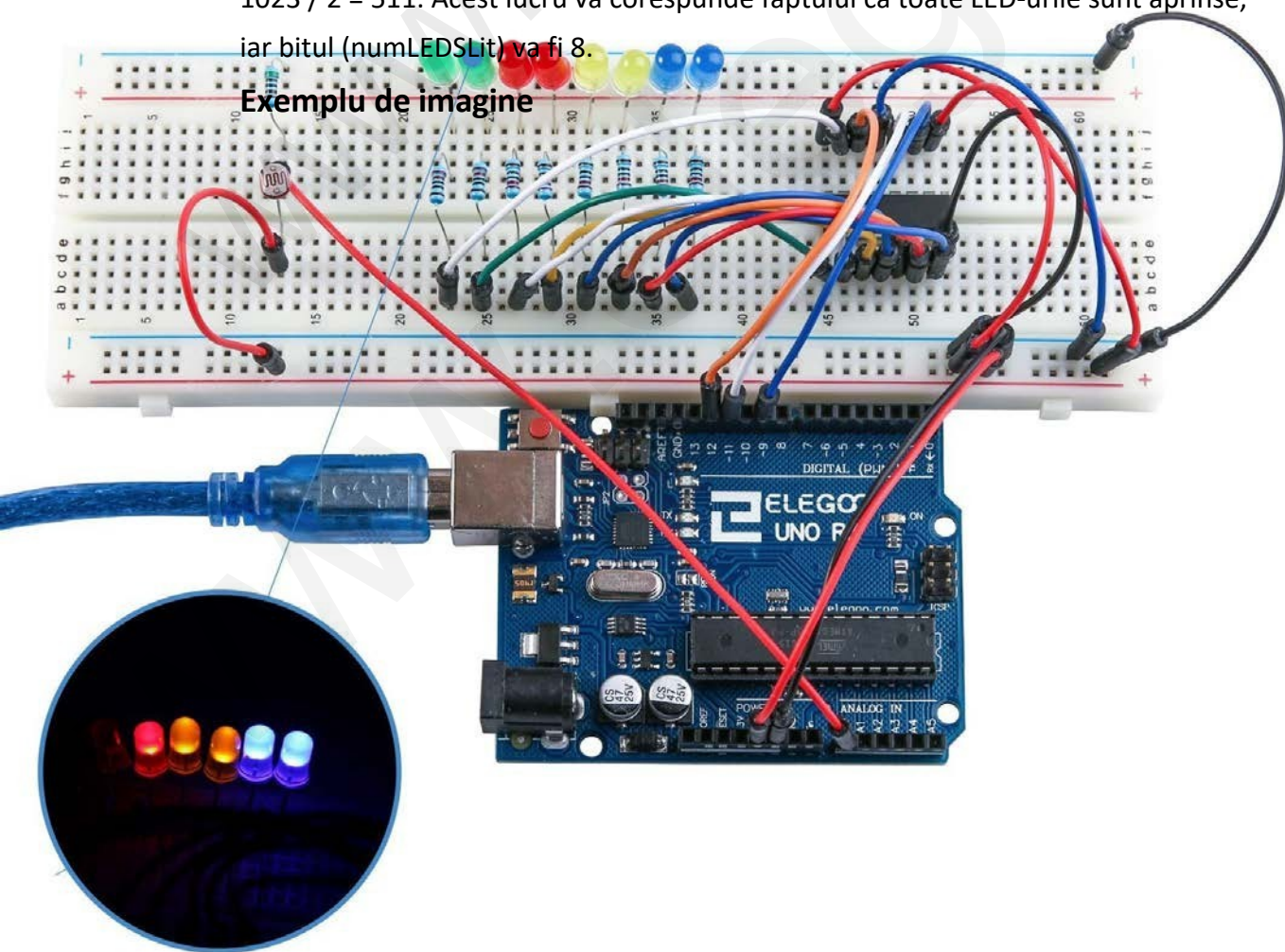
În primul rând, rețineți că am schimbat numele pinului analogic în "lightPin" și nu în "potPin" deoarece nu mai avem un potențiometru conectat.

Singura altă modificare semnificativă din schiță este linia care calculează câte LED-uri trebuie aprinse:

```
int numLEDsLit = reading / 57; // toate LED-urile sunt aprinse la 1k
```

De data aceasta împărțim valoarea brută la 57 în loc de 114. Cu alte cuvinte, o împărțim la jumătate, așa cum am făcut pentru oală, pentru a o împărți în nouă zone, de la niciun LED aprins la toate cele opt LED-uri aprinse. Acest factor suplimentar ia în considerare rezistența fixă de 1 k Ω . Aceasta înseamnă că, dacă fotocelula are o rezistență de 1 k Ω (aceeași ca rezistența fixă), cifra aproximativă va fi $1023 / 2 = 511$. Acest lucru va corespunde faptului că toate LED-urile sunt aprinse, iar bitul (numLEDsLit) va fi 8.

Exemplu de imagine



Lecția 19 74HC595 și afișajul cu segmente

Prezentare generală

După finalizarea Lecției 16, 17 și a Lecției 18, utilizați registrul de deplasare 74HC595 pentru a controla afișajul cu segmente. Afișajul cu segmente va afișa numere de la 9 la 0.

Componentă necesară:

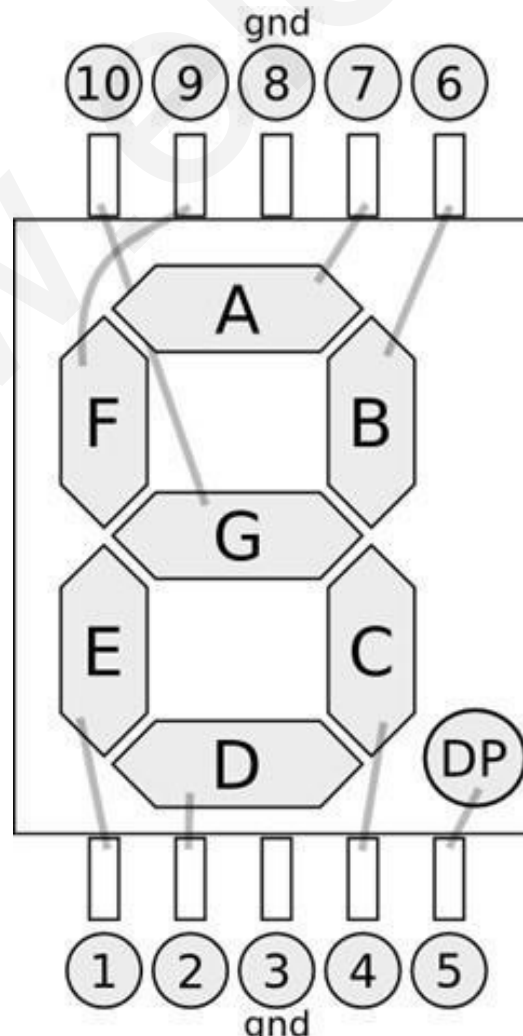
- (1) x Elegoo Uno R3
- (1) x 830 puncte de legare placă de pâine
- (1) x 74HC595 IC
- (1) x afișaj cu 1 cifră și 7 segmente
- (8) rezistențe x 220 ohm
- (26) x fire M-M (fire de conectare)



Introducere în dosar

Afișaj cu șapte segmente

Mai jos este o diagramă cu șapte segmente a pinilor.



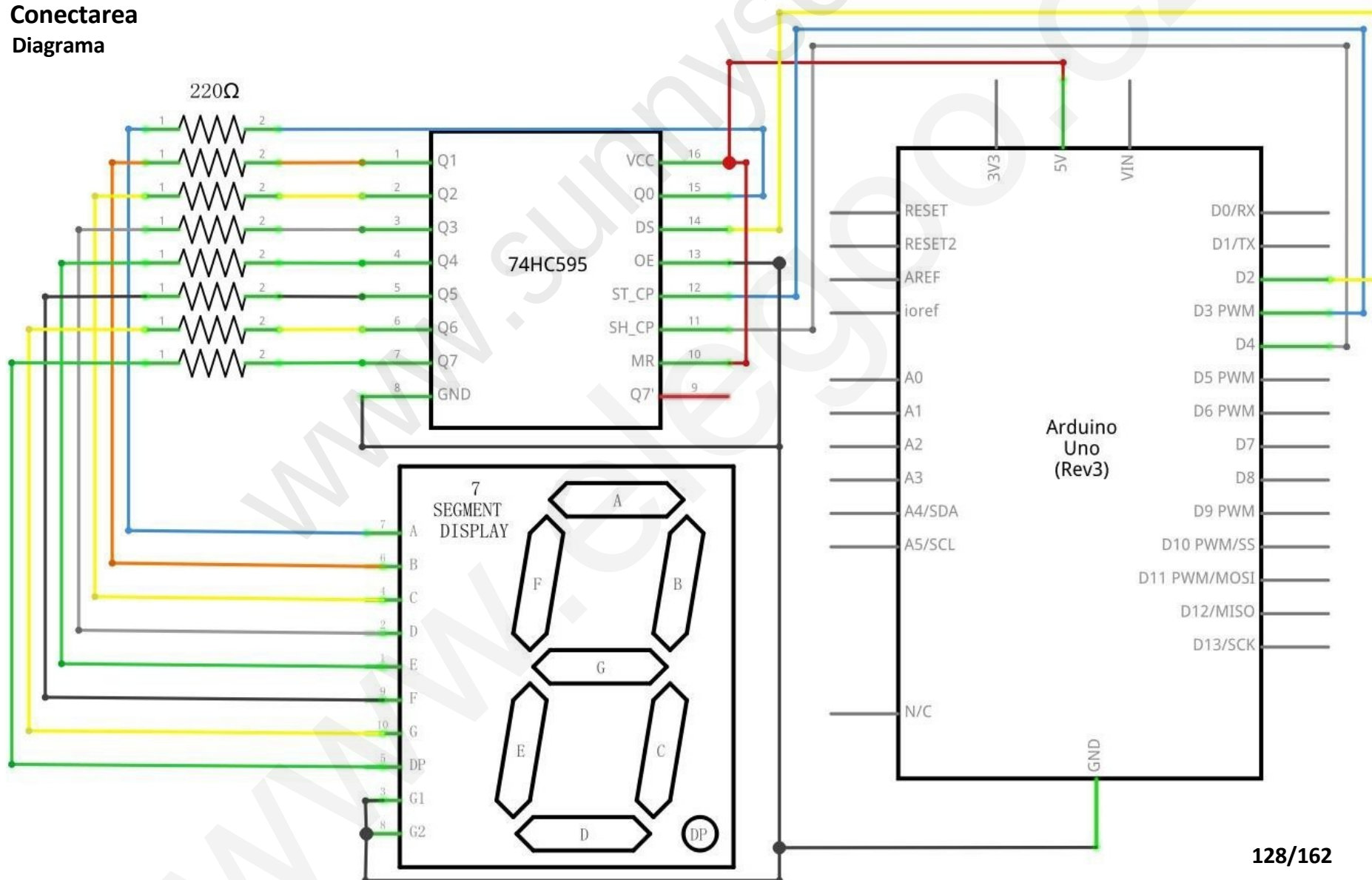
126/162

0-9 zece cifre corespund fiecărui segment după cum urmează (următorul tabel se aplică dispozitivelor de imagistică cu șapte segmente cu un catod comun, dacă utilizați un anod comun, tabelul trebuie înlocuit cu fiecare 1 0 0 toate trebuie înlocuite cu 1):

Afișaj digital	dp	a	b	c	d	e	f	g
0	0	1	1	1	1	1	1	0
1	0	0	1	1	0	0	0	0
2	0	1	1	0	1	1	0	1
3	0	1	1	1	1	0	0	1
4	0	0	1	1	0	0	1	1
5	0	1	0	1	1	0	1	1
6	0	1	0	1	1	1	1	1
7	0	1	1	1	0	0	0	0
8	0	1	1	1	1	1	1	1
9	0	1	1	1	1	0	1	1

127/162

Conectarea Diagrama



128/162

Diagram
a de
cablare

The diagram illustrates the connection of an Arduino Uno R3 to a breadboard circuit. The breadboard contains a 74HC595 shift register, six 220-ohm resistors, and a 7-segment display. The wiring is as follows:

- Power and Ground:** The 5V pin of the Arduino is connected to the positive rail of the breadboard. The GND pin is connected to the negative rail. The 3.3V pin is connected to the positive rail of the breadboard.
- Shift Register (74HC595):** The data pin (pin 14) is connected to Arduino digital pin 2. The clock pin (pin 16) is connected to Arduino digital pin 3. The master/slave select pin (pin 15) is connected to Arduino digital pin 4.
- Resistors:** Six 220-ohm resistors are used to connect the shift register's outputs (pins 1, 2, 3, 4, 5, 6) to the segments of the 7-segment display.
- 7-segment Display:** The display is connected to the breadboard rails. The common anode is connected to the positive rail, and the segments are connected to the resistors.

Tabelul următor prezintă tabelul afișajului cu șapte segmente 74HC595:

leșire 74HC595	Șapte spectacole știft de control remarcabil (cursă)
Q0	7 (A)
Q1	6 (B)
Q2	4 (C)
Q3	2 (D)
Q4	1 (E)
Q5	9 (F)
Q6	10 (G)
Q7	5 (DP)

Primul pas: Conectați 74HC595

În primul rând, cablajul este conectat la sursa de alimentare și la masă:

VCC (pin 16) și **MR** (pin 10) conectate la 5V **GND**

(pin 8) și **OE** (pin 13) la masă Conectarea pinilor

DS, **ST_CP** și **SH_CP**:

DS (pin 14) conectat la pinul 2 al plăcii UNO R3 (imaginea de sub linia galbenă)

ST_CP (pin 12, pin de blocare) conectat la pinul 3 al plăcii UNO R3 (imaginea de

sub linia albastră) **SH_CP** (pin 11, pin de ceas) conectat la pinul 4 al plăcii UNO R3 (imaginea de sub linia albă).

