

Реализација управљања, регулације, аквизиције електронског (BLDC) мотора и снимање излазних сигнала мотора

УВОД

Упознавање са електронским (BLDC) мотором, његовим принципом рада, елементима за регулацију и управљање BLDC мотора, упознавање и рад са мултифункционалним инструментом Analog Discovery 2 и снимање напонских и струјних сигнала BLDC мотора.

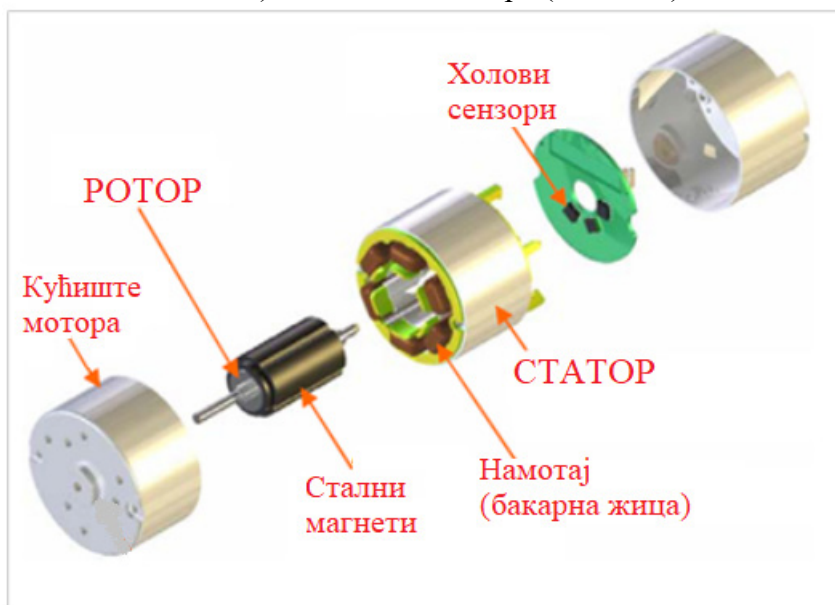
ТЕОРИЈСКИ ДЕО

Електронски мотор или скраћено BLDC (енг. BrushLess Direct Current) мотор јесте једносмерни (DC) мотор без четкица, односно мотор који у својој конструкцији не садржи комутатор (колектор) као остале једносмерне машине, већ се за његову комутацију користи електронско коло (одатле назив електронски мотор), а такође поред основног назива за овај мотор могу се наћи и називи као што су : безкомутаторски, безколекторски, безчеткични мотор.

Постоји више типова ових мотора, у зависности од њихове конструкције, употребе сензора, према броју полова и слично. У овој лабораторијској вежби коришћен је електронски мотор без сензора, са сталним магнетима на ротору и трофазним намотајем спојеним у звезду на статору.

Основни делови BLDC мотора

BLDC мотор састоји се од статора (на коме је постављен трофазни намотај), ротора (на коме се налазе стални магнети) и Холових сензора (опционо).

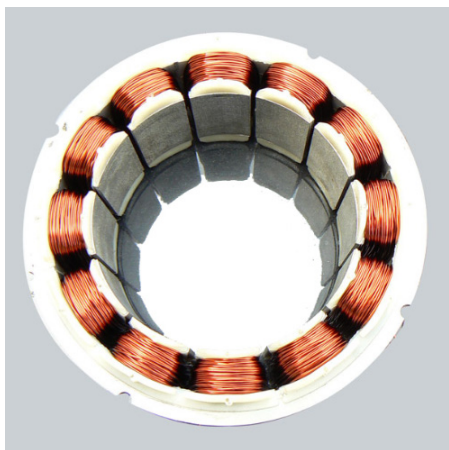


Конструкција BLDC мотора

Статор

Статор BLDC мотора израђује се од танких челичних лимова наслаганих један на други, у чије прорезе се смештају намотаји од бакарне жице. Статор BLDC мотора сличан је статору индукционих мотора, али се намотавање самог статора разликује у односу на индукционе моторе, односно на статору BLDC мотора постављају се три намотаја који се кратко спајају у звезду, а сваки од та три намотаја одређен је тачним бројем намотавања бакарне жице.

