

## PRIMENA ARDUINO TEHNOLOGIJE NA GRAFIČKU AMBALAŽU THE APPLICATION OF ARDUINO TECHNOLOGY IN PACKAGING

David Nađ, Dragoljub Novaković, Stefan Đurđević, *Fakultet tehničkih nauka, Novi Sad*

### Oblast – GRAFIČKO INŽENJERSTVO I DIZAJN

**Kratak sadržaj** – U ovom radu je pokazano kako, uz pomoć elektronskih komponenti, može da se napravi uređaj koji bi se integrisao u ambalažu. Svrha ovog uređaja je da pokaže količinu svetlosti ili toplote kojoj je izložena ambalaža i da upozori ako se ovi parametri ne slažu sa propisanim.

**Ključne reči:** *Ambalaža, Arduino, Senzori*

**Abstract** – This paper shows how, using electronics, a device can be made that would be integrated into packaging. The purpose of this device is to show the amount of light or heat to which the packaging is exposed and to warn if these parameters do not comply with the designed ones.

**Keywords:** *Packaging, Arduino, Sensors*

### 1. UVOD

Potreba za pakovanjem i skladištenjem proizvoda se pojavila još pre antičkog doba. Da bi se prenela dalje od svog proizvođača, roba se morala na neki način zaštititi. Što je pakovanje bilo efikasnije i sigurnije, to je proizvod mogao dalje i bezbednije da putuje. Danas, uz pomoć dobre ambalaže i transporta, roba može da stigne sa jednog kraja sveta na drugi [1].

Projektovanje moderne i sigurne ambalaže je u međuvremenu postalo nauka sama za sebe. Mnoge kompanije se takmiče iz dana u dan da stvore najbolju i najatraktivniju ambalažu za svoje proizvode. Pored zaštite i transporta proizvoda, današnja ambalaža ima i mnoge druge uloge. Ona mora da bude zanimljiva i efikasna, da dokaže da je proizvod koji se nalazi unutar pravi za kupce, da privuče pažnju i da pobedi konkurenciju [2].

Kako bi se ovo sve postiglo, danas se u tu svrhu upotrebljavaju najsavremenije tehnologije. Vrlo su popularne i sve više se koriste takozvane pametne (engl. smart) ambalaže. Ove moderne ambalaže su u mogućnosti da pokažu stanje robe koja je u njih upakovana, kao i da li je potrebno ubaciti ambalažu u hladniji odnosno topliji prostor [3].

Aktivne ambalaže komuniciraju sa svojim proizvodima na podesan način i tako im pružaju dodatnu zaštitu u nekom obliku, dok inteligentne ambalaže prikazuju dodatne informacije o robi, koristeći dodatne senzore i drugu elektroniku [3].

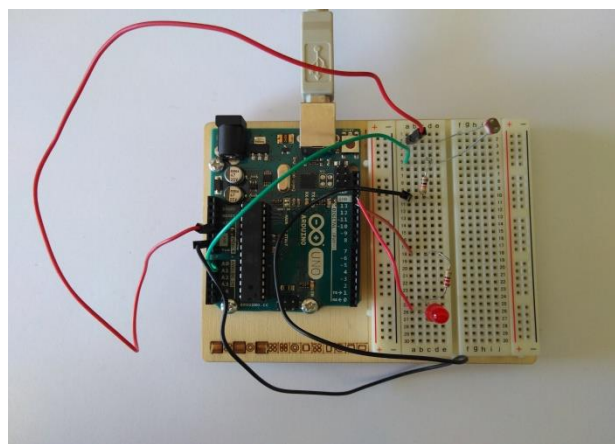
### NAPOMENA:

**Ovaj rad proistekao je iz master rada čiji mentor je bio dr Dragoljub Novaković, red. prof.**

Test uređaj koji se koristi u ovom radu konstruisan je pomoću Arduino Uno pločice [4], toplotnog i svetlosnog senzora, LED diode i dodatnih elektronskih komponenti. Programski kodovi uz pomoć kojih uređaj funkcioniše su napisani u Arduino IDE softveru koji je editor otvorenog tipa (enlgl. open-source). Jezik koji je korišćen za kodove je baziran na “Python” programskom jeziku.