Pasul doi: Conectați afișajul cu șapte segmente

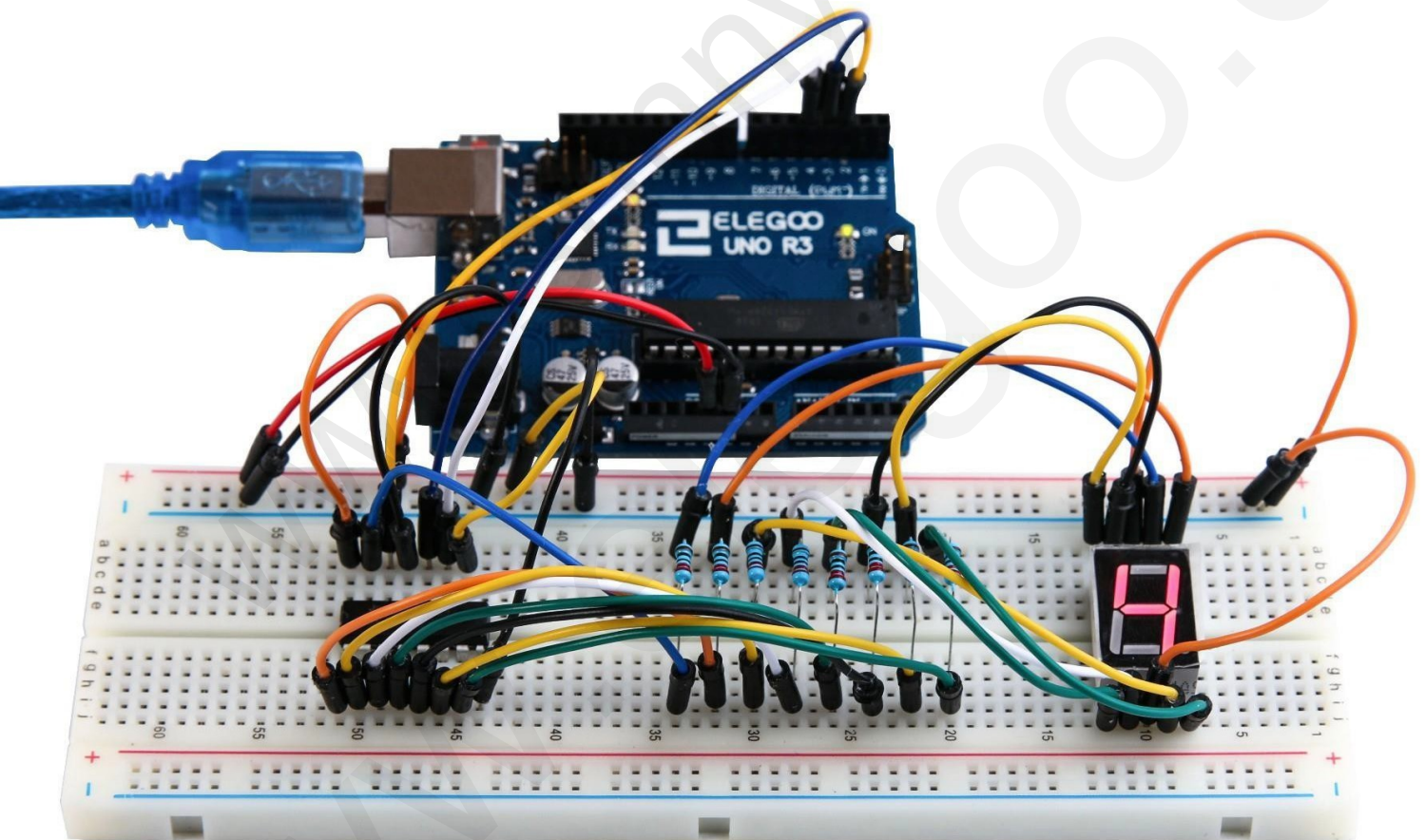
Afișaj cu șapte segmente 3, 8 pin la placa UNO R3 GND (Acest exemplu utilizează un catod comun, dacă utilizați un anod comun, conectați 3, 8 pin la placa UNO R3 + 5V).

În conformitate cu tabelul de mai sus, conectați circuitele 74HC595 Q0~Q7 la pinii corespunzători ai afișajului cu șapte segmente (A~G și DP), apoi conectați fiecare picior în serie la o rezistență de 220 ohm.

Cod

După conectare, deschideți programul din folderul cu codul - Lecția 19 74HC595 și afișajul segmentului și faceți clic pe UPLOAD pentru a încărca programul. Consultați Lecția 2 pentru detalii privind încărcarea programului dacă apar erori.

Exemplu de imagine



Lecția 20 Patru afișaje digitale cu șapte segmente

Prezentare generală

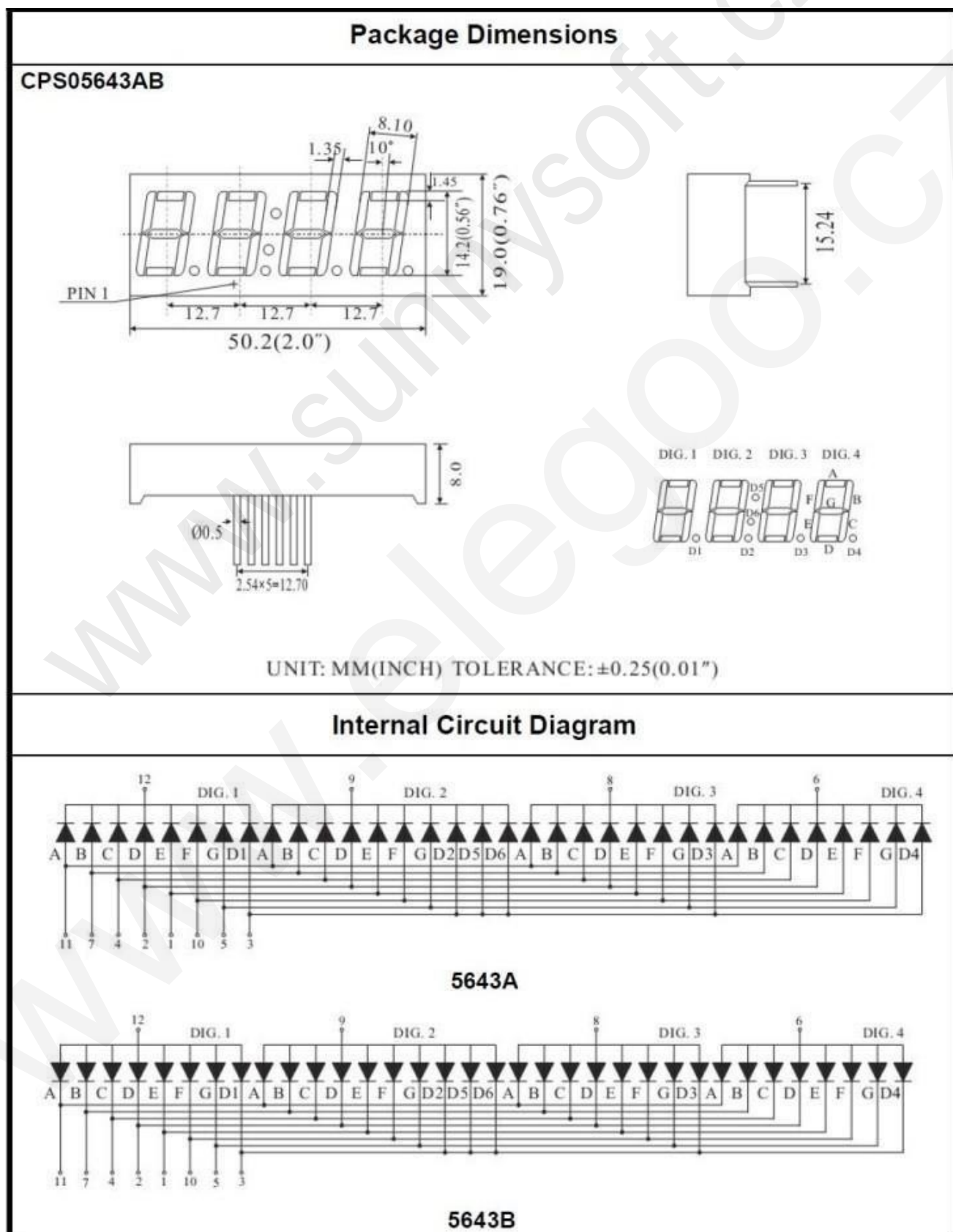
În această lecție, veți învăța cum să utilizați afișajul cu patru cifre și șapte segmente. Atunci când utilizați un afișaj de 7 segmente cu 1 cifră, rețineți că dacă este un anod comun, pinul anodului comun se conectează la sursa de alimentare; dacă este un catod comun, pinul catodului comun se conectează la GND. Atunci când utilizați un afișaj de 7 segmente cu 4 cifre, pinul anodului sau catodului comun este utilizat pentru a controla cifra afișată. Deși numai un cifre, principiul persistenței vederii vă permite să vedeți toate numerele afișate deoarece rata de scanare a fiecăruia este atât de mare încât abia observați intervalele.

Componentă necesară:

- (1) x Elegoo Uno R3
- (1) x 830 puncte de legare placă de pâine
- (1) x 74HC595 IC
- (1) x afișaj cu 4 cifre și 7 segmente
- (4) rezistențe x 220 ohm
- (23) x fire M-M (fire de interconectare mascul la mascul)



Introducere în dosar
Patru afișaje digitale cu șapte segmente



Conectarea Diagrama

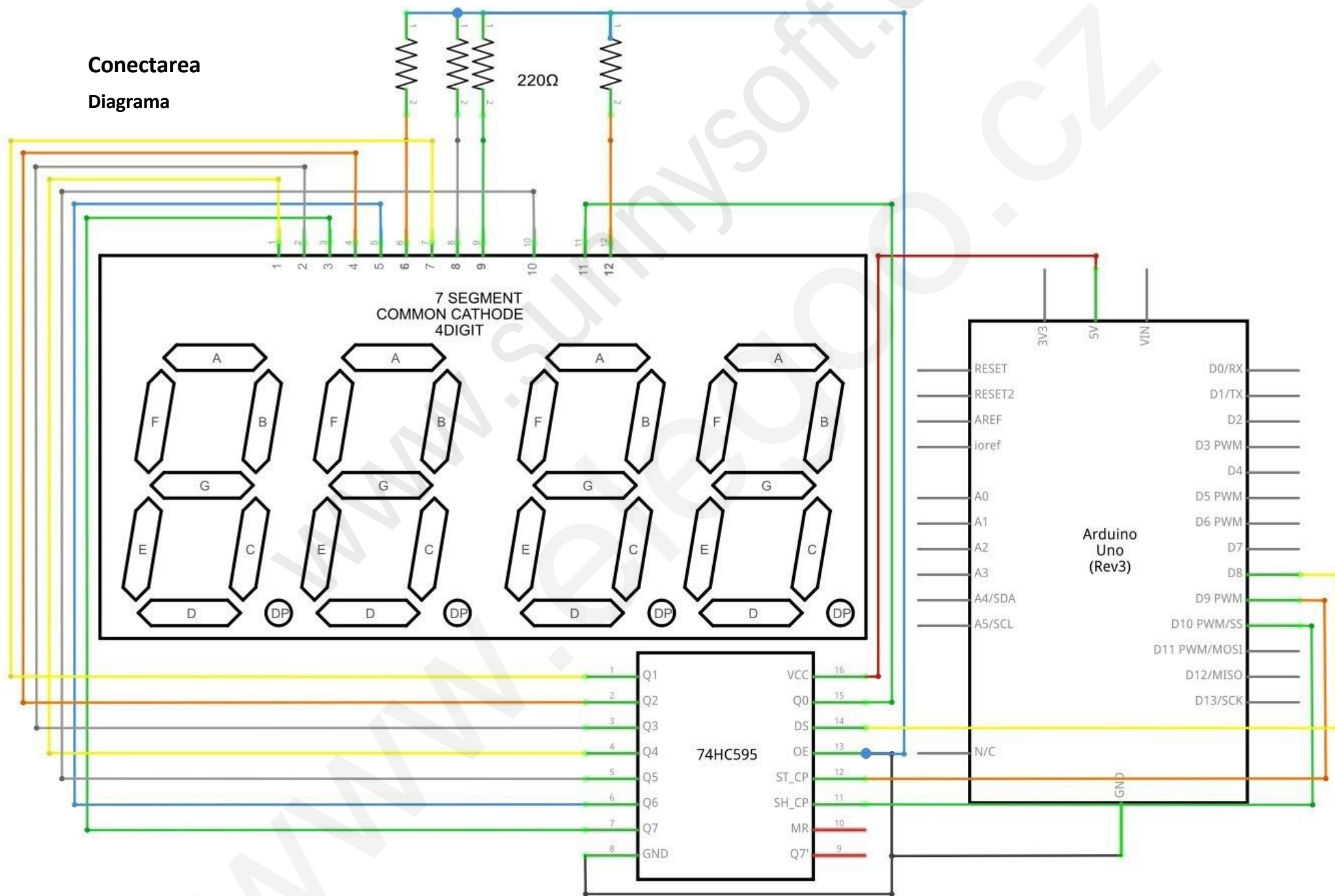
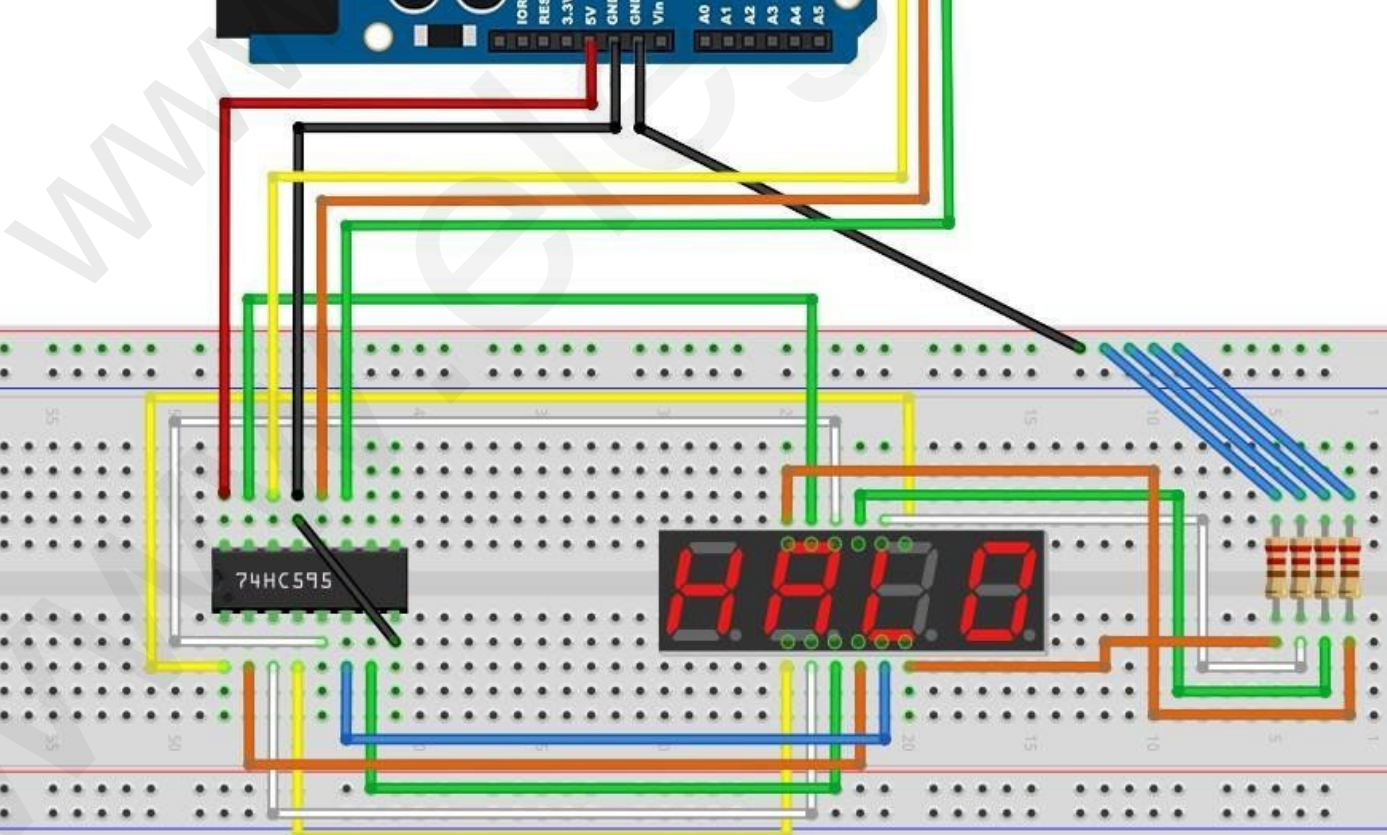


