

PROJEKTI IZ ELEKTRONIKE

Ime in Priimek: _____

Vpisna številka: _____

Študijska smer: _____

Letnik: _____

Datum: _____

Kazalo:

1 Načrtovanje in izdelava tiskanih vezij.....	3
2 Model optične komunikacije.....	4
3 Programiranje mikrokontrolerov.....	6
3.1 Arduino kot programator.....	7
3.2 Programiranje drugih AVR mikrokontrolerov.....	8
4 UART komunikacija (since 1962).....	9
5 Amplitudna modulacija signala.....	11
6 Pulzno-širinska modulacija.....	12
6.1 Krmiljenje servomotrijev.....	13
7 Integrirano vezje 555.....	14

1 Načrtovanje in izdelava tiskanih vezij

Tudi pri pedagoškem procesu je pomembno, da so vezja narisana nazorno (tako sheme, kot tudi sestavljanje vezja na prototipni ploščici). V ta namen lahko uporabljate različna orodja. Omenili bomo vsaj dva, ki sta prostodostopna in bi priporočali njihovo uporabo.

EasyEDA je spletno orodje, ki je namenjeno risanju elektronskih vezij, načrtovanju TIV in izdelavi potrebnih datotek za njihovo izdeavo.

Drugo programsko orodje pa je FRITZING. To orodje je namenjeno nazorni predstavitvi sestavljanja elektronskih vezij na testni plošči.

NALOGA:

1. V programskem orodju EasyEDA narišite shemo astabilnega-multivibratorja, izvozite sliko sheme in jo vključite v poročilo.
2. Za to vezje izrišite TIV in izpišite seznam elektronskih komponent. Izgled TIV izvozite in vstavite v poročilo. Prav tako vstavite seznam komponent.
3. V programskem orodju FRITZING sestavite vezje na prototipni ploščici in sliko vstavite v poročilo.
4. V programskem orodju FRITZING uredite tudi shemo vezja in sliko vstavite v poročilo vaje.

2 Model optične komunikacije

Predavnimi časi, ko svet še ni slišal za digitalno tehnologijo, se je gospod Samuel Morse domislil, da bi črke kodiral v kratke in dolge pulze. Te pa bi lahko kar najlažje pošiljal od ene točke do druge na najrazličnejše načine ... in telekomunikacije so prijokale na svet.

NALOGA:

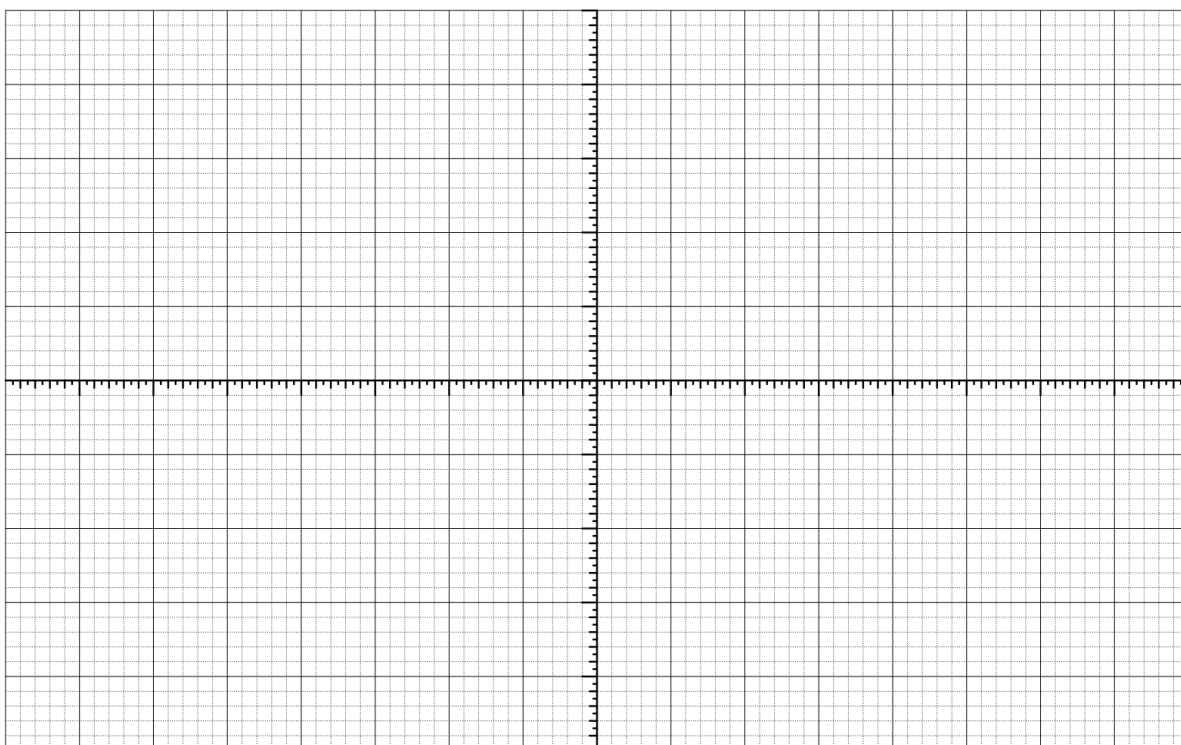
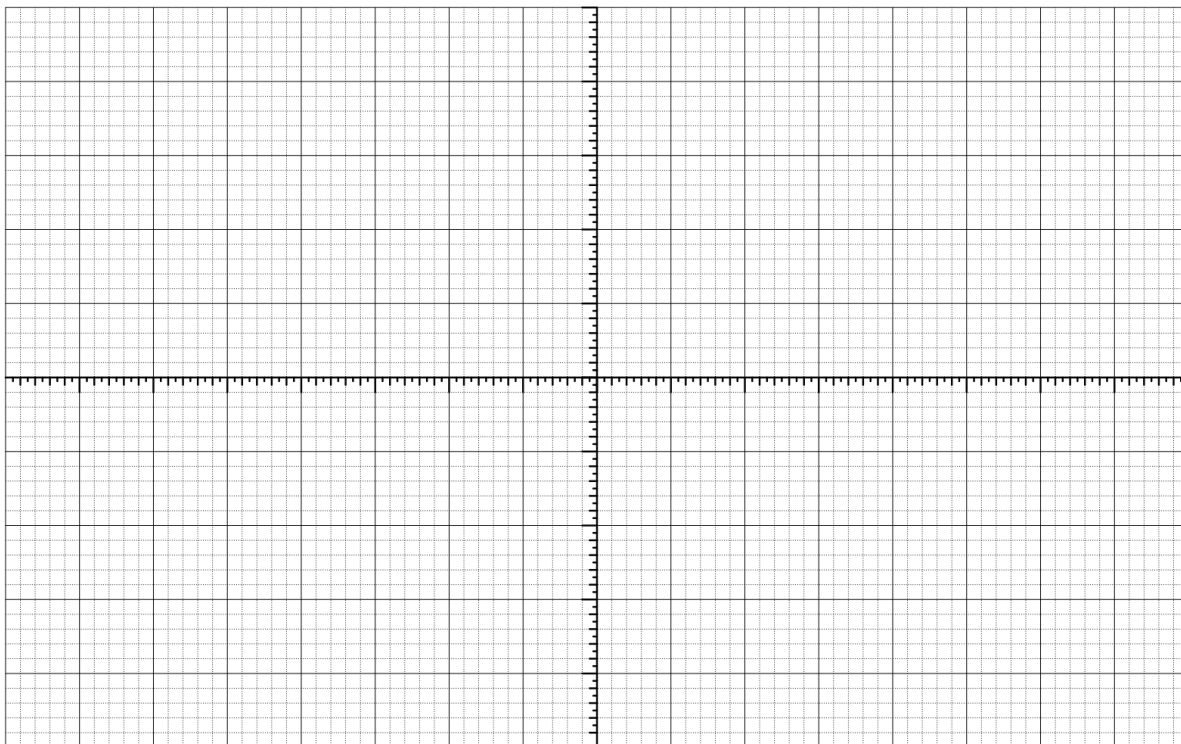
1. Za pošiljanje Morsejevih znakov uporabite žarnico.
2. Dolge in kratke pulze bomo pošiljali s svetlobnim oddajnikom. Načrtujete ustrezno rešitev (narišite shemo vezja) tako, da bomo s pritiski na tipko vklopili in izklopili svetilo (uporabite žarnico [12V in 0,6 A]). Bodite pozorni, da so tipke konstruirane za $I_{\max} = 100 \text{ mA}$. Skozi žarnico pa bo tekla večji tok. Načrtujte ustrezno rešitev.

NALOGA:

3. Izdelajte svetlobni sprejemnik, v katerega boste za zaznavanje osvetljenosti uporabili elektronski element s hitrim odzivom.
4. Analgni signal senzora modificirajte tako, da boste lahko nedvoumno podajali informacijo (npr. LED svetilo), ki jo je poslal svetlobni oddajnik.