### 2. EKSPERIMENTALNI DEO – KONSTRUKCIJA I POKRETANJE UREĐAJA

U ovom delu rada su opisani procesi uz pomoć kojih je autor napravio uređaje za prikaz informacija o toploti i količini svetlosti koja utiče na ambalažu. Obe varijante uređaja rade na sličnom principu, koristeći različite senzore, u konkretnom slučaju za toplotu i svetlost. Ovaj uređaj je napravljen koristeći Arduino početni set [5] (engl. Arduino Starter Kit), slika 1., i prikazuje samo mali deo onoga što bi se moglo u budućnosti osmisliti i napraviti za ambalažu, koristeći bolje i naprednije tehnologije koje već postoje ili će tek postojati (npr. nanotehnologije).



Slika 1. *Prototip uređaja.*

#### 2.1. Hardver

Za hardverski deo eksperimenta korišćeni su Arduino pločica, prototipska ploča, otpornici, LED dioda, svetlosni i toplotni senzori, odgovarajući provodnici i mikro kablovi za povezivanje i napajanje strujom putem USB kabla.

Kako bi uređaj mogao da funkcioniše, bilo je neophodno da svi elementi koji se postavljaju i dodaju uz pločicu formiraju zatvoreno strujno kolo. Da bi se ovo postiglo, koristi se prototipska ploča [5]. Ova ploča na sebi sadrži kontakte - pinove, koji su označeni od a-j, 1-30 i +,-. Bilo je potrebno voditi računa da su pojedini segmenti

prototipske pločice, označeni brojevima, povezani provodnim materijalom. Npr, ako se povežu dva elementa na kontakte 5, oni će biti kratko spojeni, bez ikakvih dodatnih kablova ili elemenata.

Po istom principu funkcionišu i kolone, označene sa + i -. Ove kolone se koriste da se iz Arduino pločice dovedu napajanje i uzemljenje na prototipsku ploču. Ovo je neophodno uraditi, u protivnom nijedan element, povezan za prototipsku ploču, ne može funkcionisati.

## 2.2. Proces sastavljanja uređaja

Kako bi uređaj mogao da funkcionise, potrebno mu je napajanje. Arduino pločica dobija napajanje pomoću USB kabla koji je povezan za PC računar ili pomoću baterije od 9 volti koja se dodatno kači za samu pločicu. U ovom slučaju je korišćen računar.

Osim napajanja Arduino pločice, bilo je potrebno dovesti napajanje i do prototipske pločice kako bi svi zakačeni elementi mogli da funkcionišu. Ovo se postiže pomoću provodnika koji je sa jedne strane priključen za utičnicu u Arduino pločici, a sa druge u jednu od kolona na prototipskoj ploči. Utičnica na Arduino koja služi za ovo je označena sa 5V. Ovim se započinje strujno kolo kojem se dodaju ostali elementi.

Red, gde je priključen provodnik, je tako dobio napajanje i u isti red je bilo potrebno prikačiti jedan kraj svetlosnog senzora [6]. Drugi kraj svetlosnog senzora je bio prikačen u drugi red. Kako su svi elementi provodni, to je i ovaj red dobio napon 5V.

Nakon formiranja kola, struja može biti prevelika za sam uređaj, tako da je bilo potrebno kontrolisati je. Za ovo služi otpornik [7]. Jedan kraj otpornika je bio priključen u isti red gde i drugi kraj senzora, a drugi kraj otpornika se kačio za novi, dodatni red.

Iz trećeg reda, kako bi se zatvorilo strujno kolo, bio je povezan provodnik koji vodi nazad na Arduino pločicu, u utičnicu GND, koja predstavlja uzemljenje.