Diagrama de cablare

Diagrama de cablare arată un Arduino Uno R3 conectat la un display LCD 16x2. Cablurile sunt conectate după cum urmează:

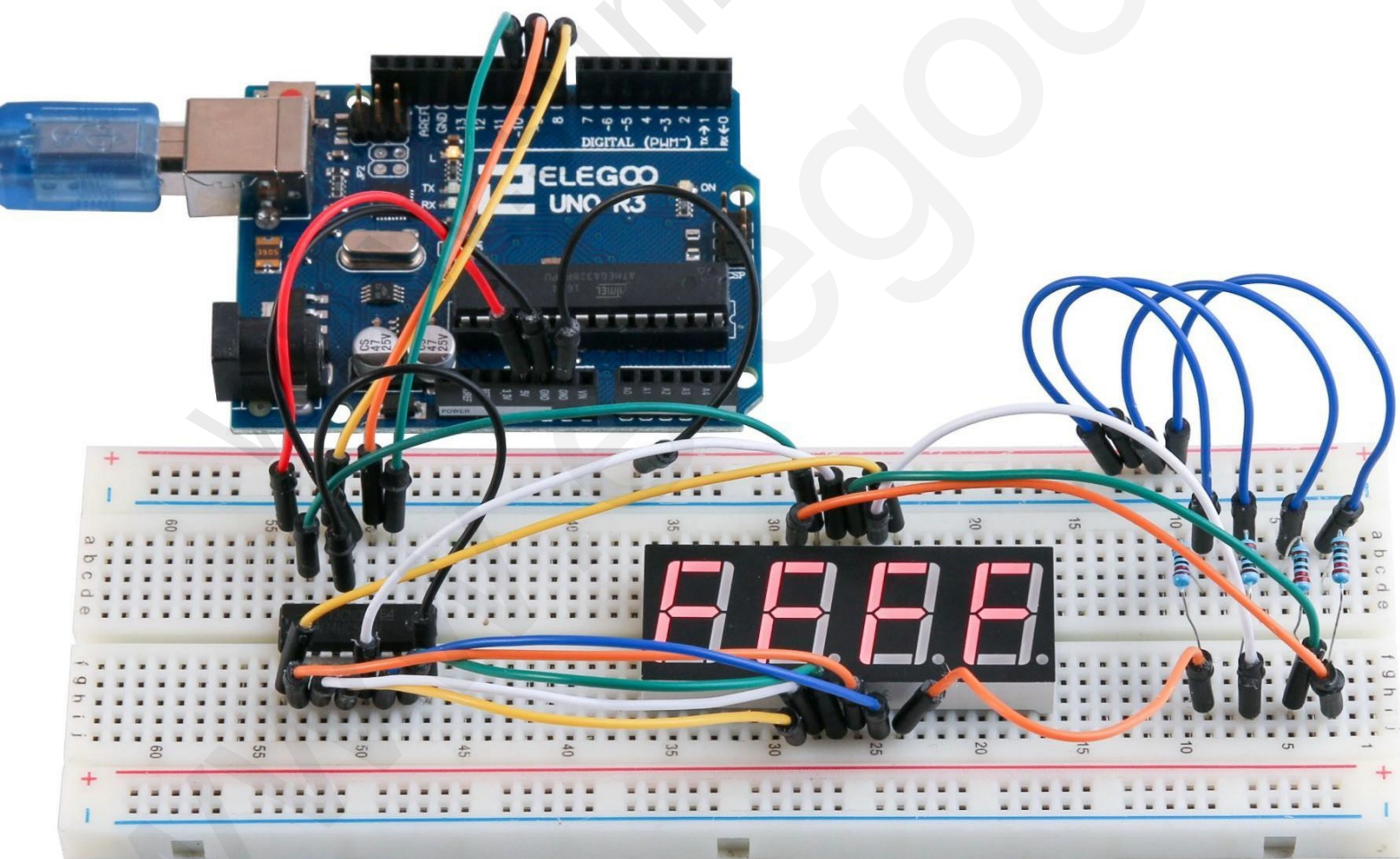
- Alimentare:**
 - Cablul roșu (VCC) este conectat la pinul 5V al Arduino.
 - Cablul negru (GND) este conectat la pinul GND al Arduino.
 - Cablul galben (VCC) este conectat la pinul VCC al display.
 - Cablul verde (GND) este conectat la pinul GND al display.
- Date:**
 - Cablul albastru (D0) este conectat la pinul TX-1 al Arduino.
 - Cablul maro (D1) este conectat la pinul RX-0 al Arduino.



Cod

După conectare, deschideți programul din folderul de coduri - Lecția 20 Patru afișaje digitale cu șapte segmente și faceți clic pe butonul UPLOAD pentru a încărca programul. Consultați Lecția 2 pentru detalii privind încărcarea programului dacă apar erori.

Exemplu de imagine



Lecția 21 Motoare de curent continuu

Prezentare generală

În această lecție veți învăța cum să controlați un mic motor de curent continuu folosind UNO R3 și un tranzistor.

Componentă necesară:

- (1) x Elegoo Uno R3
- (1) x 830 puncte de legare placă de pâine
- (1) x circuit integrat L293D
- (1) x lamă ventilator și motor 3-6V
- (5) x fire M-M (fire de conectare mascul la mascul)
- (1) x modul de alimentare
- (1) x adaptor 9V1A

Introducere în dosar

Alimentare Breadboard

Un motor DC mic va consuma probabil mai multă energie decât poate suporta ieșirea digitală a plăcii UNO R3.

direct. Dacă am încerca să conectăm motorul direct la pinul plăcii UNO R3, există o mare probabilitate ca placa UNO R3 să fie deteriorată. Prin urmare, folosim un modul de alimentare care furnizează energie la



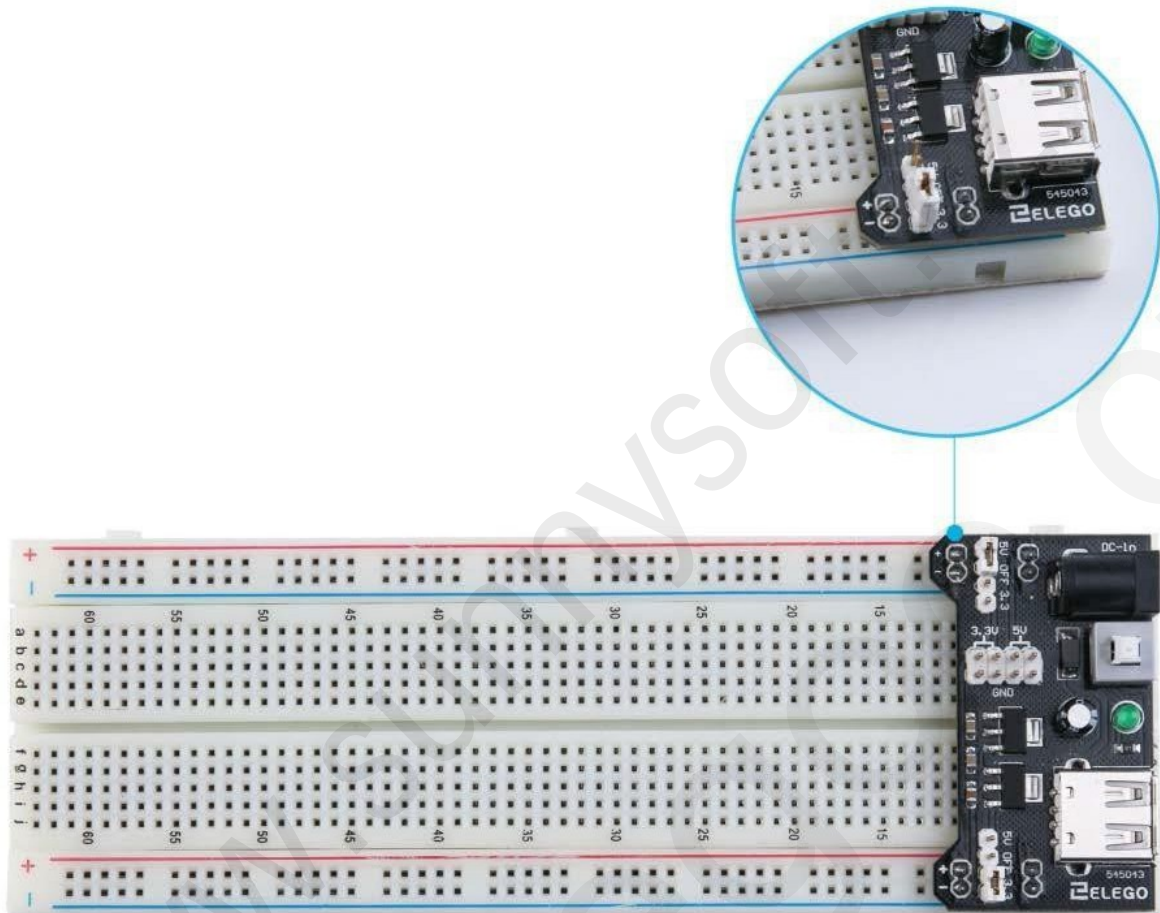
Specificațiile produsului:

- Comutator on/off blocabil
- LED indicator de alimentare
- Tensiune de intrare: 6,5-9 V (DC) prin mufa de 5,5 mm x 2,1 mm
- Tensiune de ieșire: 3,3 V/5 V
- Curent maxim de ieșire: 700 mA
- Ieșire independentă a barei de control. 0 V, 3,3 V, 5 V pentru fiecare placă de circuite
- Pini de ieșire pentru utilizare externă convenabilă
- Dimensiune: 2.1 x 1.4 inch
- Conector USB pe placă pentru alimentarea dispozitivelor externe

Setarea tensiunii de ieșire:



Ieșirile de tensiune din stânga și din dreapta pot fi configurate independent. Pentru a selecta tensiunea de ieșire, mutați jumperul la pinii corespunzători. Notă: LED-ul indicator de alimentare și șinele de alimentare de pe placă nu vor porni atunci când ambele jumperi sunt în poziția "OFF".



Notificare importantă:

Asigurați-vă că ați aliniat corect modulul pe placă. Aliniați pinul negativ (-) de pe modul cu linia albastră (-) de pe placă și aliniați pinul pozitiv (+) cu linia roșie (+). Nerespectarea acestei cerințe poate duce la repolarizarea accidentală a sursei de alimentare a proiectului.

L293D

Acesta este un cip foarte util. Acesta poate controla de fapt două motoare independent. În această lecție folosim doar jumătate din cip, majoritatea pinilor din partea dreaptă a cipului sunt pentru controlul celui alt motor.