NALOGA:

5. Za modifikacijo primerjajte dve različni vezji s podobnim namenom (komparator in schmitov sprožilnik), ter primerjajte njune lastnosti.
6. V graf narišite pare vhodnih in izhodnih napetosti obeh vezij. Izpostavite razliko v funkcionalnosti vezij.



3 Programiranje mikrokrmilnikov

Za projekte, ki vključujejo programabilno elektroniko, pogosto uporabljamo že izdelan krmilnik Arduino. Na teh vezjih lahko najdemo mikrokrmilnike proizvajalca Atmel. Najbolj pogosto uporabljena krmilnika (Arduino Uno in Arduino NANO) temeljita na mikrikrmilniku Atmega328p. Mikrokrmilniki na krmilnikih Arduino so že opremljeni s programom (angl. »boot loader«), ki poskrbi za ustrezno prepisovanje programske vsebine, ki jo računalnik pošlje preko USB vodila. Tako lahko enostavno programiramo mikrokrmilnike, ki so na ploščah krmilnikov Arduino.

NALOGA:

1. Iz programskega okolja Arduino IDE prepisite nastavitve programatorja:
 - Board (plošča): _____
 - Processor (procesor): _____
 - Port (vrata): _____
 - Programmer (programator): _____
2. Iz primerov, ki so vključeni v programskem okolju Arduino IDE izberite »Blink« in ga preskusite.

Za programiranje mikrokrmilnika skrbi odprtokodna programska koda - avrdude (<http://savannah.nongnu.org/projects/avrdude>), ki se izvaja v ozadju programskega okolja Arduino IDE. Proces programiranja lahko bolj natančno spremljamo tako, da vključimo možnost:

File --> Preferences :

Show verbose output during: compilation upload

NALOGA:

3. Prepisite ukazno vrstico programa avrdude za prenos strojne kode in opišite pomen parametrov.
(http://www.nongnu.org/avrdude/user-manual/avrdude_4.html#Option-Descriptions)

3.1 Arduino kot programator

Vendar, popolnoma novi mikrokrmilniki programa za nalaganje programske kode nimajo. Zato jih moramo sprogramirati s posebnimi vezji za ta namen – programatorji. Na srečo vlogo programatorja lahko izvedemo tudi s krmilnikom Arduino.

To lahko storimo z naslednjimi koraki:

1. Najprej si moramo pripraviti Arduino krmilnik v vlogi porgramatorja. To storimo tako, da na Arduino krmilnik maložimo program : `File --> Examples --> ArduinoISP`
2. Nato povežemo programator (Arduino krmilnik) z mikrokrmilnikom tako kot prikazuje spodnja tabela:

Programator (Arduino krmilnik)	AVR mikrmokrmilnik
10 - SS	RESET
11 - MOSI	MOSI
12 - MISO	MISO
13 - SCK	SCK
+5V	Vcc
GND	GND

3. In nazadnje dodamo še kondenzator s kapaciteto 10 μ F med nožici RESET in GND na Arduino krmilniku, ki ga uporabljamo kot programator.
4. Sedaj smo priporavljeni za prenos progama... v Arduino IDE nastavite pravilne nastavitve plošče in mikrokrmilnika, ki ga programirate.
5. Nato izberite komunikacijska vrata na kateri je priključen programator,
6. ter v `Tools --> Programmer` nastavite `Arduino as ISP`.
7. Odprite in naložite program:
`File --> Examples --> 01.Basics --> Blink`
8. Če želite, da bo novi mikrokrmilnik, ki ga programirate deloval v Arduino krmilniku, morate nanj naložiti Bootloader. To storite tako, da izberete:
`Tools --> Burn Bootloader`

NALOGA:

1. Preskusite Arduino Nano¹ krmilnik v vlogi programatorja tako, da z njim sprogramirate drug Arduino Nano² krmilnik s programom `Blink.ino`
2. Ali lahko ta (2) Arduino Nano krmilnik uporabljamo samostojno z Arduino IDE (brez dodatnega programatorja)?
3. Na krmilnik Arduino Nano (2) naložite Bootloader porgram.

1 Ta krmilnk naj bo programator

2 Ta pa naj bo sprogramiran preko programatorja in ne neposredno preko USB povezave.

3.2 Programiranje drugih AVR mikrokrmilnikov

Z Arduino lahko programiramo tudi druge mikrokrmilnike družine AVR, kot so to naprimer mikrokrmilniki ATmegaXXX ali ATTinyXXX...

