

PRVI KORAKI V ARDUINO – STIK S SVETOM IN RAČUNALNIKOM

▼ Milan Gaberšek in Slavko Kocijančič

Krmilnik Arduino ima veliko možnosti za komunikacije s svetom. Tokrat bomo predstavili svetlobni senzor in pošiljanje podatkov v računalnik prek serijskega vodila USB. Sestavili bomo napravo, ki bo samodejno vklopila lučko, v našem primeru svetlečo diodo, ko se bo stemnilo.

Material

- krmilnik Arduino Nano ali podoben,
- USB-kabel za povezavo krmilnika z računalnikom,
- prototipna ploščica (angl. breadboard),
- fotoupor, npr. GL5539,
- vezne žičke (najbolje rdeča, vijoličasta, modra in črna),
- upor vrednosti 1 k Ω (rjava, črna, rdeča in zlata),
- upor vrednosti 10 k Ω (rjava, črna, oranžna in zlata) ali več,
- svetleča dioda (rdeče barve).

Orodja in pripomočki

- osebni računalnik z nameščenim operacijskim sistemom Windows, Linux ali Mac OS,
- Arduino IDE, integrirano programsko razvojno okolje, ki je brezplačno dostopno na spletni strani www.arduino.cc.

Izvedba

Glede na električno shemo (slika 1) oziroma sliko, narejeno s pomočjo programa Fritzing (slika 2), povežemo posamezne elemente. Od uporabljenih elementov je nov le fotoupor, ki se mu upornost spreminja glede na količino vpadle svetlobe. Krajše ga označimo s kratico LDR, ki izhaja iz začetnic angleških besed Light Dependent Resistor, kar v prevodu pomeni svetlobno odvisni upor. Fotouporu podobno kot svetleči diodi zaporedno vezemo upor 10 k Ω ali večji (slika 3). Prvo nožico fotoupora povežemo na pin +5 V krmilnika Arduino (rdeča vezna žička), drugo povežemo na upor, tega pa podobno kot pri svetleči diodi neposredno na GND (črna vezna žička). V nasprotju s svetlečo diodo v spoj fotoupora in upora vstavimo še dodatno vezno žičko, v našem primeru modre barve, ki jo povežemo s pinom A0 na krmilniku.

Ko povežemo vse potrebno, se lahko lotimo programiranja krmilnika Arduino.

```
/*
Program Arduino - Stik s svetom
in računalnikom
*/
```

```
const int led = 2;
const int svetlobniSenzor = A0;
// Vrednost iz senzorja bomo
// shranili v spremenljivko
int vrednost = 0;

void setup() {
// Branje podatkov preko
// serijskega vodila USB
Serial.begin(9600);
pinMode(led, OUTPUT);
}

void loop() {
// Z ukazom analogRead preberemo
// digitalno vrednost senzorja
vrednost = analogRead(svetlobniSenzor);
// Dobljeni podatek izpišemo na zaslon
Serial.println(vrednost);
// Če je vrednost manjša od 100,
// svetleča dioda sveti,
// drugače jo izklopimo
if (vrednost <= 100) {
digitalWrite(led, HIGH);
} else {
digitalWrite(led, LOW);
}
}
```

Program prenesemo na krmilnik Arduino in že lahko preizkusimo, ali se svetleča dioda odziva na spremembo jakosti vpadle svetlobe na fotoupor. Pri tem lahko fotoupor pokrijemo s prstom, s čimer ustvarimo podobne pogoje kot ob mraku, zaradi česar bi morala dioda zasvetiti. Neodvisno od tega krmilnik prek povezave prek USB-vodila v računalnik ves čas pošilja tudi informacije o stanju na fotouporu, kar bomo spoznali v nadaljevanju.