Specificațiile produsului:

- Prezentarea produselor Unitrode L293 și L293D de la Texas Instruments
- Interval larg de tensiune de alimentare: 4,5 V până la 36 V
- Sursă logică de intrare separată
- Protecție internă împotriva descărcărilor electrostatice
- Oprire termică
- Intrări cu imunitate ridicată
- Funcțional similar cu SGS L293 și SGS L293D
- Curent de ieșire 1 A pe canal (600 mA pentru L293D)
- Curent de ieșire de vârf 2 A pe canal (1,2 A pentru L293D)
- Diode de clemă de ieșire pentru suprimarea rezistențelor inductive (L293D)



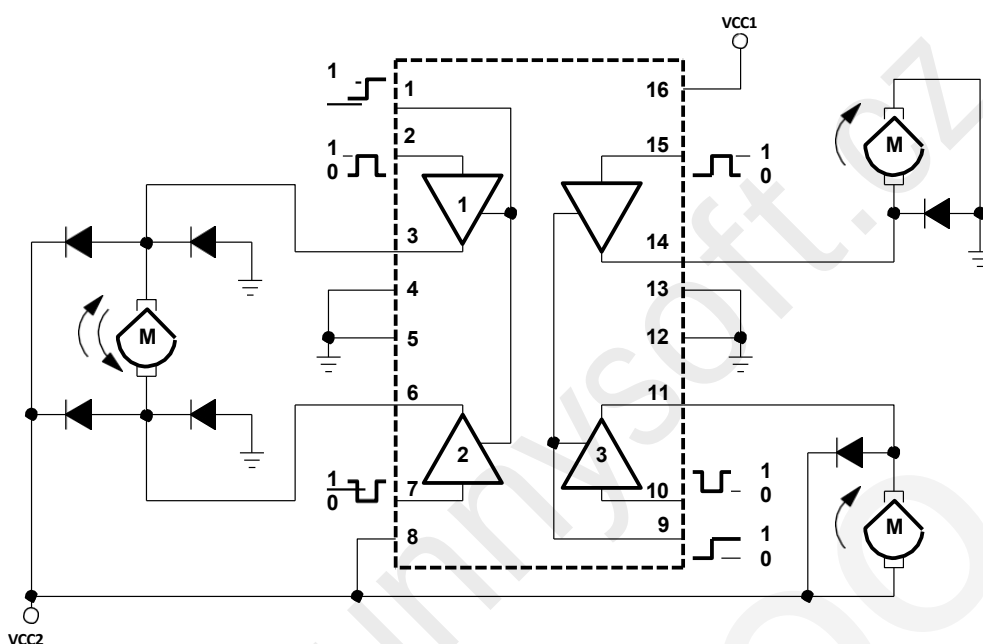
Descriere/informații de comandă

Modelele L293 și L293D sunt convertoare semi-H cu curent ridicat. Modelul L293 este proiectat pentru a furniza un curent de control bidirecțional de până la 1 A la tensiuni de la 4,5 V la 36 V. Modelul L293D este proiectat pentru a furniza un curent de control bidirecțional de până la 600 mA la tensiuni de la 4,5 V la 36 V. Ambele dispozitive sunt concepute pentru a controla sarcini inductive, cum ar fi relee, solenoizi, motoare pas cu pas de curent continuu și bipolare, precum și alte sarcini de curent mare/tensiune mare în aplicații de putere pozitivă.

Toate intrările sunt compatibile TTL. Fiecare ieșire este un circuit totem complet cu tranzistor Darlington și sursă pseudo-Darlington. Driverile sunt activate în perechi, cu driverile 1 și 2 activate cu 1,2EN și driverile 3 și 4 activate cu 3,4EN. Când intrarea de activare este ridicată, driverile respective sunt activate și ieșirile lor sunt active și în fază cu intrările lor. Când intrarea de activare este scăzută, aceste drivere sunt dezactivate, iar ieșirile lor sunt oprite și într-o stare de impedanță ridicată. Cu intrările de date corecte, fiecare pereche de drivere formează o comandă full-H (sau punte) reversibilă potrivită pentru aplicații electromagnetice sau cu motor.

Diagrama bloc

www.sunnyssoft.cz
www.elegoo.cz



M-am săturat de diagramele indescifrabile ale pinilor din fișele tehnice, așa că mi-am proiectat propria diagramă, care cred că oferă informații mai relevante.

Există 3 fire conectate la Arduino, 2 fire la motor și 1 fir la baterie.

Notă: Pinul 16 trebuie să fie activat pentru ca L293D să funcționeze normal.

L293D					
M1 PWM	1	16	Battery +ve		
M1 direction 0/1	2	15	M2 direction 0/1		
M1 +ve	3	14	M2 +ve		
GND	4	13	GND		
GND	5	12	GND		
M1 -ve	6	11	M2 -ve		
M1 direction 1/0	7	10	M2 direction 1/0		
Battery +ve	8	9	M2 PWM		
Motor 1			Motor 2		

Folosind această priză:

Partea stângă se ocupă de primul motor, partea dreaptă se ocupă de al doilea motor. Da, îl puteți utiliza cu un singur motor conectat.

Conexiune Arduino

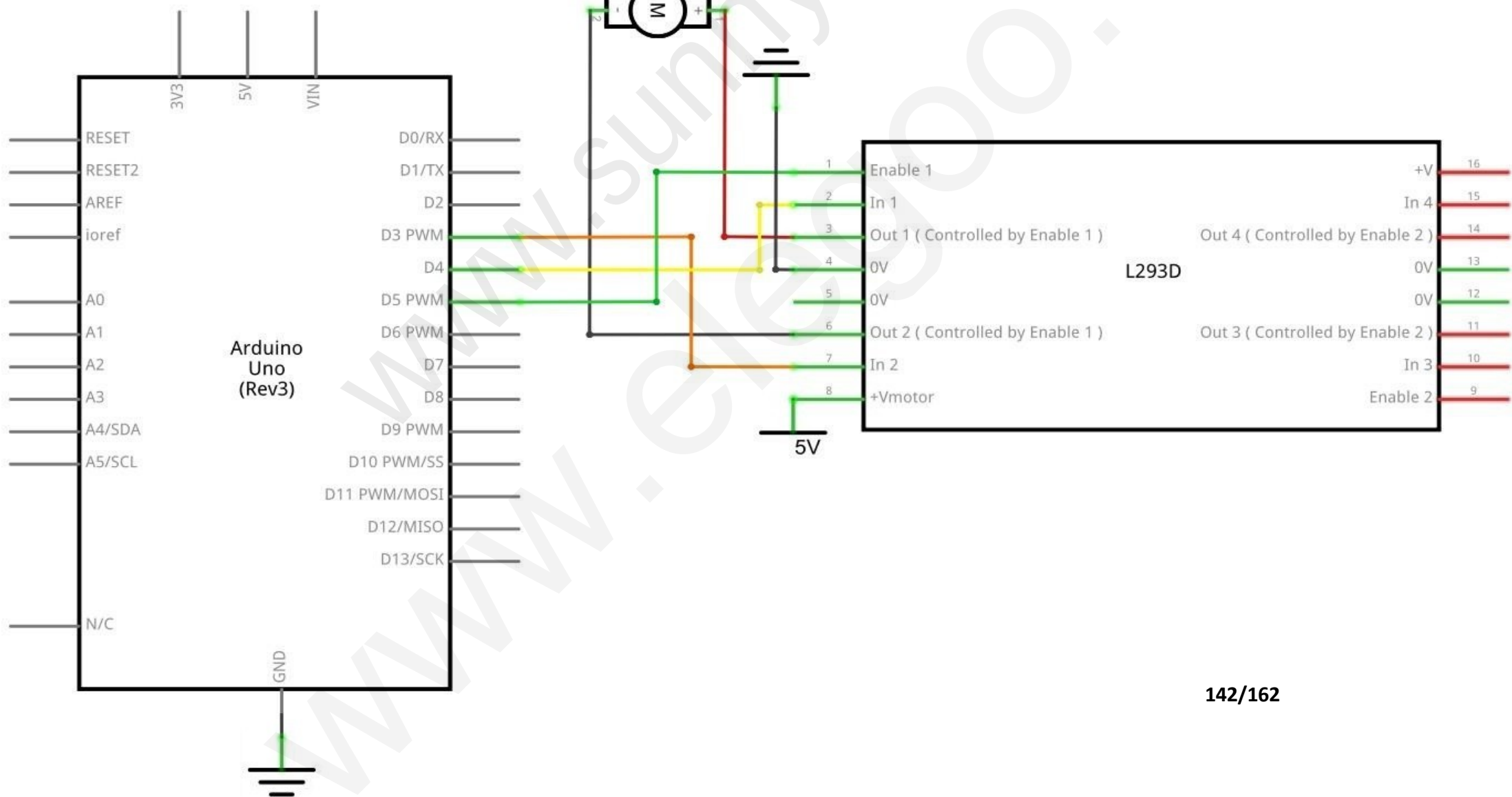
M1 PWM - conectați-l la pinul PWM de pe Arduino. Pe Uno, acestea sunt marcate, de exemplu pinul 5. Scrieți orice număr întreg între 0 și 255, unde 0 este oprit, 128 este jumătate de viteză și 255 este

viteza maximă.

M1 direcția 0/1 și M1 direcția 1/0 - Conectați-le la cei doi pini digitali ai Arduino. Scoateți un pin ca HIGH și celălalt ca LOW și motorul se va roti într-o direcție.

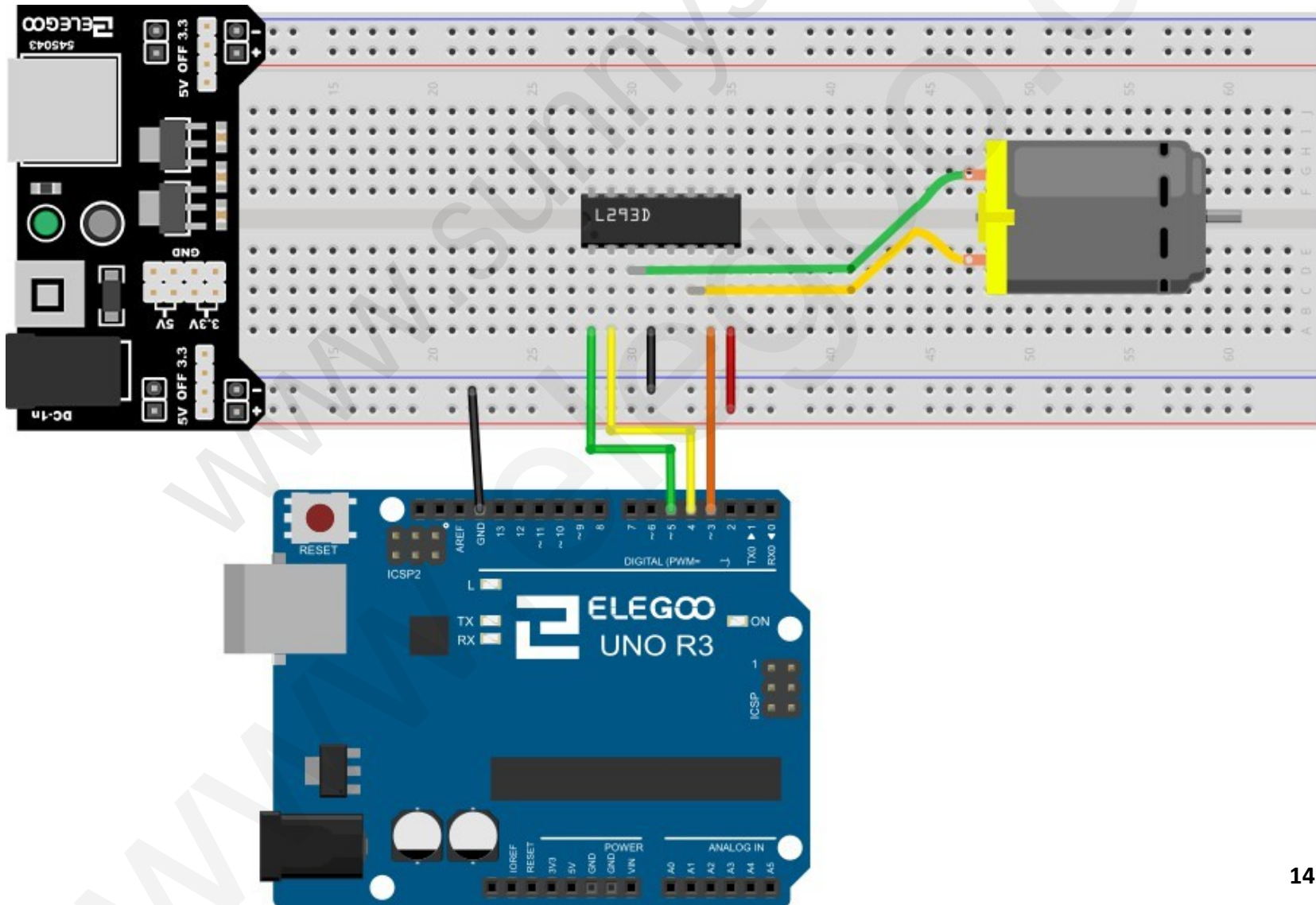
Inversați ieșirile la LOW și HIGH și motorul se va roti în direcția opusă.