У зависности од конструкције мотора, у пракси се срећу два типа статора приказани на сликама испод :



Статор IN-RUNNER типа BLDC мотора



Статор OUT-RUNNER типа BLDC мотора

Ротор

Ротор BLDC мотора израђује се од сталних магнета, кружно распоређених око вратила ротора. Број магнетних полова (N и S) који се постављају на ротор, креће се од најмање једног пара полова, па све до осам пари полова, а магнетни материјали који се користе за израду су најчешће феритни магнети (феромагнети), а такође све већу употребу налазе и материјали као што су : неодијум (Nd), самаријум-кобалт (SmCo), као и легура неодијума, гвожђа и бора (NdFeB).

У зависности од конструкције мотора, у пракси се срећу два типа ротора приказани на сликама испод :



Ротор IN-RUNNER типа BLDC мотора



Ротор OUT-RUNNER типа BLDC мотора

Принцип рада

Принцип рада BLDC мотора заснива се на наизменичном пропуштању струје кроз одређене фазе, чиме се успоставља обртно магнетно поље, које ротор тежи да стигне, тј. тежи да се постави у осу са тим магнетним пољем. На пример, ако крајеве трофазног намотаја обележимо са А, В и С, прво пропустимо струју да тече кроз фазни намотај А, у њему се ствара магнетно поље које привлачи сталне магнете ротора, тј. ротор тежи да се постави у осу са тим пољем. Затим у одређеном тренутку прекидамо напајање фазе А и почињемо напајање фазе В, која је просторно померена за 120° , па ће и магнетно поље бити померено за 120° и ротор ће наставити да се креће ка том магнетном пољу у циљу да се постави у осу са њим. Затим, непосредно пре постављања ротора у осу са магнетним пољем фазе В, врши се искључење те фазе и укључење фазе С, чиме смо постигли да је магнетно поље опет померено за 120° и ротор наставља даље да се креће у том смеру. Овим наизменичним укључењем/искључењем фаза постигли смо да се магнетно поље стално мења, тј. ротор ће стално тежити да то магнетно поље стигне и постави се у осу са њим, односно ротор ће се обртати и остварили смо рад машине.

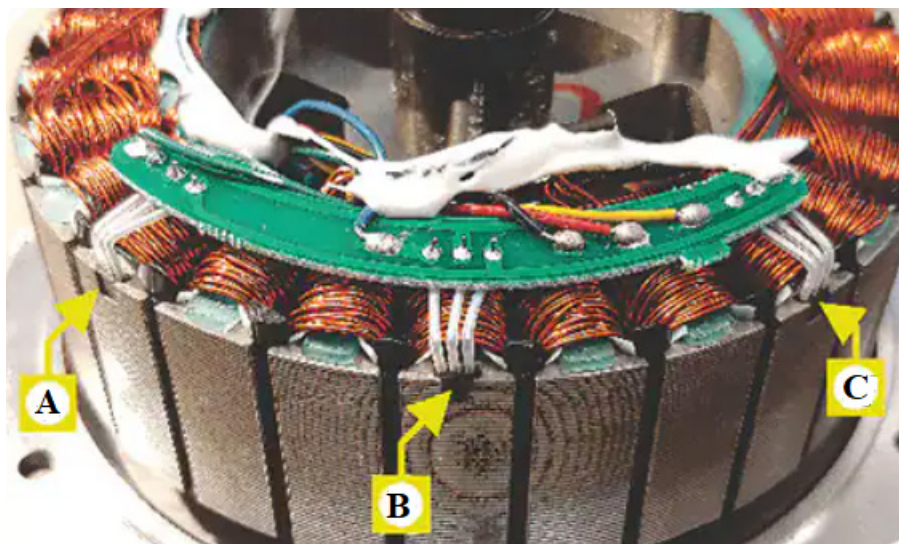
За тачност и синхронизацију укључења/искључења фаза користи се давач сигнала (положаја) ротора, а то су најчешће Холови сензори или енкодери.

Холов сензор

Холов сензор представља уградни елемент BLDC мотора, обично се поставља на статору на непогонском крају вратила мотора и његова основна улога јесте праћење положаја ротора при раду мотора и слање тих информација микроконтролеру који их обрађује и врши успешну комутацију, тј. контролу управљања и рада мотором.