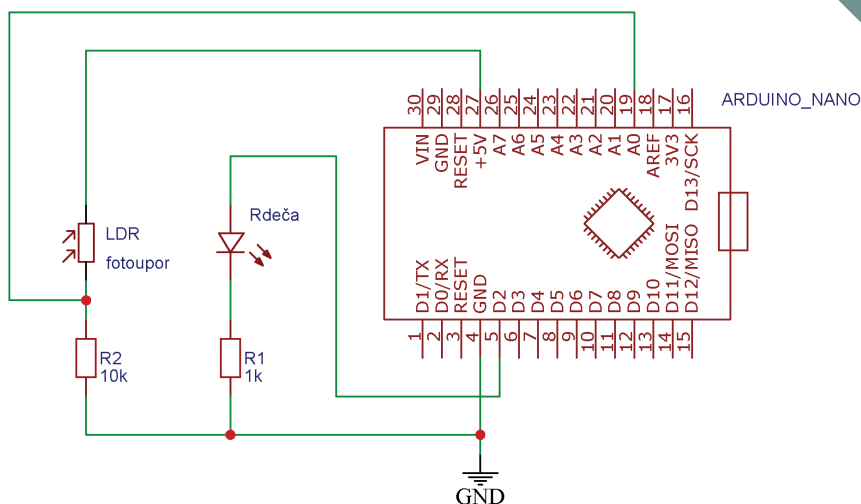
Delovanje

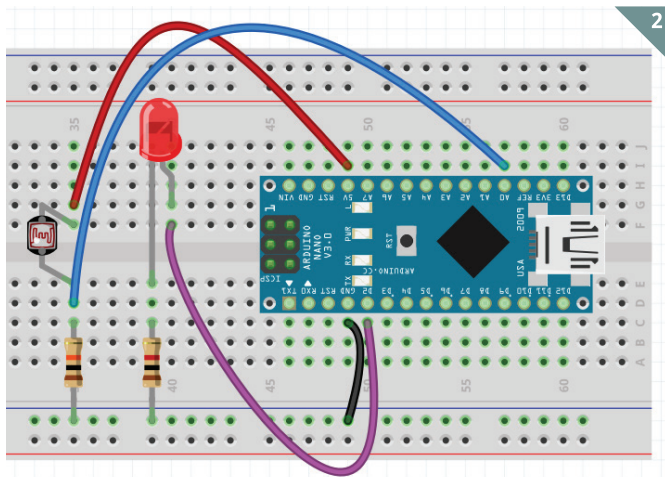
In kako vezje deluje? Fotouporu se s spremembo jakosti vpadle svetlobe spreminja upornost. Ker je k fotouporu

zaporedno vezan navaden upor R2, se napajalna napetost 5 V, ki jo zagotavlja krmilnik, porazdeli med oba upora. Ko se poveča jakost vpadle svetlobe, se upornost fotoupora zniža, kar pomeni, da se zniža tudi napetost na njem. Posledično se poveča napetost na uporu R2 (vsota obeh napetosti je ves čas 5 V). Krmilnik Arduino pri tem ves čas meri napetost med modro vezno žičko, ki smo jo povezali na pin A0, in priključkom GND, s čimer stalno meri napetost na uporu.

Doslej smo se srečevali le z digitalnima stanjema TRUE in FALSE (oziroma 1 in 0). Če bi merili napetosti na katerem od izhodov krmilnika Arduino, bi videli, da ta kaže v primeru TRUE kar vrednost napajanja, to je približno 5 V, v primeru FALSE pa približno 0 V. Torej sta možni le dve vrednosti napetosti, čemur pripišemo digitalni vrednosti 1 in 0. Pri vezju s fotouporom se napetost na pinu A0 spreminja med 0 in 5 V, čemur pravimo analogna vrednost. Krmilnik Arduino Nano ima za merjenje analognih napetosti predvidenih osem pinov, od A0 do A7 (od tu označena A). Mikrokrmilnik, srce krmilnika Arduino, »razmišlja« digitalno, torej le z vrednostma 0 in 1. Zato mora napetost, ki jo izmeri na katerem od pinov, pretvoriti v zaporedje ničel in enic, kar je za 10-bitno vrednost lahko videti nekako takole 0010110001. Tako zaporedje predstavlja dvojiško število (tudi binarno število), ki ga lahko pretvorimo v običajni desetiški sistem, kjer predstavlja desetiško številsko vrednost 177 (podrobnosti bomo izpustili). Tako lahko na vsakem od pinov A0 do A7 dobimo številске vrednosti med 0 (dvojiško 000000000) in 1023 (dvojiško 111111111).