Da lahko to storimo moramo storiti naslednje korake:

1. Prav tako kot v vaji 3.1 (koraki 1, 2 in 3) si moramo pripraviti Arduino krmilnik v vlogi programatorja.
2. V Arduino IDE moramo pravilno nastaviti Board. To storimo tako, da najprej:
 1. Na strani <https://github.com/arduino/Arduino/wiki/Unofficial-list-of-3rd-party-boards-support-urls> poiščemo ustrezno *_index.json datoteko s podatki o našem AVR mikrokrmilniku in URL skopiramo v
`File --> Preferences --> Additional Boards Manager`
 2. Nato izberemo `Tools --> Board --> Board Manager`, v iskalno polje vpišemo vpišemo iskano knjižnico (npr: `ATTinyCore`), jo izberemo in kliknemo `[Install]`.
 3. Zopet izberemo `Tools --> Board -->` in izberemo želen mikrokrmilnik (npr.: `ATTiny2313`) ter nastavimo še ostale potrebne nastavitve...
3. Sedaj lahko preskusimo povezavo tako, da krmilnik sprogramiramo s programom `Blink.ino`. Po potrebi boste morali spremeniti `LED_BUILTIN` spremenljivko v ustrezno številko pina.

NALOGA:

1. Sestavite vezje in shemo narišite v poročila vaj.
2. Preskusite sprogramirati mikrokrmilnik s programom **Blink.ino** in krmilnik pravilno opremite z LED.

4 UART komunikacija (since 1962)

Nekaj več o [UART komunikaciji](#)³ si lahko preberete vsepovsod na svetovnem spletu. Ker jo uporabljamo že več kot pol stoletja, lahko rečemo, da sodi med osnovne komunikacijske protokole.

NALOGA:

1. Preučite UART protokol in prerišite časovni potek signala (teoretično).
2. Na teoretičnem primeru označite pomen napetostnega signala, kako si sledijo podatkovne informacije, označite start in stop bit ter nakažite kako hitrost komunikacije vpliva na sam potek (npr.: baud = 9600 b/s).

Komunikacija UART je tako razširjena, da jo vključujejo v skoraj vse programabilne elektronske komponente in Arduino NANO ni nobena izjema. Mikrokrmilnik ATmega328p vsebuje enoto za komunikacijo UART in je dostopna na priključkih 0 (Rx) in 1 (Tx). Preko te enote lahko pošiljamo/sprejemamo podatke drugih zunanjih naprav.

NALOGA:

3. Preučite [shemo Arduino UNO/NANO krmilnika](#)⁴ in poiščite omenjeno povezavo.
4. Preskusite naslednji program mikrokrmilnika za pošiljanje nekega besedila računalniku in odziv spremljajte v serijskemu oknu programa ArduinoIDE:

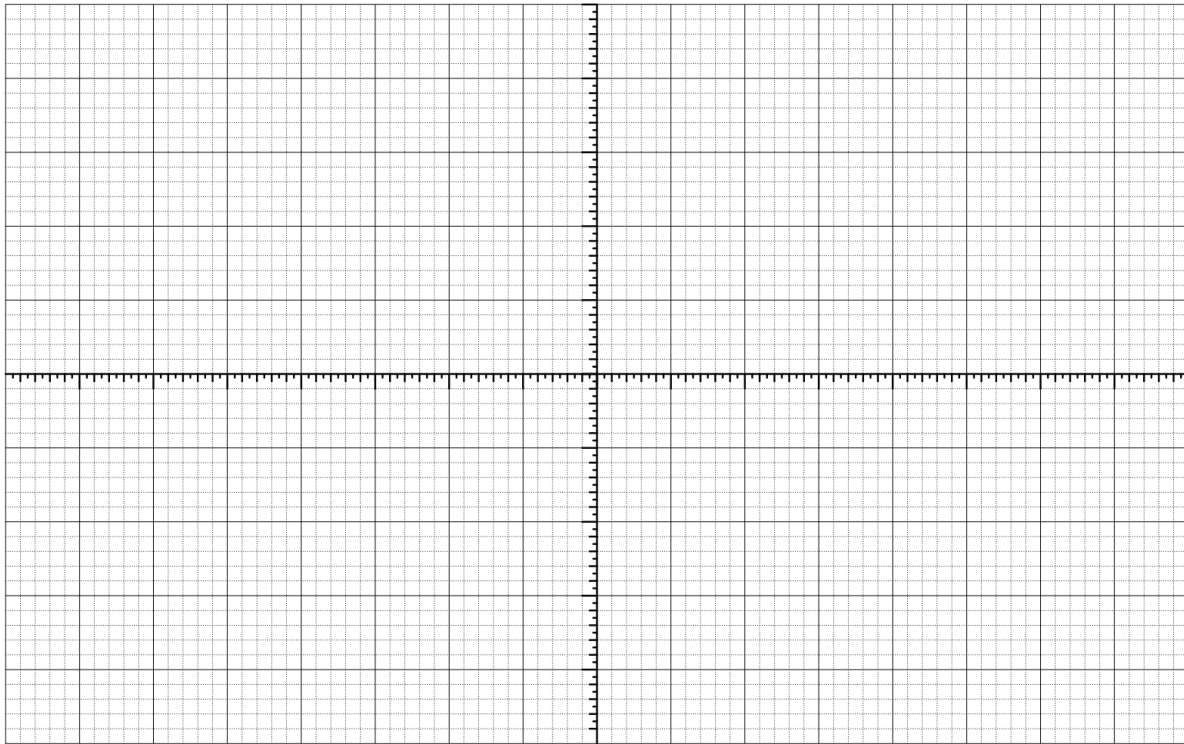
```
void setup() {  
  // nastavitev hitrosti komunikacije:  
  Serial.begin(9600);  
}  
  
void loop() {  
  Serial.println("Pozdravljen svet.");  
  delay(1000);  
}
```