Холов сензор обртање ротора претвара у сигнал бинарног кода (код нула и јединица), где се за сваки бројни код зна тачан положај ротора. Постоји укупно осам комбинација бинарног кода, а то су кодови : 000, 001, 010, 011, 100, 101, 110, 111. Од свих осам, кодови 000 и 111 јављају се када имамо грешку у систему (јер је немогуће да све фазе буду искључене - 000 или све фазе укључене - 111), а осталих шест кодова јавља се при нормалном раду мотора.

Када имамо промену сигнала то је знак да је постигнут равнотежни положај ротора са том фазом и да треба ту фазу искључити, а укључити следећу фазу у жељеном редоследу. На пример Холов сензор А укључује фазу С, сензор В укључује фазу А, сензор С укључује фазу В.

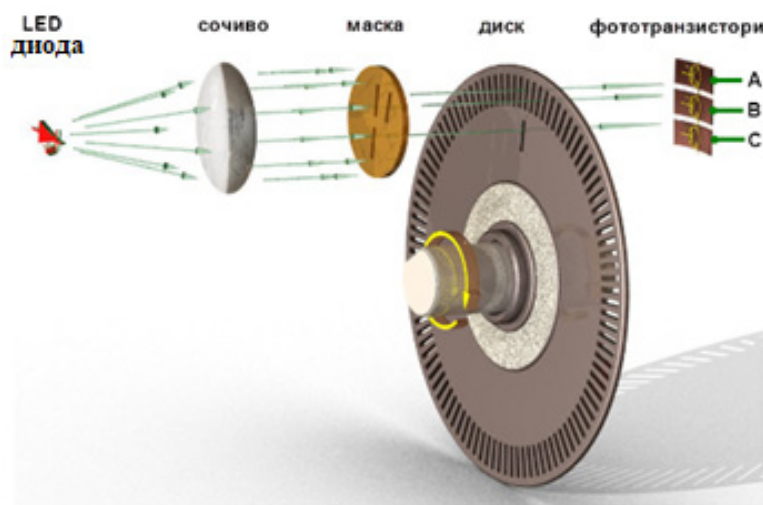


Холови сензори постављени на статору мотора

Енкодер

Осим Холових сензора за одређивање положаја ротора, користе се још и енкодери као уређаји који обртно кретање ротора претварају у електричне сигнале, на основу којих се даље помоћу микро-контролера врши управљање и контрола BLDC мотора. Основни делови сваког енкодера (апсолутни или инкрементални), су: кућиште енкодера, LED диода, диск са зарезима и фотоелемент (углавном је то фототранзистор).

Принцип рада енкодера заснива се на томе да се диск зарезима углавном поставља на вратило ротора мотора или се спаја са вратилом преко одређене спојнице, са једне стране диска налази се LED диода која емитује светлост, а са друге стране диска налазе се фототранзистори. Приликом рада мотора ротор мотора се обрће, а са њим и диск са зарезима (јер су директно спрегнути). LED диода емитује светлост, која пролази кроз зарезе на диску у зависности од обртања диска пролазиће и светлост у различитим тренуцима, фототранзистор прима тај светлосни сигнал и преводи их у електрични сигнал који иде до микропроцесора који тај сигнал обрађује и врши контролу и управљање мотора.

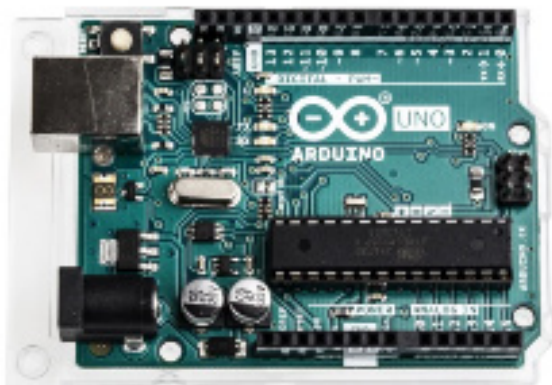


Конструкција енкодера (инкрементални енкодер)