Još jedan provodnik je bio prikačen, kako bi se vrednosti koje senzori očitavaju mogle prikazati na PC računaru u Arduino softveru. Jedan kraj dodatnog provodnika je bio u redu gde je jedan kraj svetlosnog senzora, a drugi u Arduino pločici, u pinu A0. Pinovi A0 – A5 predstavljaju mesta analognog ulaza - inputa. Mogu da prime informaciju i da je prikažu na kompjuteru ili na nekom dodatnom elementu (npr. LCD ekranu).

Za LED diodu [8] koja služi kao indikator kada na ambalažu deluje neadekvatna svetlost ili temperatura, napravljeno je dodatno malo strujno kolo. Jedan provodnik je povezan jednim krajem za Arduino pločicu, na pin 13. Drugi kraj provodnika se postavlja u novi red na prototipskoj ploči.

U isti red je postavljen drugi otpornik kako bi regulisao količinu struje koja prolazi kroz malo kolo. Jedan kraj otpornika postavljen je u isti red gde i dodati provodnik, a drugi u novi red.

U taj novi red je postavljen jedan kraj - anoda LED diode. Drugi kraj - katoda LED diode priključena je u novi red, za uzemljenje. Na kraju, provodnik je prikačen između katode LED diode i vodi nazad na Arduino pločicu, u pin GND kao što se može videti na slici 1.

U slučaju kada se koristi toplotni senzor [9] umesto svetlosnog, procedura sastavljanja test uređaja je potpuno ista, osim što toplotni senzor ima 3 kontakta umesto 2. Srednji kontakt toplotnog senzora služi da se u njegov red na protobordu zakači provodnik koji vodi u pin A0, Arduino pločice, za očitavanje informacija.

## 2.3. Softver za test uređaj

Najbitniji deo Arduino pločice je njegov mikrokontroler. Ovaj mikrokontroler je u stanju da primi veću količinu programskog koda i naredbi koje omogućavaju pločici da izvršava mnogo različitih i kompleksnih funkcija.

Softver u kom se Arduino pločica programira zove se Arduino IDE. Sam programski kod se podiže na pločicu preko USB kabla.

## 2.4. Arduino IDE

Arduino IDE (Integrated Development Environment) je aplikacija koje je napisana u programskom jeziku Java [10]. Koristi se za pisanje i podizanje programa i komandi na ploče koje su kompatibilne sa Arduino, ali u nekim slučajevima može se koristiti i za ploče drugih proizvođača. Ova aplikacija je otvorenog koda - open source, i može se besplatno preuzeti sa Arduino sajta.

Kada se otvori novi sketch, dobija se početni (engl. default) kod. Ovo je automatski podešeno u softveru. Pisanje koda za Arduino pločicu podrazumeva da se pišu 3 "bloka". U delu *void setup()* se upisuje deo koda koji služi da definiše neke parametre pisanog programa i ovo je komanda koje se pokreće samo jedanput tokom izvođenja koda. U delu *void loop()* se upisuje novi deo koda koji će Arduino pločica izvršavati više ili beskonačno puta. U ovom delu se obično nalazi glavni deo koda. Treći blok zapravo nema svoj definisani deo već se piše pre dva void dela. U tom delu koda definišu se promenljive i zadaju vrednosti nekim parametrima ukoliko je to potrebno.

## 2.5. Svetlosni senzor

U kodu su najpre zadati nazivi pinova za LED diodu i svetlosni senzor. Pomoću kodova "const int ledPin = 13;" i "const int ldrPin = A0;" objašnjeno je da će pin 13 nositi naziv ledPin, zato što je za njega prikačena LED dioda, a A0 nositi naziv ldrPin zbog svetlosnog senzora. Ovo su konstante koje se neće menjati.