Conectarea Diagrama

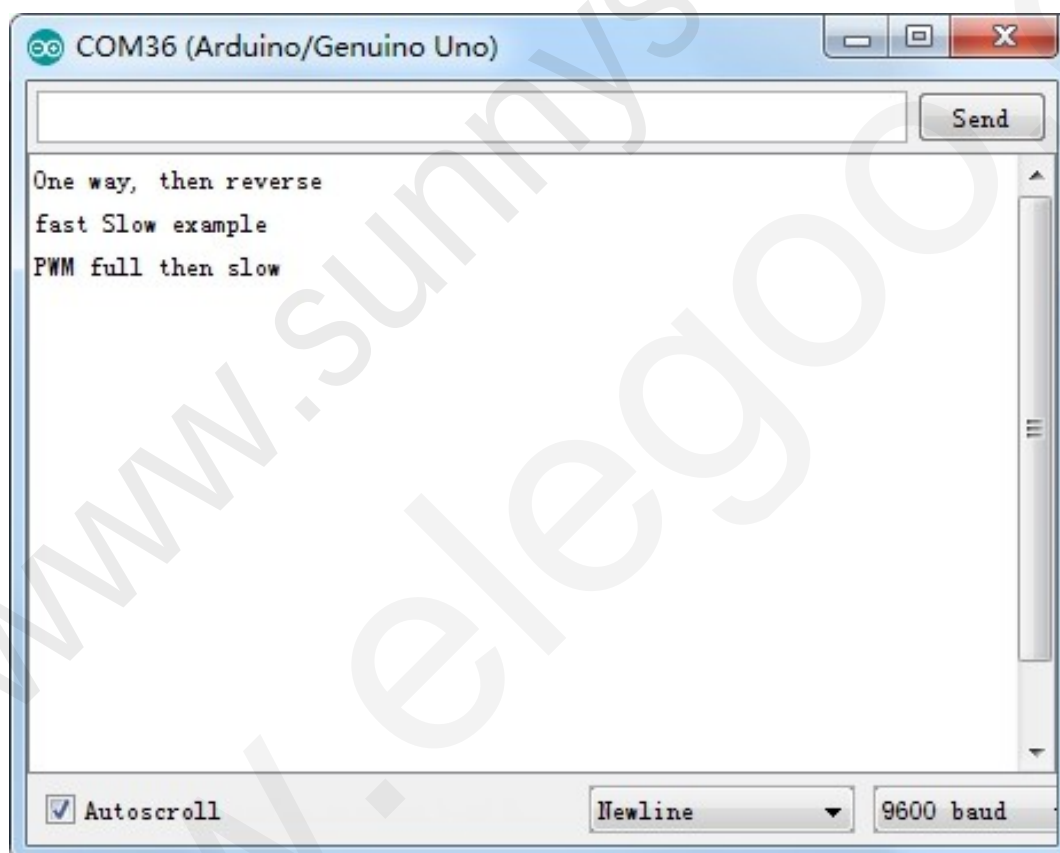


142/162

Diagrama de cablare



Nu trebuie să conectați niciodată motorul direct la Arduino, deoarece veți obține feedback electric atunci când motorul este oprit. Cu un motor mic, acest lucru va deteriora Arduino, iar cu un motor mare puteți vedea un efect interesant de flăcări și scântei.



Cod

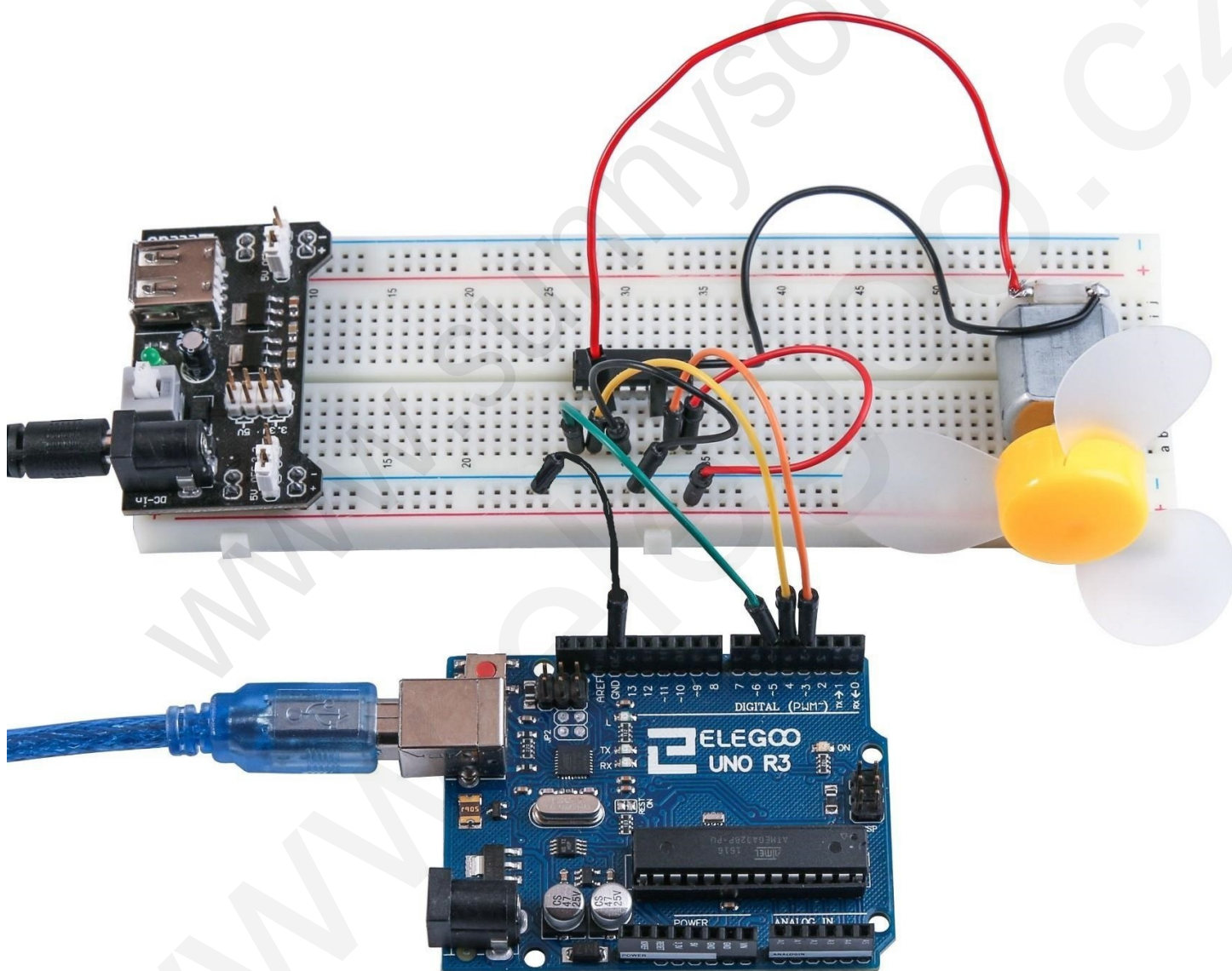
După conectare, deschideți programul din folderul cu codul - [Lecția 21 Motoare de curent continuu](#) și faceți clic pe butonul **UPLOAD** pentru a încărca programul. Consultați [Lecția 2](#) pentru detalii privind încărcarea programului dacă există erori.

După încărcarea programului, porniți toate comutatoarele. Motorul se va roti ușor în sensul acelor de ceasornic și în sens invers acelor de ceasornic de cinci ori. Apoi va continua să se rotească rapid în sensul acelor de ceasornic. După o scurtă pauză, se va roti dramatic în sens invers acelor de ceasornic. Apoi placa de control va trimite un semnal PWM pentru a acționa motorul, motorul își va reduce încet viteza maximă la un minim și o va crește din nou la un maxim. În cele din urmă, se

va opri timp de 10 s până la începerea ciclului următor.

www.sunnysoft.cz

Exemplu de imagine



Lecția 22 Releu

Prezentare generală

În această lecție veți învăța cum să utilizați releele.

Componentă necesară:

- (1) x Elegoo Uno R3
- (1) x 830 puncte de legare placă de pâine
- (1) x lamă de ventilator și motor de 3-6 V CC
- (1) x circuit integrat L293D
- (1) x releu de 5V
- (1) x modul de alimentare
- (1) x adaptor 9V1A
- (8) x fire M-M (fire de conectare mascul la mascul)



Introducere în dosar

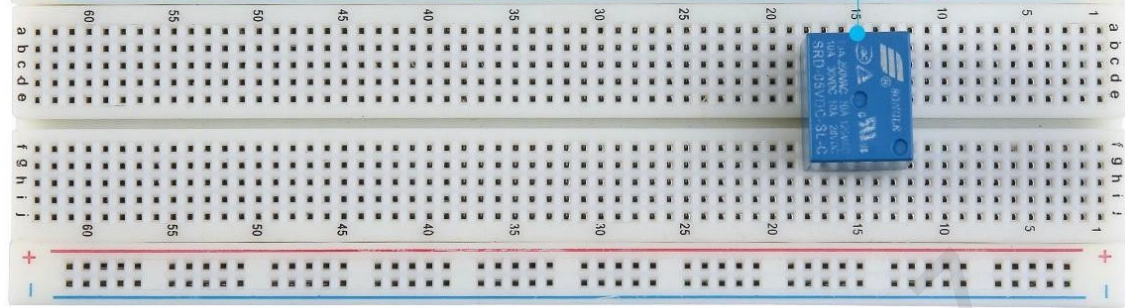
Releu:

Releul este un comutator acționat electric. Multe releee utilizează un electromagnet pentru a controla mecanic comutatorul, dar sunt utilizate și alte principii de control, cum ar fi în cazul releelor cu stare solidă. Releeele sunt utilizate atunci când este necesar să se controleze un circuit cu un semnal de mică putere (cu izolare electrică completă între circuitul de control și cel controlat) sau atunci când este necesar să se controleze mai multe circuite cu un singur semnal. Primele releee au fost utilizate în circuitele telegrafice pe distanțe lungi ca amplificatoare. Acestea repetau un semnal provenit de la un circuit și îl retransmiteau către un alt circuit. Releeele au fost utilizate pe scară largă în centralele telefonice și în primele calculatoare pentru a efectua operații logice.

Tipul de releu care poate gestiona puterea mare necesară pentru a controla direct un motor electric sau alte sarcini se numește contactor. Releeele cu stare solidă controlează circuitele electrice fără piese în mișcare, folosind în schimb un dispozitiv cu stare solidă pentru a comuta. Releeele cu caracteristici de funcționare calibrate și, uneori, cu bobine de funcționare multiple sunt utilizate pentru a proteja circuitele electrice împotriva suprasarcinilor sau defecțiunilor. În sistemele electrice moderne, aceste funcții sunt îndeplinite de dispozitive digitale denumite "releee de protecție".

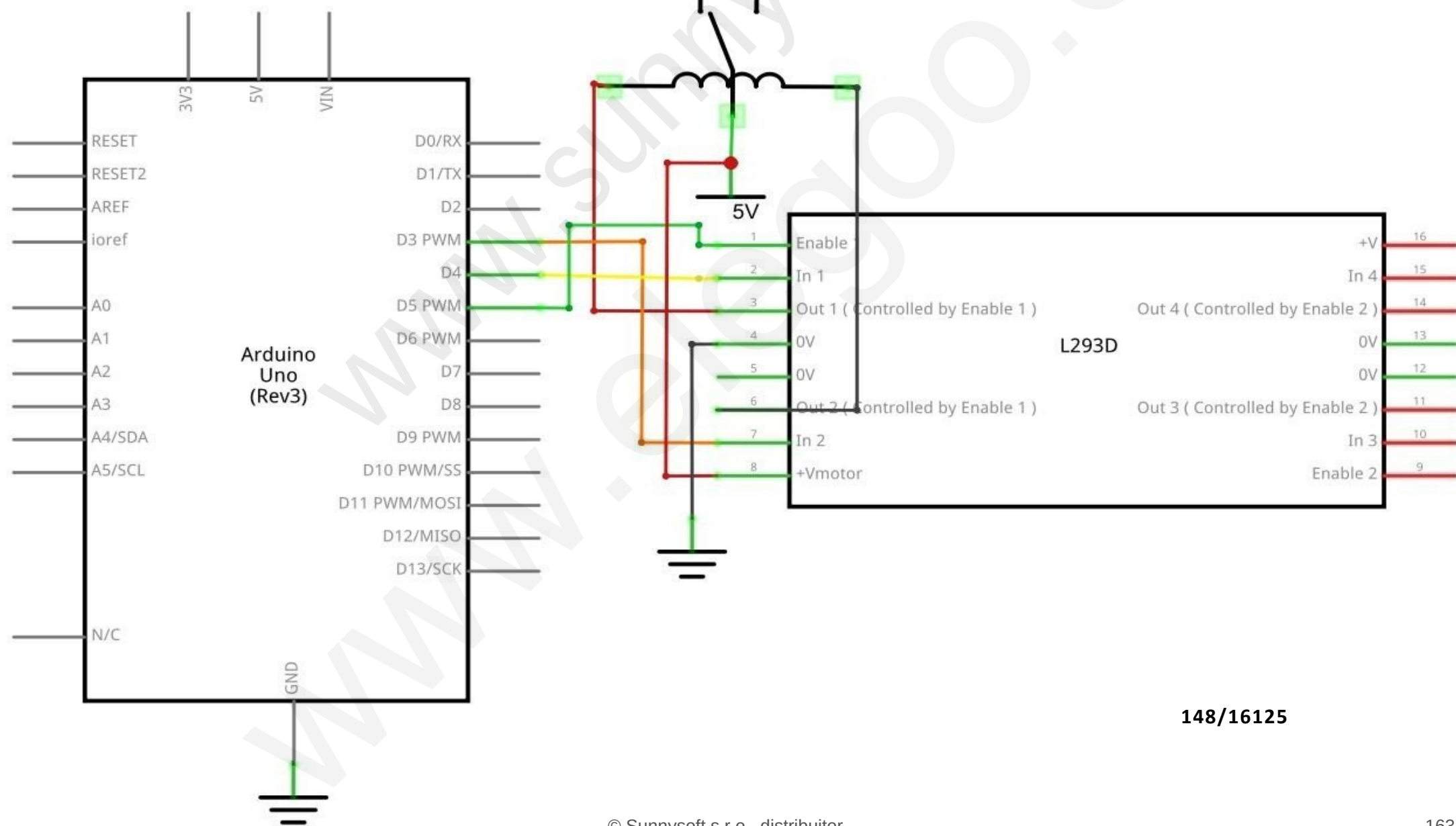
Mai jos este o diagramă a modului de control al releului folosind Arduino.

Este posibil să fiți confuz cu privire la modul de introducere a releului în placă. După cum arată imaginea de mai jos, va trebui să îndoiți ușor unul dintre pinii releului și apoi să îl introduceți în placă.