5. Z osciloskopom posnemite napetostni signal pošiljanja enega samega znaka in parametre primerjajte s teoretičnimi lastnostmi komunikacije:

```
Serial.print("A");
```

³ https://en.wikipedia.org/wiki/Universal_asynchronous_receiver-transmitter

⁴ <https://www.arduino.cc/en/uploads/Main/arduino-uno-schematic.pdf>

**NALOGA:**

6. Sestavite dva preprosta senzorja (kot delilnik napetosti). Enega za osvetljenost in drugega za temperaturo. Senzorja naj bosta priključena na analogni vhod mikrokrmilnika (vsak na svojega). Prebrane podatke pa pošiljajte računalniku. Lahko si pomagate s primerom programa, ki je že priložen v Arduino IDE okolju (program dodajte v poročilo):

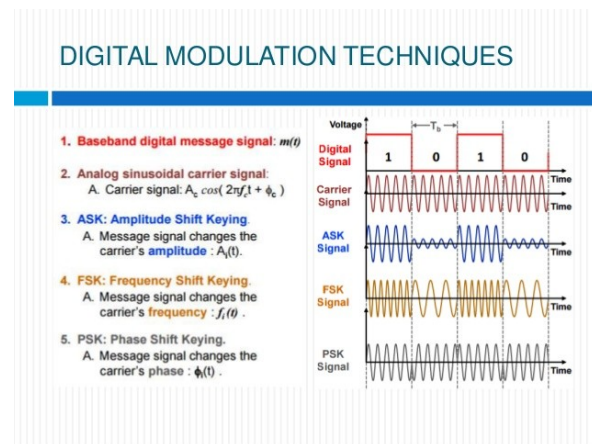
File --> Examples --> 01.Basics --> ReadAnalogVoltage

7. Dopolnite program tako, da boste lahko preko UART protokola izbirali katere podatke naj mikrokrmilnik pošilja računalniku. Naprimer: če iz računalniške strani pošljemo mikrokrmilniku znak »t«, naj nam mikrokrmilnik pošlje podatek o temperaturi. Pomagate si lahko s primerom:

```
char inData;
void setup() {
  Serial.begin(9600);
}
void loop() {
  if (Serial.available()){ //Če dobimo UART podatek,
    inData = Serial.read(); //ga preberemo.
    if (inData == "t"){ //Nato pa preverimo,
      //to-do //če je ta podatek = t.
    }
    if (inData == "o"){ //Ali pa če = o.
      //to-do
    }
  }
}
```

5 Amplitudna modulacija signala

Naprimera, da želimo poslati nek podatek v digitalni obliki. Le-ta je seveda sestavljen iz logičnih "1" in "0" (zgoraj na sliki⁵ 5.1). Pri prenosih signalov le-te pogosto moduliramo. Digitalna amplitudna modulacija je ena najpreprostejših modulacij, ki ga prikazuje slika 5.1 na točki 3. Tako moduliran signal se večinoma uporablja za prenos podatkov daljinskega upravljalnika hišnih naprav (TV, setup-box, radio ...). Za sprejemanje in demodulacijo teh signalov pa najčejekrat uporabimo elektronski element kot naprimer TSOP32230⁶.