Do vrednosti na posameznem analognem vhodu pridemo z ukazom analogRead, pri čemer v oklepaju navedemo ime pina A0 do A7. Podobno kot pri digitalnem vhodu oziroma izhodu je tudi pri analognih vhodih imena pinov priporočljivo shraniti v spremenljivke ustreznih imen, saj je s tem program preglednejši. To smo naredili v začetnem delu z navedeno konstante const int svetlobniSenzor = A0;. Trenutno vrednost na pinu A0 tako shranimo v spremenljivko s programsko vrstico vrednost = analogRead(svetlobni-



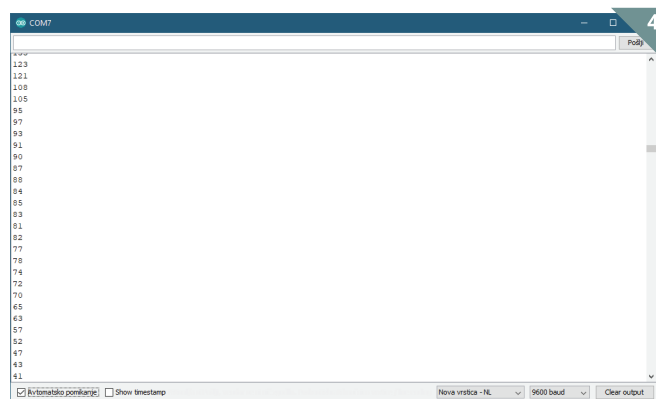
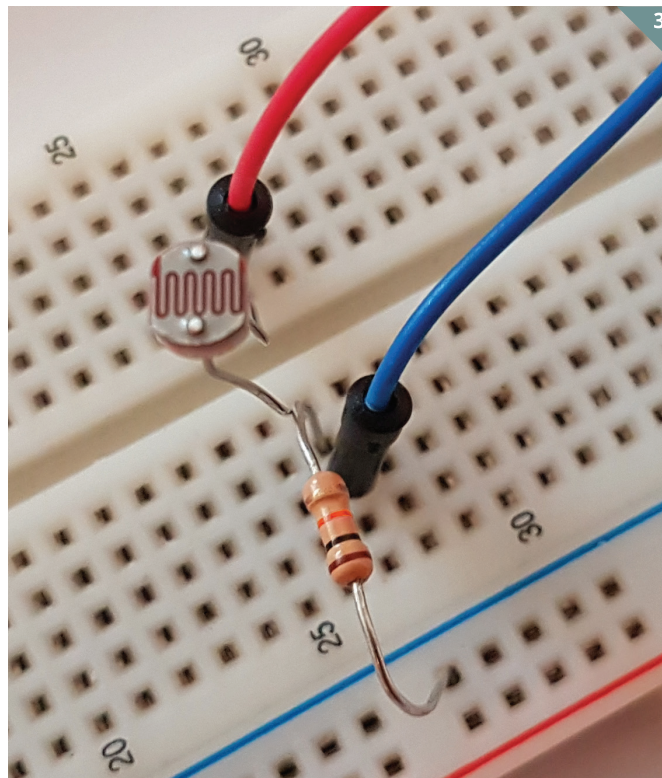


Senzor). Kot smo vajeni, z ukazom `if (vrednost <= 100)` vrednost spremenljivke (med 0 in 1023, odvisno od osvetljenosti) primerjamo s številom 100. Če je vrednost manjša ali enaka 100, kar pomeni, da se je stemnilo, z ukazom `digitalWrite(led, HIGH)` vklopimo svetlečo diodo, v nasprotnem primeru pa svetlečo diodo izklopimo.