147/162

Conectarea Diagrama



148/16125

Diagrama de cablare

Diagrama de cablare prezintă următoarele componente și conexiuni:

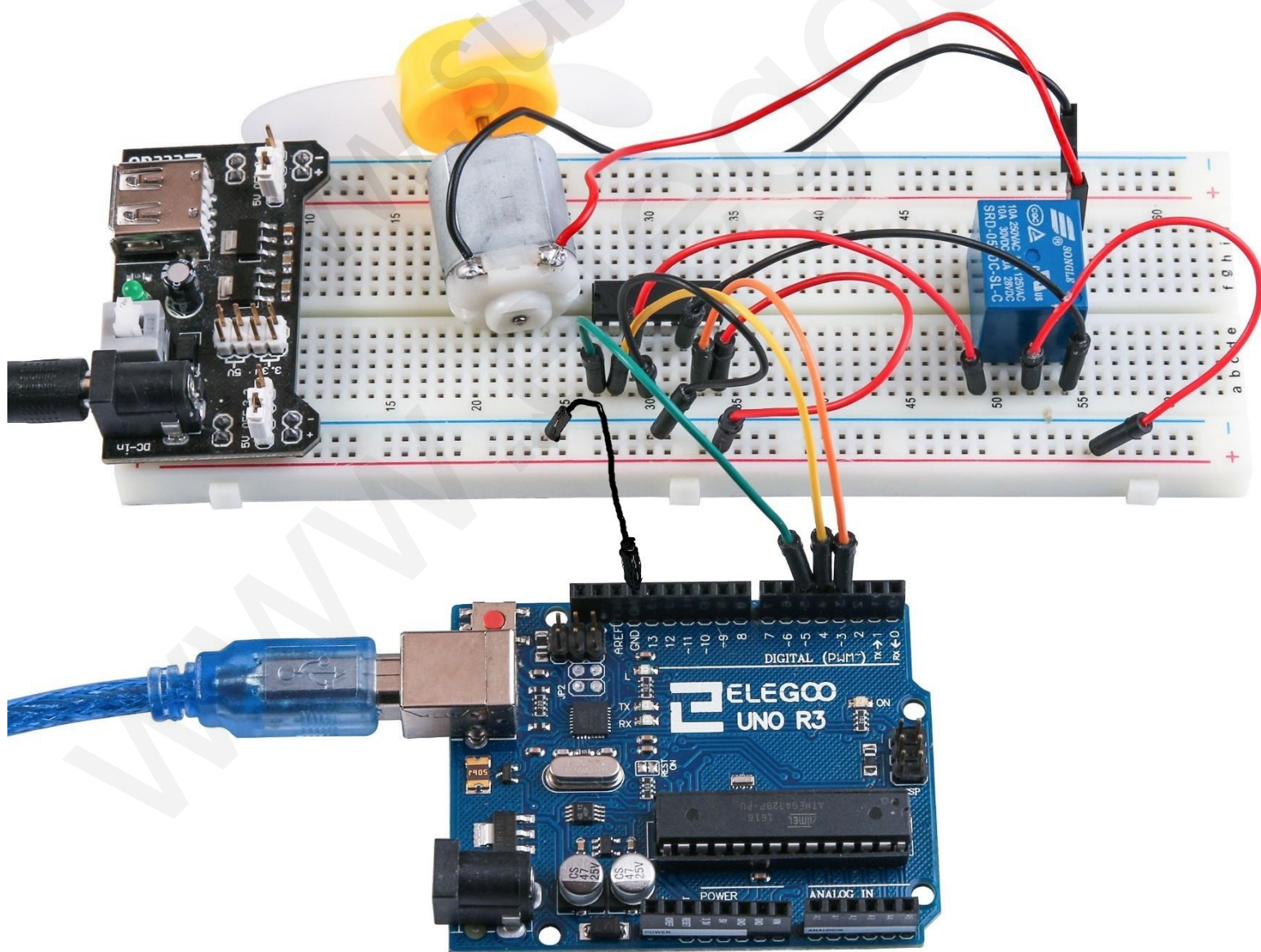
- Motor DC:** Conectat la breadboard prin cabluri galbene, verzi și roșii.
- Driver L293D:** Conectat la breadboard. Pinii de alimentare (Vcc și GND) sunt conectați la sursa de alimentare. Pinii de control (ENA, ENB) sunt conectați la pinii digitali ai Arduino (pin 2 și pin 4).
- Convertor de tensiune SRD-05VDC-SL-C:** Conectat la breadboard. Pinii de alimentare (Vin și GND) sunt conectați la sursa de alimentare. Pinii de ieșire (Vout și GND) sunt conectați la pinii de alimentare ai driverului L293D.
- Module de alimentare ELEGOO S450A3:** Conectat la breadboard. Pinii de alimentare (5V, GND, 3.3V) sunt conectați la pinii de alimentare ai driverului L293D.

Conexiunile sunt realizate pe o placă de breadboard, iar Arduino Uno R3 este conectat la breadboard prin cabluri de culoare corespunzătoare pinilor de alimentare și control.

Cod

După conectare, deschideți programul din folder cu codul - Lecția 22 Relay și faceți clic pe butonul UPLOAD pentru a încărca programul. Consultați Lecția 2 pentru detalii privind încărcarea programului dacă apar erori. După ce programul este încărcat, porniți toate comutatoarele. Releul va emite un sunet de sonerie. Apoi motorul va începe să se rotească. După un timp, releul se va elibera și motorul se va opri.

Exemplu de imagine



Lecția 23 Motor pas cu pas

Prezentare generală

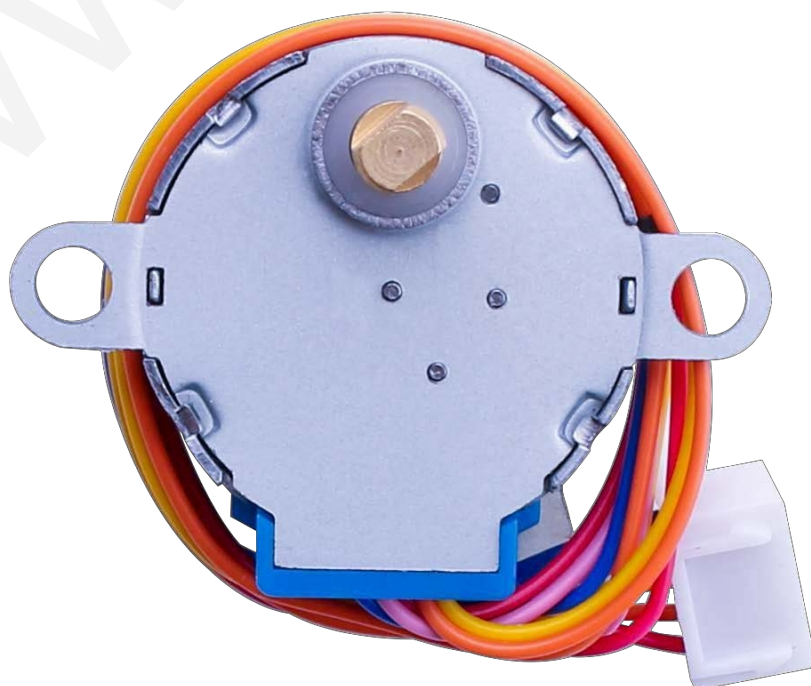
În această lecție veți învăța un mod simplu și distractiv de a controla un motor pas cu pas. Driverul pas cu pas pe care îl folosim vine cu propria sa placă de driver, astfel încât este ușor să îl conectăm la UNO.

Componentă necesară:

- (1) x Elegoo Uno R3
- (1) x 830 puncte de legare placă de pâine
- (1) x modul de comandă a motorului pas cu pas ULN2003
- (1) x motor pas cu pas
- (1) x adaptor 9V1A
- (1) x modul de alimentare
- (6) x fire F-M (fire DuPont de la femelă la mascul)
- (1) x fir M-M (fir de conectare de la mascul la mascul)

Introducere în dosar

**Motor
pas cu
pas**



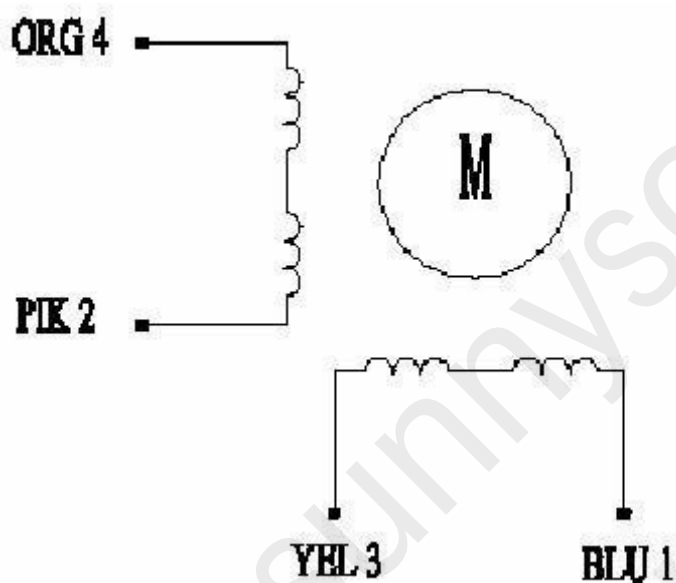
151 / 162

Un motor pas cu pas este un dispozitiv electromecanic care convertește impulsurile electrice în mișcări mecanice discrete. Arborele sau axul unui motor pas cu pas se rotește în pași discreți atunci când îi sunt aplicate impulsuri electrice de comandă în secvența corectă. Rotația motorului are mai multe relații directe cu aceste impulsuri de intrare aplicate. Ordinea impulsurilor aplicate este direct legată de direcția de rotație a arborelui motorului. Viteza de rotație a arborelui motorului este direct legată de frecvența impulsurilor de intrare, iar durata de rotație este direct legată de numărul de impulsuri de intrare aplicate. Unul dintre cele mai importante avantaje ale motorului pas cu pas este capacitatea sa de a fi controlat cu precizie într-un sistem cu buclă deschisă. Controlul în buclă deschisă înseamnă că nu este necesară nicio reacție de poziție. Acest tip de control elimină nevoia de dispozitive scumpe de detectare și feedback, cum ar fi encoderele optice. Poziția este cunoscută prin simpla urmărire a impulsurilor treptei de intrare.

Motor pas cu pas 28BYJ-48 Parametri

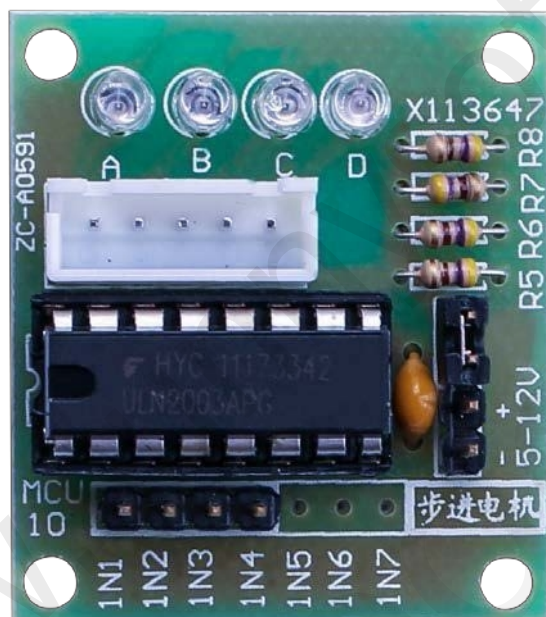
- Model: 28BYJ-48
- Tensiune nominală: 5VDC
- Număr de faze: 4
- Raportul de schimbare a vitezei: 1/64
- Unghi de pas: $5,625^\circ / 64$
- Frecvență: 100 Hz
- Rezistență DC: $50\Omega \pm 7\%$ (25 °C)
- Frecvența de inactivitate: > 600 Hz
- Frecvența de inactivitate: > 1000 Hz
- Cuplu de tracțiune >34,3 mN.m (120 Hz)
- Cuplu de autopозиționare >34,3 mN.m
- Cuplu de frecare: 600-1200 gf.cm
- Cuplu de tracțiune: 300 gf.cm
- Rezistență izolată > 10M Ω (500V)
- Putere electrică izolată: 600VAC/1mA/1s
- Grad de izolare: A
- Creșterea temperaturii <40K (120Hz)
- Nivel de zgomot <35 dB (120 Hz, fără sarcină, 10 cm)

WIRING DIAGRAM



De obicei, dintr-un motor pas cu pas bipolar ies patru fire. Spre deosebire de motoarele stepper unipolare, motoarele stepper bipolare nu au o conexiune centrală comună. În schimb, acestea au două seturi independente de bobine. Ele pot fi deosebite de motoarele pas cu pas unipolare prin măsurarea rezistenței dintre conductoare. Ar trebui să găsiți două perechi de fire cu aceeași rezistență. Dacă aveți pinii contorului conectați la două conductoare care nu sunt conectate (adică nu sunt conectate la aceeași bobină), ar trebui să vedeți o rezistență infinită (sau discontinuitate).