U *void setup* sekciji se zadaju početni parametri programa. To su parametri koji definišu ceo program. Komanda „Serial.begin(9600);” zadaje brzinu kojom Arduino šalje informacije. U ovom slučaju Arduino šalje 9600 bita po sekundi. Ukratko, ovom komandom se zadaje brzina ponavljanja izvršavanja zadatah operacija.

Takođe u ovoj sekciji je definisan mod pinova. Naredba "pinMode(ledPin, OUTPUT);” definiše LED pin kao mesto koje će izbacivati povratne i obrađene informacije koje možemo da očitamo na prikazu, a naredba "pinMode(ldrPin, INPUT);” pin svetlosnog senzora kao mesto koje će da prima i očitava informacije iz okoline.

Kao što je već naznačeno ranije, *void loop* sekcija služi za glavni deo koda. Prvo što je u ovoj sekciji definisano je očitavanje količine svetlosti koja pada na svetlosni senzor. Ovo je urađeno uz pomoć komande „int ldrStatus = analogRead(ldrPin);”.

Svetlosni senzor funkcioniše tako što Arduino pločici šalje vrednost između nivoa 0 i nivoa 1024. Ovu vrednost Arduino pločica, u opsegu mogućih 5 V, prevodi u 10-bitni kod. 0 predstavlja kompletni mrak, dok je 1024 maksimalno osvetljenje. Ove brojke je moguće uz pomoć dodatnih kodova i formula prevesti u luxeve ili lumene ako je to korisniku potrebno, slika 2.

Sledeće što je napisano u kodu je *If Else* petlja koja služi da upali ili ugasi LED lampicu u zavisnosti od količine svetlosti kojoj je senzor izložen. Komanda "If (ldrStatus <=300)" služi da proveri da li je količina svetlosti kojoj je senzor izložen manja od vrednosti 300.

U slučaju da je situacija takva, sledeće što je definisano je da je potrebno uključiti LED lampicu pomoću naredbe "digitalWrite(ledPin, HIGH);". Posle toga, komandom "Serial.println("LDR is DARK, LED is ON");" se korisniku objašnjava da je količina svetlosti koja pada na senzor premala i da je LED lampica aktivna zbog toga.

U slučaju da je količina svetlosti kojoj je svetlosni senzor izložen veća od vrednosti 300 (else), komanda "digitalWrite(ledPin, LOW);" deaktivira LED lampicu. Ovim je ceo kod završen kao što se vidi na slici 2.

```

//set pin numbers
//const won't change
const int ledPin = 13; //the number of the LED pin
const int ldrPin = A0; //the number of the LDR pin

void setup() {

  Serial.begin(9600);
  pinMode(ledPin, OUTPUT); //initialize the LED pin as an output
  pinMode(ldrPin, INPUT); //initialize the LDR pin as an input
}

void loop() {

  int ldrStatus = analogRead(ldrPin); //read the status of the LDR value

  //check if the LDR status is <= 300
  //if it is, the LED is HIGH

  if (ldrStatus <=300) {

    digitalWrite(ledPin, HIGH); //turn LED on
    Serial.println("LDR is DARK, LED is ON");

  }
  else {

    digitalWrite(ledPin, LOW); //turn LED off
    Serial.println("-----");

  }
}

```

Slika 2. Kod za svetlosni senzor.

## 2.6. Senzor za temperaturu

Na sličnom principu funkcioniše i uređaj koji upozorava na temperaturu. Temperaturni senzor očitava temperaturu kojoj je izložen i tu informaciju šalje Arduino pločici. Ako je temperatura previsoka, pali LED lampicu i daje upozorenje. Ako je temperatura niska, LED lampica je isključena.

Kao i kod svetlosnog senzora, prvo što je zadato su nazivi pinova, samo ovaj put za temperaturni senzor i LED diodu. Kodovi "const int ledPin = 13;" i "const int tempPin = A0;" služe da objasne da pin 13 nosi naziv ledPin zbog prikazane LED diode, a A0 tempPin zato što je za taj pin prikazan temperaturni senzor.