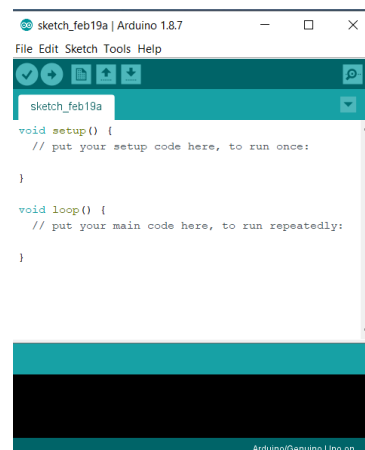
Микроконтролерска платформа ARDUINO

Ардуино је open-source електронска прототипна платформа, заснована на флексибилном и једноставном за употребу хардверу и софтверу. Ова платформа има веома широку примену код програмера и дизајнера, за управљање и контролу неког процеса, уређаја или машине, јер је платформа отвореног кода и базира се на програмском пакету који је једноставан за коришћење и разумевање. У односу на неке друге микроконтролере има слабије перформансе, али му је цена нижа па зато налази и широку примену.

Сама микроконтролерска платформа састоји се од хардвера и софтвера. Хардвер се од једноставног отвореног хардверског дизајна Arduino плоче са Atmel AVR процесором и пратећим улазно-излазним елементима. Софтвер се састоји од развојног окружења који чини стандардни компајлер (преводи изворни код у машински код) и bootloader-а (покретач оперативног система) који се налази на самој плочи. За програмирање Ардуина користи се ARDUINO IDE развојно окружење, за писање програма користи се програмски језик Java, C/C++ и слично.



Хардвер ARDUINO UNO платформе



Софтвер ARDUINO UNO платформе

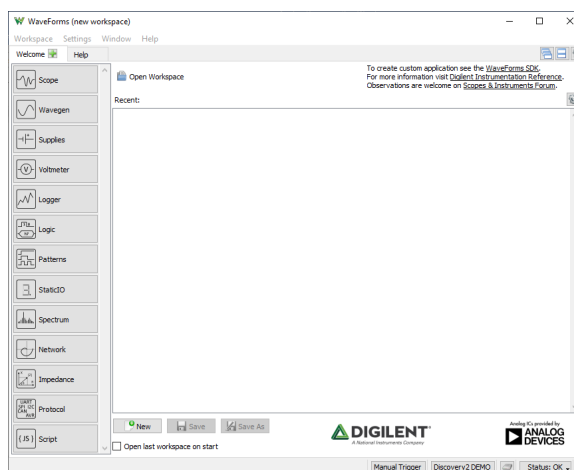
Мултифункционални инструмент ANALOG DISCOVERY 2

Овај инструмент омогућује мерење, визуелизацију, генерисање, снимање и контролу сигнала свих врста. У принципу то је USB осцилоскоп и логички анализатор, генератор сигнала, може служити као DC напајање, pattern генератор и мрежни анализатор који у себи садржи мноштво других мерних инструмената, чиме постижемо уштеду у простору, новцу, постижемо лакше и једноставно прикупљање одређених вредности сигнала.

Коришћење овог инструмента веома је једноставно, потребно је само покренути софтвер WaveForms и повезати овај инструмент са рачунаром, преко USB-а, где се на рачунару веома практично и лако подешавају жељени параметри, подешавање функције инструмента (да ради као волтметар, генератор сигнала и слично), да вршимо мерење одређених вредности, задајемо који се сигнал мери, снимамо одзиве тих сигнала и слично.



Изглед ANALOG DISCOVERY 2 инструмента

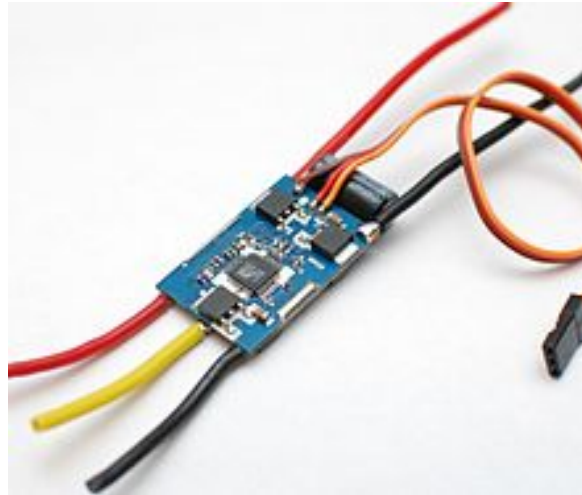


Радно окружење WaveForms софтвера

Електронски регулатор брзине (ESC)

Када желимо остварити пуштање у рад BLDC мотора који не поседује Холове сензоре, најчешће се користи електронски регулатор брзине (ESC), којим се врши контрола брзине и управљање BLDC мотора (што је случај у овој лабораторијској вежби).