Placă driver ULN2003



Descrierea produsului

- Dimensiuni: 42 mm x 30 mm
- Utilizați un cip ULN2003, 500mA
- A. B. C. LED-ul D indică starea de funcționare a motorului pas cu pas cu patru faze.
- Conectorul alb este conectorul standard pentru un motor pas cu pas cu patru faze.
- Pinii de alimentare sunt separați
- Pinii rămași ai cipului ULN2003 au fost lăsați pentru prototipuri ulterioare. Cel mai simplu mod de a conecta un circuit pas cu pas unipolar la un Arduino este de a utiliza hub-ul cu matrice de tranzistori ULN2003A. ULN2003A conține șapte drivere pentru tranzistori Darlington și este un fel de a avea șapte tranzistori TIP120 într-un singur pachet. ULN2003A poate trece până la 500 mA pe canal și are o cădere de tensiune internă de aproximativ 1 V atunci când este pornit. Acesta include, de asemenea, diode de prindere interne pentru a disipa vârfurile de tensiune atunci când conduce sarcini inductive. Pentru a pune în funcțiune driverul pas cu pas,

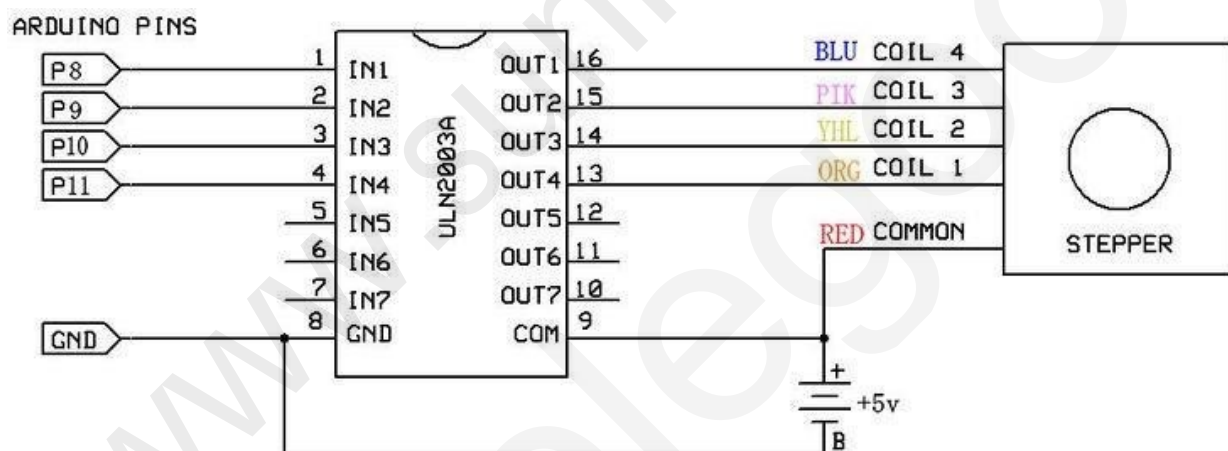
aplicați tensiune la fiecare dintre bobine în succesiune.

www.sunnysoft.cz

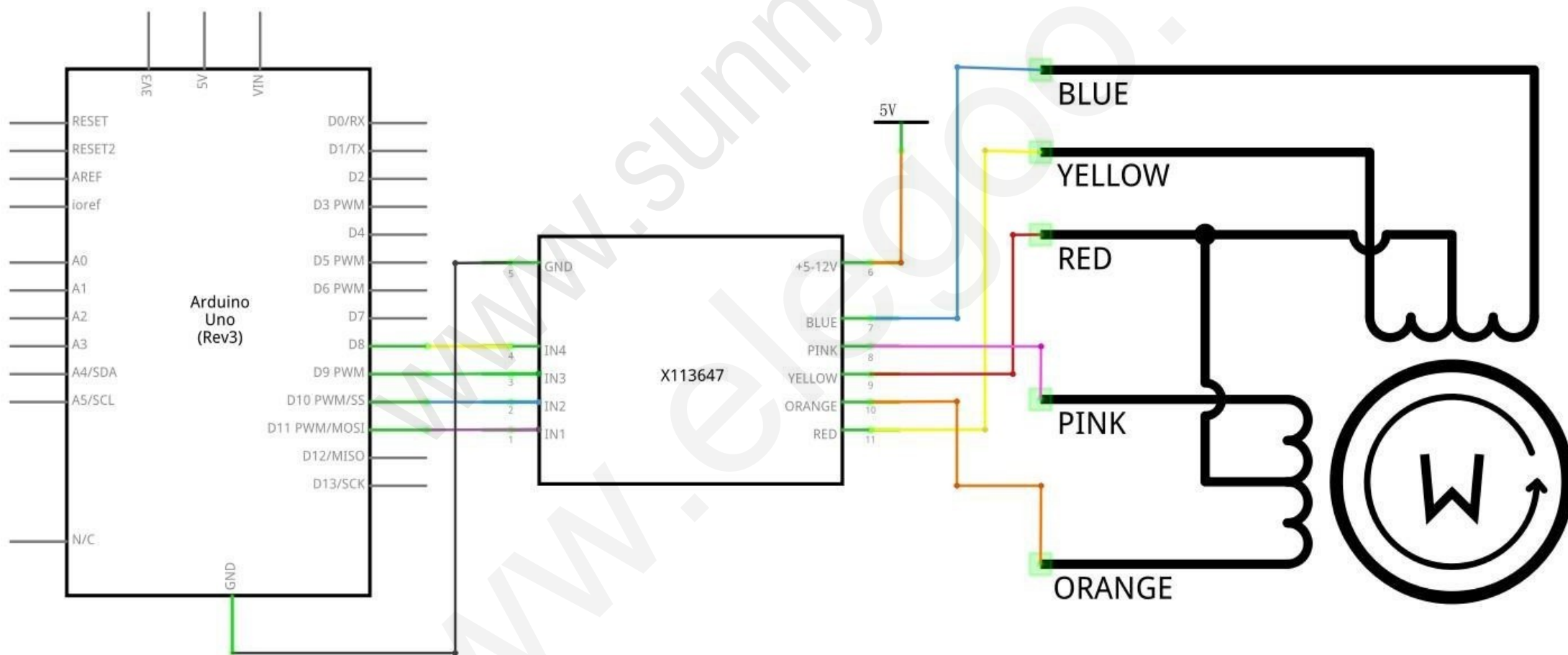
Ordinea ar fi următoarea:

Lead Wire Color	---> CW Direction (1-2 Phase)							
	1	2	3	4	5	6	7	8
4 ORG	-	-						-
3 YEL		-	-	-				
2 PIK				-	-	-		
1 BLU						-	-	-

Iată diagramele care arată cum să conectați un motor pas cu pas unipolar cu patru pini folosind ULN2003A, și să arate cum să se interconecteze folosind patru controlere comm.

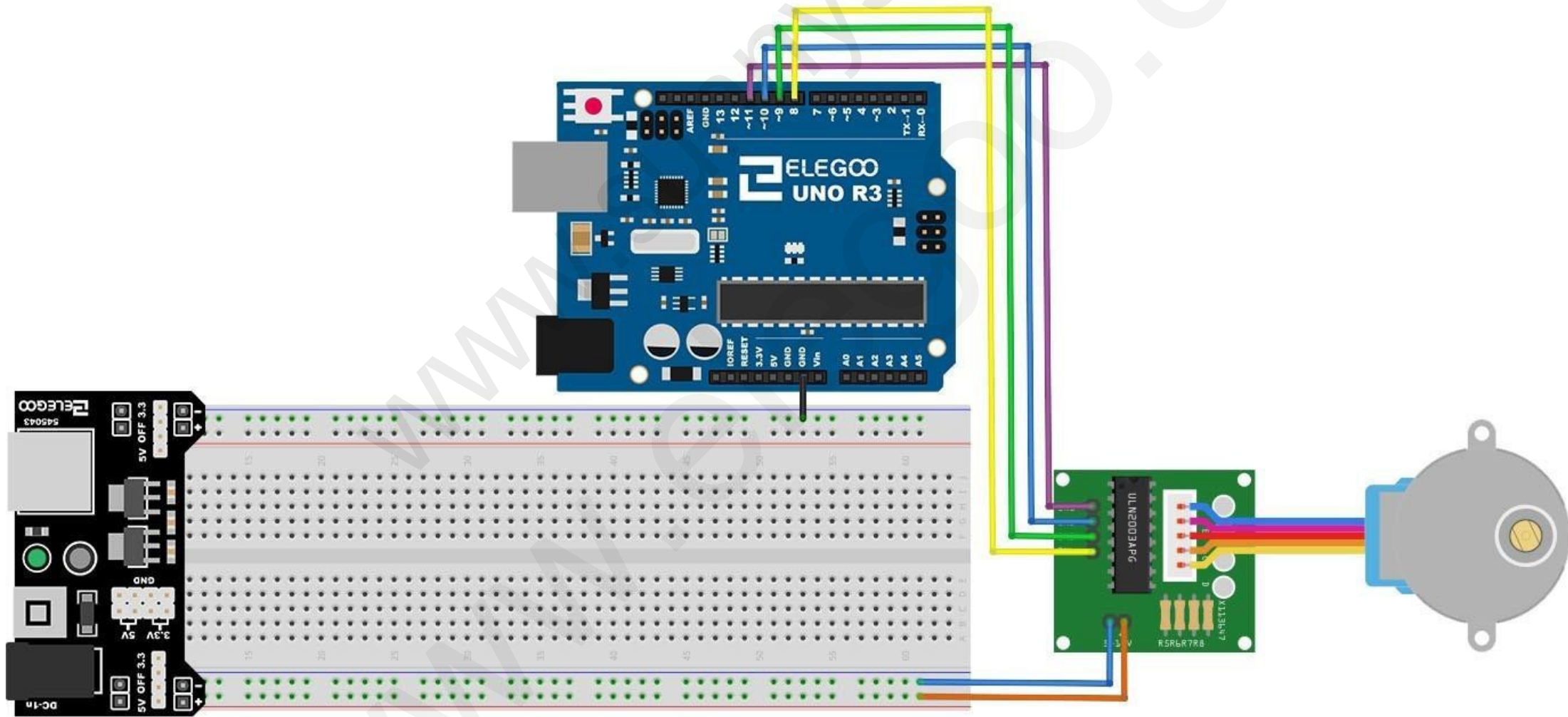


Conectarea Diagrama



156/162

Diagrama de cablare



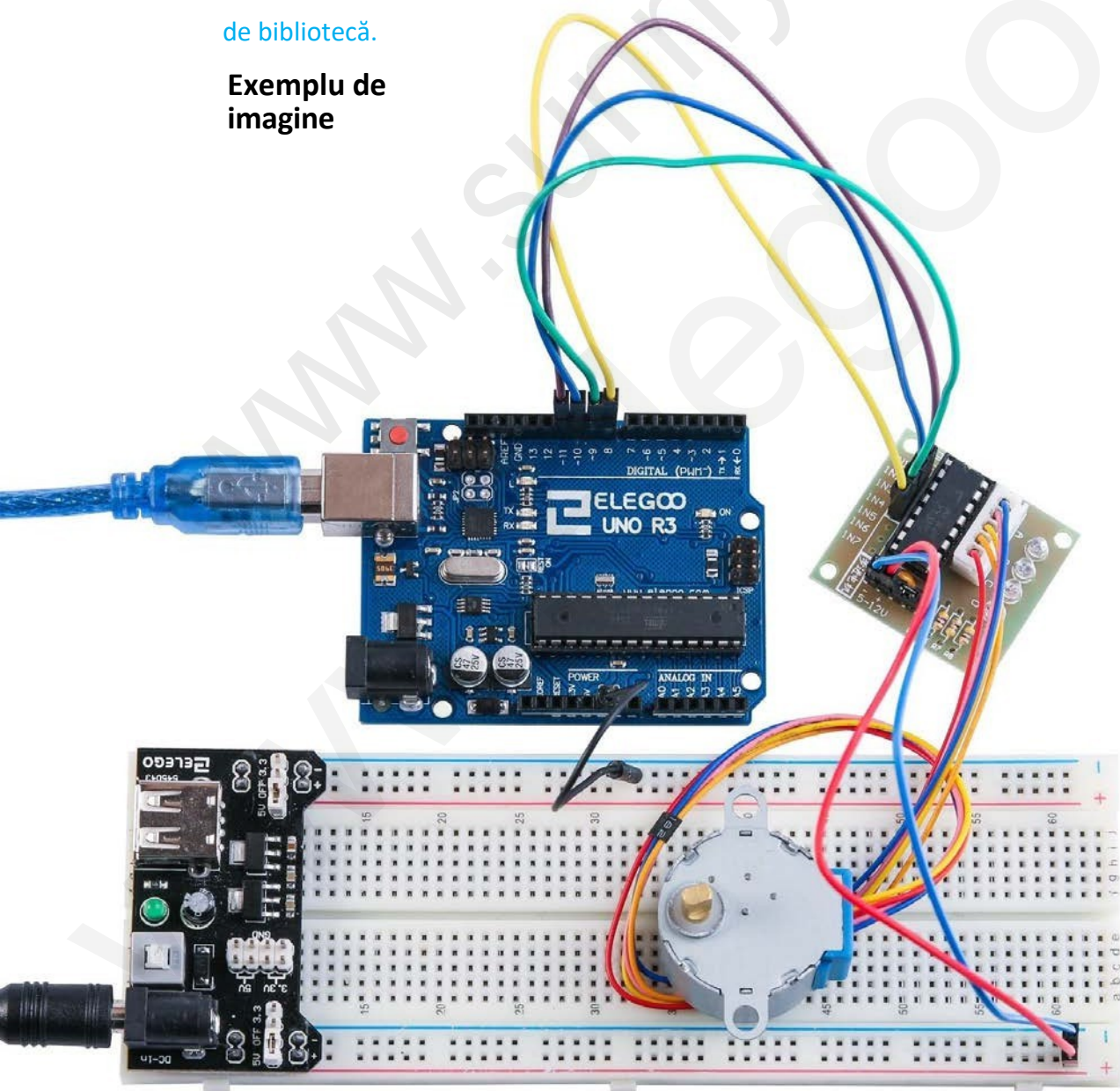
157/162

Folosim 4 pini pentru a controla driverul pas cu pas. Pinii 8-11 controlează motorul pas cu pas. Conectați împământarea de la UNO la Steppermotor.

Cod

După conectare, deschideți programul din folderul de cod - Lecția 23 Stepper Motor și faceți clic pe butonul UPLOAD pentru a încărca programul. Consultați Lecția 2 pentru detalii privind încărcarea programului dacă există erori. Înainte de a începe, asigurați-vă că aveți instalată biblioteca < Stepper >, sau reinstalați-l. În caz contrar, codul dvs. nu va funcționa. Consultați Lecția 1 pentru detalii privind încărcarea fișierului de bibliotecă.