Slika 5.1: Različne modulacijske tehnike.

NALOGA:

- Preko svetlobnega oddajnika boste morali poslati svoje ime in priimek na sprejemnik, ki je priključen na glavnem računalniku v učilnici. Komunikacija naj bo skladna z UART protokolom (RS232, 8 bit, Brez paritete in 1 stop bit). Hitrost komunikacije (ang.: baud rate) prilagodite pogojem sprejemnika. V vajo dodajte sheme in potrebne programe. Za lažji potek reševanja naloge sledite spodnjim nalogam.

- Poiščite podatke za sprejemnik TSOP32230 in izpolnite na naslednje bistvene lastnosti elementa:

Razporeditev nožic: 1-_____, 2-_____ in 3-_____

f_0 (nosilna frekvenca) = _____

t_{pi} (minimalni čas trajanja poslanega signala za logično 1) = _____

λ (valovna dolžina svetlobe pri maksimalni občutljivosti) = _____

- Za sprejemnik TSOP32230 narišite časovno odvisnost sprejetega in oddanega signala, ki opisujeta delovanje sprejemnika.

- Za svetlobni oddajnik boste uporabili diodo LD271-L. Iz dokumenta z njenimi podatki prepisite njene bistvene lastnosti:

λ = _____

U_k (kolenska napetost) = _____

I (nazivni tok) = _____

I_{MAX} (kratkotrajni maksimalni tok) = _____

- Primereno frekvenco nosilnega signala generirajte z astabilnim multivibratorjem, ki zagotavlja enako dolžino zgornjega in spodnjega nasičenja.

- Modulacijo lahko naredite z eno od digitalnih operacij (IN, ALI, NE, NE-IN, NE-ALI, ESKL.-ALI). Preverite kater način ustreza 3. točki.

5 vir: <https://www.slideshare.net/nidhibaranwal/digital-modulation-techniques-57572540>

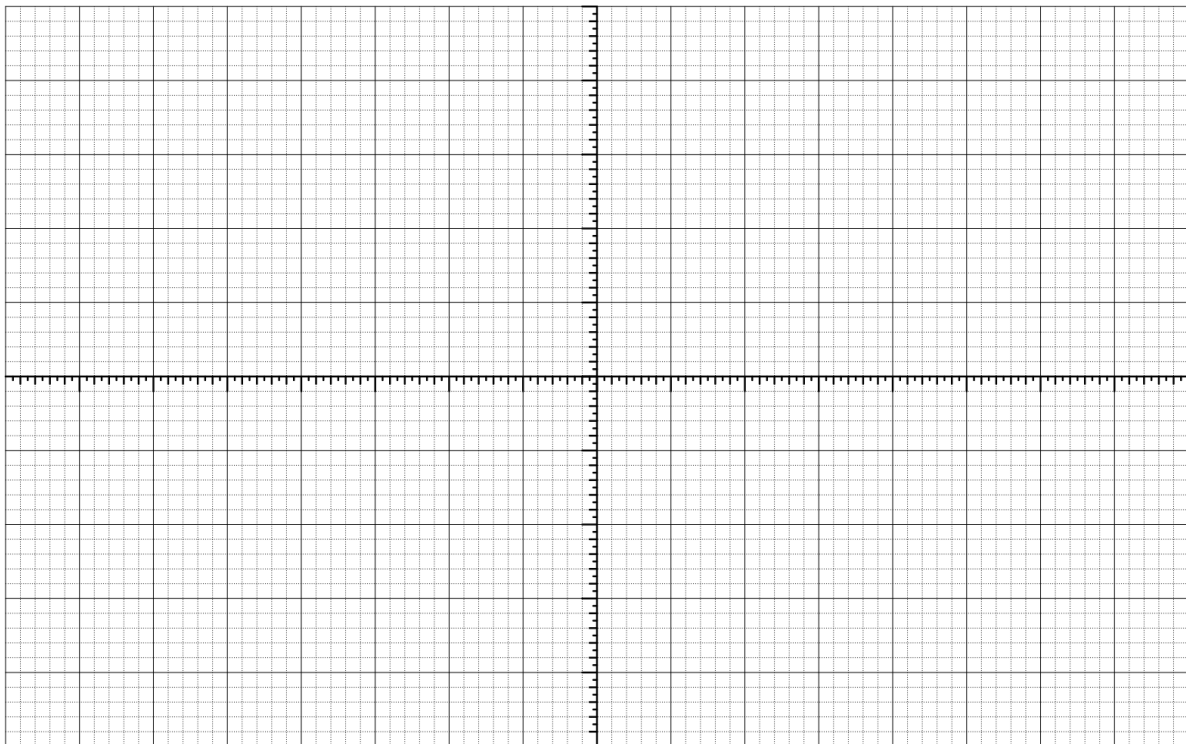
6 <https://www.vishay.com/docs/82489/tsop322.pdf>