U *void setup* sekciji je prvo zadata komanda "Serial.begin(9600);" kako bi Arduino slao 9600 bita po sekundi. Posle toga, slede komande "pinMode(ledPin, OUTPUT);" koja definiše LED pin kao mesto koje služi da izbacuje obrađene informacije koje se mogu očitati na prikazu, i "pinMode(tempPin, INPUT);" koje definiše pin temperaturnog senzora kao mesto koje će skupljati informacije iz okoline i očitavati temperaturu kojoj je senzor izložen.

U void loop sekciji je ispisan glavni deo koda. Prva naredba koja je zadata, "int temStatus = analogRead(tempPin);" služi da Arduino očita i unese podatke kolika temperatura utiče na temperaturni senzor, slika 3.

```

//set pin numbers
//const won't change
const int ledPin = 13; //the number of the LED pin
const int tempPin = A0; //the number of the TEM pin

void setup() {

  Serial.begin(9600);
  pinMode(ledPin, OUTPUT); //initialize the LED pin as an output
  pinMode(tempPin, INPUT); //initialize the TEM pin as an input
}

void loop() {

  int temStatus = analogRead(tempPin); //read the status of the TEM value

  float voltage = temStatus * 5.0;
  voltage /= 1024.0;
  float temperatureC = (voltage - 0.5) * 100 ;

  Serial.print(voltage); Serial.print(" volts - ");
  Serial.print(temperatureC); Serial.print(" degrees C - ");

  //check if the TEM status is >= 50
  //if it is, the LED is HIGH

  if (temperatureC >= 50) {

    digitalWrite(ledPin, HIGH); //turn LED on
    Serial.println("TEM is HOT, LED is ON");

  }
  else {

    digitalWrite(ledPin, LOW); //turn LED off
    Serial.println("-----");

  }
}

```

Slika 3. Kod za temperaturni senzor.

Kako bi temperatura koja utiče na temperaturni senzor mogla korisniku da se prikaže u stepenima celzijusa, napisano je još par linija koda. Kao i kod svetlosnog senzora, količinu toplote koja utiče na njega, toplotni senzor prenosi Arduino pločici kao vrednost između 0 i 1024. Komanda "float voltage = temStatus \* 5.0;" i "voltage /= 1024.0;" pretvara tu vrednost u volte. Na kraju, kako bismo mogli da izračunamo tačnu temperaturu u celzijusima, koristi se komanda "float temperatureC = (voltage - 0.5) \* 100 ;".

Kako bi korisnik mogao da dođe do informacije koliko volti i stepeni utiče na temperaturni senzor, zadate su naredbe "Serial.print(voltage); Serial.print(" volts - ");" koja ispisuje volte i "Serial.print(temperatureC); Serial.print(" degrees C - ");" koja ispisuje stepene celzijuse.

Na kraju je zadata *If Else* petlja koja objašnjava Arduinu da li da aktivira LED lampicu ili ne, u zavisnosti od količine temperature kojoj je izložen senzor. Za ovaj projekat, izabrani prag tolerancije je 50°C. Komandom “if (temperatureC >= 50)” se definiše ovaj prag.

Ako je temperatura kojoj je izložen senzor veća od 50°C, komanda “digitalWrite(ledPin, HIGH);” objašnjava Arduinu da je potrebno upaliti LED lampicu. Nakon toga, komanda “Serial.println(“TEM is HOT, LED is ON”);” objašnjava korisniku da je temperatura prevelika i da je lampica uključena zbog toga.

Ako je temperatura manja od 50°C, komanda “digitalWrite(ledPin, LOW);” objašnjava Arduinu da treba da ostane isključena LED lampica. Ceo kod za temperaturni senzor se može videti na slici 3.