In kako vemo, da moramo vrednost primerjati s številom 100, saj ne vidimo, kakšne vrednosti dobi v krmilnik Arduino? Najprej smo v okviru podprograma `void setup()` uporabili ukaz `Serial.begin(9600);`, s čimer smo krmilniku povedali, da bomo podatke prek vodila USB pošiljali v računalnik. Podatke nato v okviru podprograma `void loop()` z ukazom `Serial.println(vrednost);` dejansko pošljemo v računalnik. Vrednosti lahko vidimo v oknu razvojnega okolja Arduino IDE (slika 4), ki ga vklopimo v meniju Orodja, Serijski vmesnik, oziroma krajše s hkratnim pritiskom tipk `Ctrl, Shift in M`. Po prikazu okna preverimo, ali je v spodnjem delu okna v spustnem seznamu hitrost prenosa nastavljena na 9600 baudov. Če ni, jo nastavimo, saj se mora številka ujemati s tisto v programski kodi. Na podoben način kot vrednosti na vhodih lahko v računalnik pošljemo tudi druge podatke, celo besedilo. Za izziv lahko prek USB na zaslonu izpišete, da je zunaj svetlo oziroma temno. Poigrate se lahko tudi s pogojem 100, saj bo vaša naprava morda bolje delovala pri drugačni vrednosti. Če bi radi iz vhoda A0 dobivali večje vrednosti (maksimalna vrednost je 1023), lahko 10-kiloomski upor nadomestimo s kakim drugim višje vrednosti.

Zaključek

Uporaba povezave prek vodila USB in izpisa na računalnik nam lahko zelo olajša odčitavanje vrednosti na senzorjih, hkrati pa ponuja možnosti pošiljanja sporočil ali podatkov za poznejšo obdelavo v zahtevnejših programih, ki so za krmilnik prezahtevni. Uporaba analognih vhodov krmilnika omogoča priklop



različnih elektronskih senzorjev, ki jih z digitalnimi vhodi ne bi mogli uporabljati. Nekatere bomo spoznali tudi v prihodnjih prispevkih. Razmislite tudi, kako bi z manjšo predelavo programa napravo lahko uporabili za odgovor na staro vprašanje, ali luč v hladilniku sveti tudi po tem, ko zapremo vrata hladilnika. S pomočjo žepne svetilke lahko s prekinitvijo svetlobnega snopa napravo spremenite tudi v model preprostega alarma, še zlasti, če namesto svetleče diode uporabite piezo piskač.



- TN 1 motorni letalski RV-model basic 4 star
- TN 2 RV-jadrnica lipa I
- TN 3 RV jadralni model HOT-94
- TN 4 polmaketa letala cessna 180
- TN 5 RV-model katamarana KIM I
- TN 6 Timov HLG, jadralni RV-model za spuščanje iz roke
- TN 7 RV jadralni model HOT-95
- TN 8 Timov HLG-2, jadralni RV-model za spuščanje iz roke
- TN 9 tomy-E, elektromotorni jadralni RV-model
- TN 10 polmaketa lovskega letala polikarpov I-15 bis
- TN 11 jadralni RV-model gita
- TN 12 racoon HLG-3
- TN 13 akrobat 40, trenažni motorni RV-model
- TN 14 maketa vodnega letala utva-66H
- TN 15 RV-model trajekta
- TN 16 spitfire, RV polmaketa za zračni boj
- TN 17 trener 40, trenažni motorni RV-model
- TN 18 lupo, elektromotorni RV-model
- TN 19 P-40 warhawk, RV-polmaketa za zračni boj
- TN 20 potepuh, RV-model motorne jahte
- TN 21 bambi, šolski jadralni RV-model
- TN 22 slovenka, RV-jadrnica metrskega razreda

- TN 23 e-trainer, trenažni RV-model z električnim pogonom
- TN 24 P-51 B/D mustang, RV-polmaketa za zračne boje
- TN 25 messerschmitt Bf-109E, RV-polmaketa za zračni boj
- TN 26 RV-polmaketa Aeronca L-3
- TN 27 fokker E III, RV-polmaketa park-fly
- TN 28 vektra, RV-model z električnim pogonom v potisni izvedbi
- TN 29 Eifflov stolp, 1 m visoka maketa iz vezane ploščice
- TN 30 maketa bagra CAT 262
- TN 31 RV motorni letalski model z električnim pogonom orion
- TN 32 maketa hitre patrolne ladje SV Ankanan

6,50 €*

*Cena posameznega načrta, k čemu pristojejo poštni stroški

Naročila sprejemamo na:
ZOTKS, revija TIM,
Zatošna 65, 1000 Ljubljana,
tel.: 01/479-02-20,
e-pošta: revija.tim@zotks.si.