6 Pulzno-širinska modulacija

V elektroniki pogosto uporabljamo pulzno-širinsko modulacijo (angl. Pulse width modulation, v nadaljevanju PWM) za krmiljenje motorjev. Bodisi s PWM signalom neposredno krmilimo moč, ki jo dovajamo motorju⁷ ali celo tak signal uporabimo za prenos informacije, kot je to v primeru servo-motorjev⁸. V obeh primerih pa je signal digitalen, pri čemer je dolžina trajanja logične 1 odvisna spremenljivka.

NALOGA:

1. Pulzno-širinsko modulacijo napravite s preprosto »while« zanko, ki "teče" v mikrokrmilniku. V njej krmilite nek digitalni izhod, ki ga postavljate v logično "1" in "0" za različno dolge intervale. Program (glavni del) dodajte v poročilo vaje.
2. Napetostni signal preverite z osciloskopom in ga narišite.



7 <https://www.allaboutcircuits.com/textbook/semiconductors/chpt-11/pulse-width-modulation/>

8 https://en.wikipedia.org/wiki/Servo_control

NALOGA:

3. V nadaljevanju si pri programiranju pomagajte s funkcijo `analogWrite(x)`⁹. Spremenjen program priložite poročilu.
4. Na izhod priključite svetlečo diodo in opazujete njeno svetilnost. Za vsaj tri primere določite:
nosilno frek., delež log. "1" in zapišite ugotovitve glede svetilnosti.
 1. _____
 2. _____
 3. _____
5. Vezju (mikrokrmilniku) dodajte potenciometer tako, da boste lahko programsko odčitali s sredinjskega priključka. Glede na to vrednost krmilite izhodni signal PWM modulacije. Narišite vezje in priložite program.
6. Nato dopolnite vezje še s potrebno elektroniko, da boste lahko krmilili enosmerni motor. Narišite vezje, ga sestavite in preizkusite.

6.1 Krmiljenje servomotrjev

V uvodu v PWM modulacijo smo omenili krmiljenje servo motrjev. Ker se ti motorji zelo pogosto uporabljajo v modelarstvu je prav, da jih поблиže spoznamo. Oglejte si primer uporabe funkcije `servo.write()`¹⁰.

NALOGA:

7. Na krmilnik priključite servo-motor in ga krmilite s primernim programom tako, da bo motor spreminjal svojo orientacijo gredi glede na položaj potenciometra. Narišite vezje in priložite program.

9 <https://www.arduino.cc/reference/en/language/functions/analog-io/analogwrite/>

10 <https://www.arduino.cc/en/Tutorial/Knob>

7 Integrirano vezje 555

Tako imenovani "časovnik 555" je integrirano vezje (angl. Integrated circuit - IC), ki ga uporabljamo v različnih aplikacijah, kjer želimo generirati časovno odvisne napetostne pulze. To nam pride prav za izgradnjo časovnih zakasnitev, oscilatorjev itd.