### 3. ZAKLJUČAK

Pakovanja i ambalaže se menjaju svakodnevno. Napreduju u pogledu funkcionalnosti, dizajna i izdržljivosti. Danas, sa razvojem tehnologije, mogućnosti ambalaže su se proširile mnogostruko. “Smart” ili inteligentne ambalaže se već koriste za pojedine proizvode. U bliskoj budućnosti one će da postanu nešto standardno, vezano za svako pakovanje.

Ovaj rad je pokazao samo jedan mali deo načina na koji bi Arduino okruženje moglo u budućnosti da se iskoristi. Uređaj koji je testiran u ovom radu se može iskoristiti najefikasnije na sekundarnim ili tercijarnim ambalažama zbog svoje veličine. Njegova glavna primena jeste da detektuje uslove oko ambalaže, obradi ih, a zatim pošalje podatke ili upozorenje osobi koja je nadležna sa sigurnost proizvoda. U ovom radu pokazano je kako da uređaj daje upozorenje za neodgovarajuće uslove svetlosti i temperature, ali to je samo mali deo onoga što bi ovakvi uređaji u budućnosti mogli da rade i prikazuju.

Ono što je nabitnije je da je ovaj rad pokazao kako će ovakvi uređaji postati svakodnevni deo svih ambalaža i da su njihove mogućnosti zaista velike. Uz odgovarajuće programe i kodove, oni bi mogli da prikazuju, osim temperature i količine svetlosti, boju proizvoda kao i da li je proizvod prethodno neovlašćeno vađen iz pakovanja. Mogli bi da detektuju i prikazuju atmosferske uslove oko proizvoda kao što su zemljotresi, požari i kiše.

Uz nekoliko klikova na telefonu, mogli bismo i da dobijemo tačne informacije o tome koliko stepeni celzijusa trenutno ima naš proizvod, kao i to gde se on nalazi u nekom skladištu.

Kompjuteri postaju sve manji i kompaktniji i uskoro će mikrokompjuteri moći da se ugrađuju bez problema i na primarne ambalaže. Mali mikroprocesori bi mogli da javljaju kupcima na telefon da li je konzerva kojoj su prišli oštećena i da li joj je istekao rok upotrebe.

### 4. LITERATURA

- [1] <https://www.industrija.rs/vesti/clanak/istorija-pakovanja-proizvoda> (Pristupljeno 16.2.2020.)
- [2] <https://studenti.rs/skripte/ekonomija/ambalaza-i-pakovanje-robe-3/> (Pristupljeno 16.2.2020.)
- [3] <https://www.ingredientsnetwork.com/what-is-smart-packaging-news036518.html> (Pristupljeno 16.2.2020.)
- [4] <https://www.arduino.cc> (Pristupljeno 16.2.2020.)
- [5] <https://www.sk.rs/2016/10/sklp04.html> (Pristupljeno 16.2.2020.)
- [6] <https://www.seeedstudio.com/blog/2020/01/08/what-is-a-light-sensor-types-uses-arduino-guide/> (Pristupljeno 16.2.2020.)
- [7] <https://prakticnaelektronika101.wordpress.com/komponente/otpornik/> (Pristupljeno 16.2.2020.)
- [8] <https://www.rohm.com/electronics-basics/leds/what-are-leds> (Pristupljeno 16.2.2020.)
- [9] <https://www.electronicstutorials.ws/io/thermistors.html> (Pristupljeno 16.2.2020.)
- [10] <https://opensource.com/resources/what-arduino> (Pristupljeno 16.2.2020.)

Adresa autora za kontakt:

MSc David Nađ  
[Dejv2007@gmail.com](mailto:Dejv2007@gmail.com)

Prof. dr Dragoljub Novaković  
[novakd@uns.ac.rs](mailto:novakd@uns.ac.rs)

dr Stefan Đurđević  
[djurdjevic@uns.ac.rs](mailto:djurdjevic@uns.ac.rs)