Exemplu de imagine



Lecția 24 Controlul unui motor pas cu pas cu ajutorul unei telecomenzi

Prezentare generală

În această lecție veți învăța o modalitate simplă și distractivă de a controla un motor pas cu pas de la distanță folosind o telecomandă IR.

Driverul pas cu pas pe care îl folosim vine cu propria sa placă de driver, astfel încât este ușor să îl conectăm la UNO.

Deoarece nu dorim să acționăm motorul direct de la UNO, vom folosi o mică sursă de alimentare ieftină care se conectează direct la breadboard-ul nostru și va fi alimentată de 9V 1Amp.

Senzorul IR este conectat direct la UNO deoarece nu consumă aproape deloc energie.

Componentă necesară:

- (1) x Elegoo Uno R3
- (1) x 830 puncte de legare placă de pâine
- (1) x modul receptor IR
- (1) x telecomandă IR
- (1) x modul de comandă a motorului pas cu pas ULN2003
- (1) x motor pas cu pas
- (1) x modul de alimentare
- (1) x adaptor 9V1A
- (9) x fire F-M (fire DuPontwires de la femelă la mascul)
- (1) x fir M-M (fir de conectare de la mascul la mascul)

Conectarea Diagrama

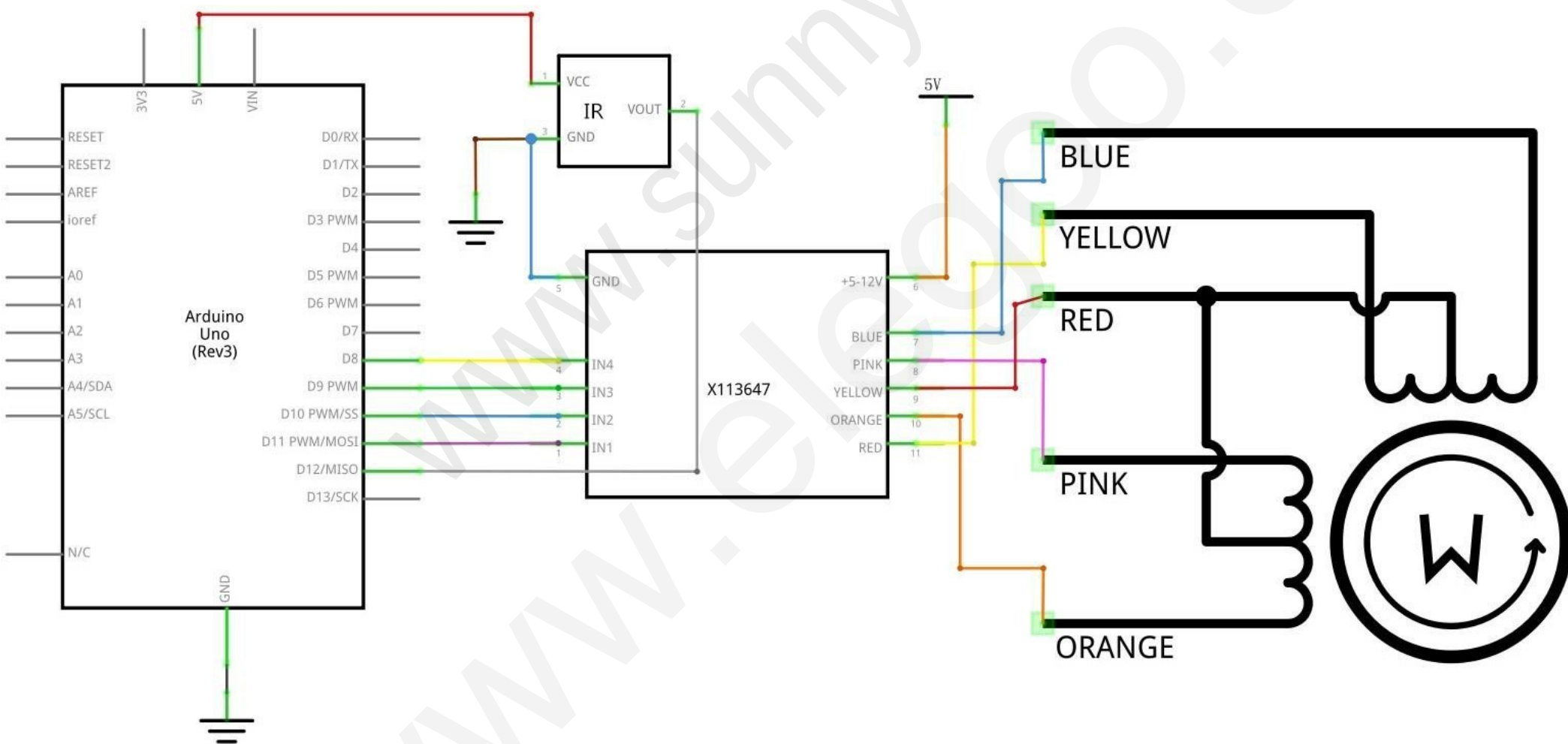
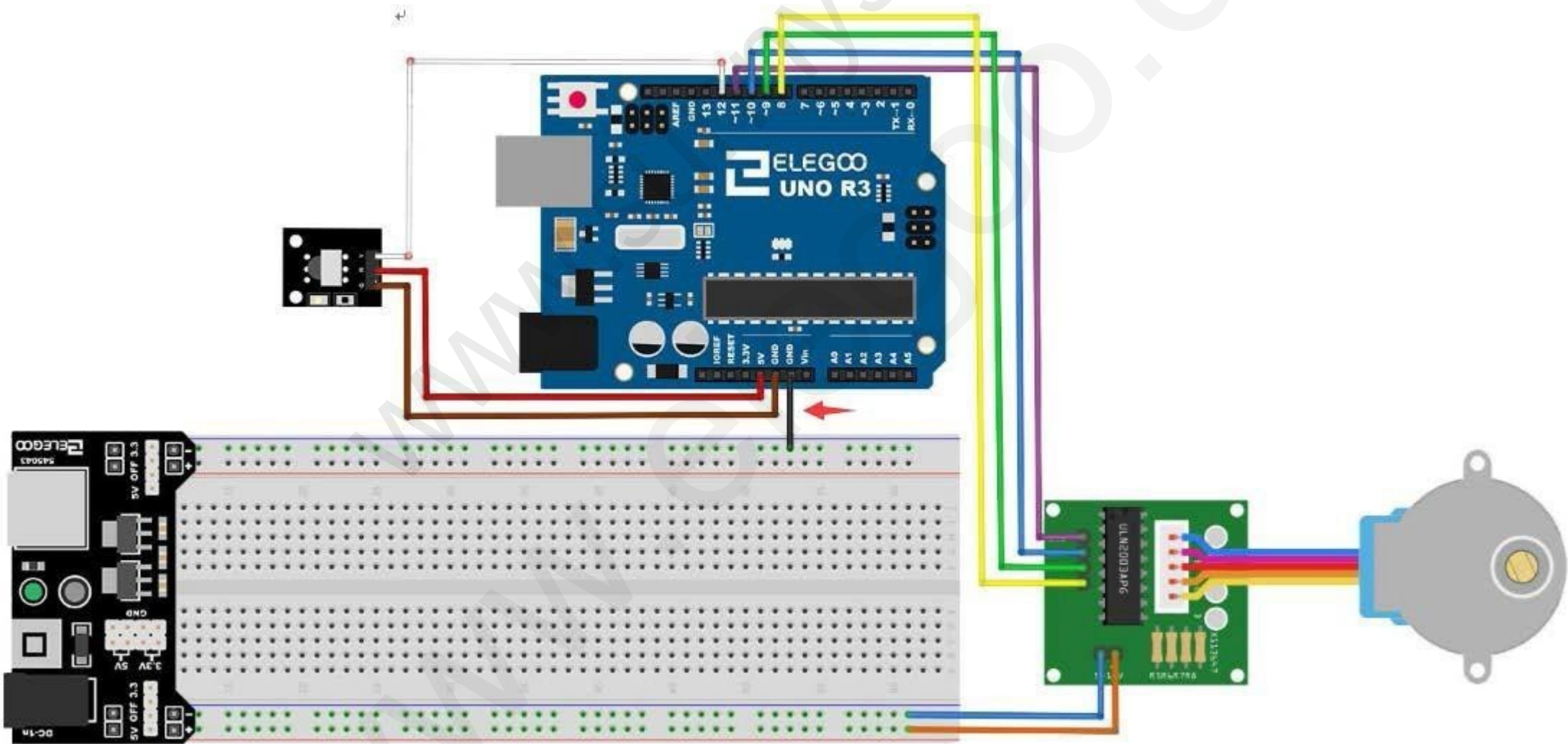


Diagrama de cablare



Folosim 4 pini pentru a controla motorul pas cu pas și 1 pin pentru senzorul IR. Pinii 8-11 controlează motorul pas cu pas, iar pinul 12 primește informațiile IR.

Conectați 5V și masa de la UNO la senzor. Folosiți alimentarea de pe placa de lucru pentru a alimenta motorul pas cu pas doar pentru a fi siguri, deoarece acesta poate consuma mai multă energie și nu dorim să deteriorăm sursa de alimentare UNO.

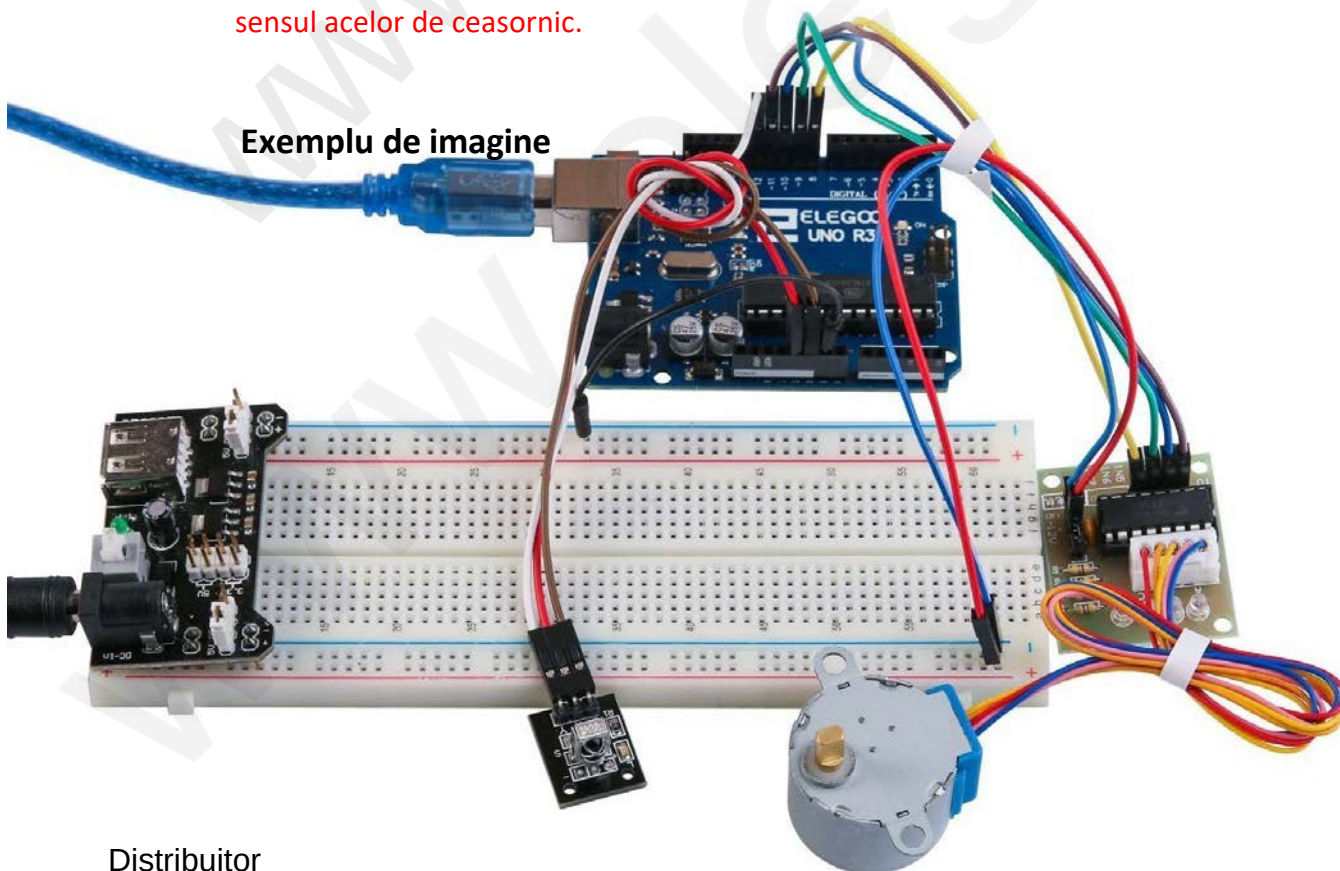
Cod

După conectare, deschideți programul din folderul de cod - Lecția 24 Controlul motorului pas cu pas cu telecomanda și faceți clic pe butonul UPLOAD pentru a încărca programul. Consultați Lecția 2 pentru detalii privind încărcarea programului dacă apar erori.

Înainte de a utiliza această funcție, asigurați-vă că aveți < IRremote > instalat.

< Stepper >library sau reinstalați-o dacă este necesar. În caz contrar, codul dvs. nu va funcționa. Consultați Lecția 1 pentru detalii privind încărcarea unui fișier de bibliotecă. Codul recunoaște doar 2 valori de la telecomanda cu infraroșu: VOL+ și VOL-. Atunci când VOL+ este apăsat pe telecomandă, motorul se va roti complet în sensul acelor de ceasornic.

Exemplu de imagine



Distribuitoare
Sunnysoft s.r.o.
Kovanecká 2390/1a
190 00 Praga 9
Republica Cehă
www.sunnysoft.cz