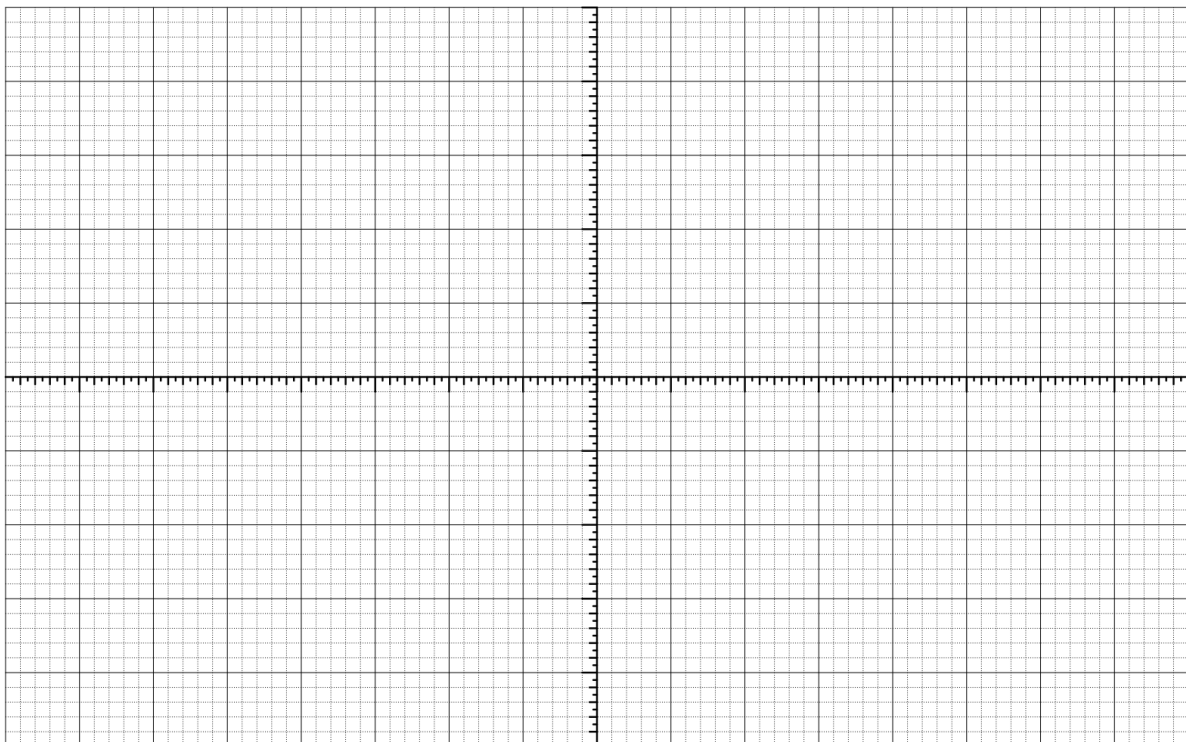
Čeprav so ga na trg dali že leta 1972 (pred 45 leti), je zaradi nizke cene in enostavne uporabe¹¹ še danes pogosto uporabljeno IC. Pravzaprav je IC 555 najbolj priljubljeno vezje nasploh. Pravjo¹², da še niso napisali knjige za elektroniko, v kateri nebi bilo projekta s tem vezjem, zato naj tudi ta skripta ne bo izjema.

NALOGA:

1. Na spletnih straneh poiščite kako je zgrajeno vezje 555, prerišite shemo vezja in poskušajte razumeti njegovo delovanje. Odgovorite po čem je to vezje dobilo svoje ime.

ODGOVOR: _____

2. Integrirano vezje 555 uporabite kot astabilni-multivibrator tako, da boste lahko nanj lahko priključili svetlečo diodo, ki jo boste videli utripati. Narišite shemo vezja in Preverite napetostni signal z osciloskopom.



3. Integrirano vezje 555 zvežite v način monostabilnega-multivibratorja tako, da ko boste s pritiskom na tipko sprožili en sam pulz, ki bo trajal približno 3 s.

11 <https://electronicsclub.info/555timer.htm>

12 Charles Platt, Make: Electronics (Learning by Discovery)

NALOGA:

1. Če je vhodna napetost dalj časa manjša od $2/3$ napajalne napetosti, se pri osnovni izvedbi m.m. izhodnji pulz ne konča po določenem času. To napako lahko rešimo z dodatnim RC členom. Narišite shemo rešitve in jo preskusite.

8 Operacijski ojačevalnik - praktična vezja

Med zelo pomembnimi elementi ožičene elektronike najdemo tudi operacijski ojačevalnik. IC (integrirano vezje) vsebuje več bipolarnih tranzistorjev, ki so skrbno načrtovani in postavljeni v svoje delovne točke. Z nekaj več dodatnimi elementi pa lahko operacijski ojačevalnik spremenimo v nekaj zelo uporabnih vezij.

NALOGA:

V simulacijskem okolju Yenka sestavite in priložite naslednja vezja. Za vsa naredite diagram časovnih odvisnosti vhodnih in izhodnih napetosti:

1. Komparator napetosti.
2. Invertirajoči ojačevalni sistem z ojačanjem $A = |3|$.
3. Seštevalni ojačevalni sistem, ki bo sešteval 4 vhodne napetosti v razmerju 1:2:4:8. Tak sistem lahko uporabimo kot digitalno analogni pretvornik.
4. Diferencialni ojačevalni sistem z ojačanjem $A = |3|$.
5. Instrumentacijski ojačevalni sistem z ojačanjem $A = |3|$.
6. Nizko-propustni sito.
7. Visoko-propustno sito.
8. Elektronski integralni člen.