Сам ESC представља једно електронско коло које на себи садржи изводе за повезивање напајања (једносмерног), изводе за повезивање са мотором (преко којих му задаје референце и контролише га), изводе за повезивање са Arduinoм (може и директно, али је то најчешће преко неког чворишта), помоћу кога се овај контролер програмира.



Изглед ESC контролера

ЛАБОРАТОРИЈСКА ВЕЖБА

Задатак вежбе

За конкретан BLDC мотор, остварити његово пуштање у рад и помоћу мултифункционалног инструмента (осцилоскопа) Analog Discovery 2 снимити његове напонске и струјне сигнале.

Опис вежбе

Употребом осцилоскопа Analog Discovery 2 снимити и приказати следеће реализације:

1. Снимити напонске сигнале у све три фазе
2. Снимити напонске и струјне сигнале једне фазе за задату брзину у празном ходу
3. Снимити напонске и струјне сигнале једне фазе при константној брзини уз помоћ PID регулације, при задатом оптерећењу

Спецификација опреме и прибора за извођење вежби:

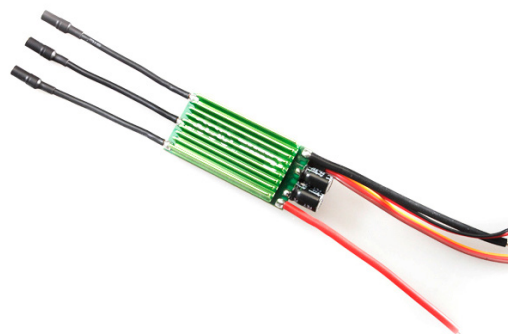
Електронски (BLDC) мотор “MARS BL3738-1300kv” :

Произвођач: Gens ACE
Снага: 350W
Напон: 9-12.6 V
Тежина: 265g
Брзинска константа: 1300kv
Дужина мотора (L1): 38,5mm
Пречник мотора (D1): 27,2mm
Speed control: 45A



Електронски регулатор брзине (ESC) “MARS BL45A” :

Произвођач: Gens ACE
Називна струја: 45A
Максимална струја: 55A
Улазни напон: 5-25.2 V
Тежина: 62g



Микроконтролерска платформа “ARDUINO” :

Произвођач: Smart Projects
Модел: Arduino Uno
Број аналогних пинова: 6
Број дигиталних пинова: 14
Улазни-излазни напон: 5V
Улазо-излазна струја по пину: 20mA
Тип USB конектора: USB-B
Тежина: 25g



Ротациони енкодер “GTS06-0P-RA1000B-2M” :

Произвођач: SAH Electronics

Тип енкодера: Инкрементални

Број импулса по обртају: 1000 импулса/обр

Резолуција: 3000 обр/мин

Напон: 5-24 V

Максимална струја: 45mA

Број канала: 2 (A,B)



Осцилоскоп “ANALOG DISCOVERY 2” :

Произвођач: Digilent

Тип: Двоканални осцилоскоп

Тип канала: Диференцијални

Број аналогних канала: 2

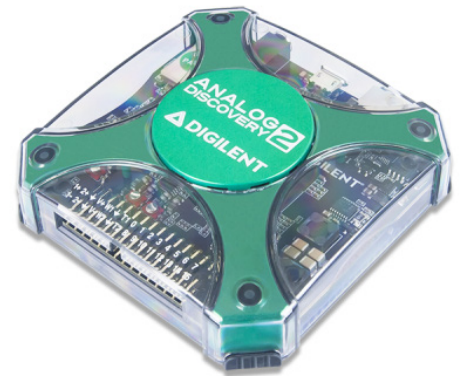
Број дигиталних канала: 16

Резолуција: 14-bit

Улазни опсег: $\pm 25V$

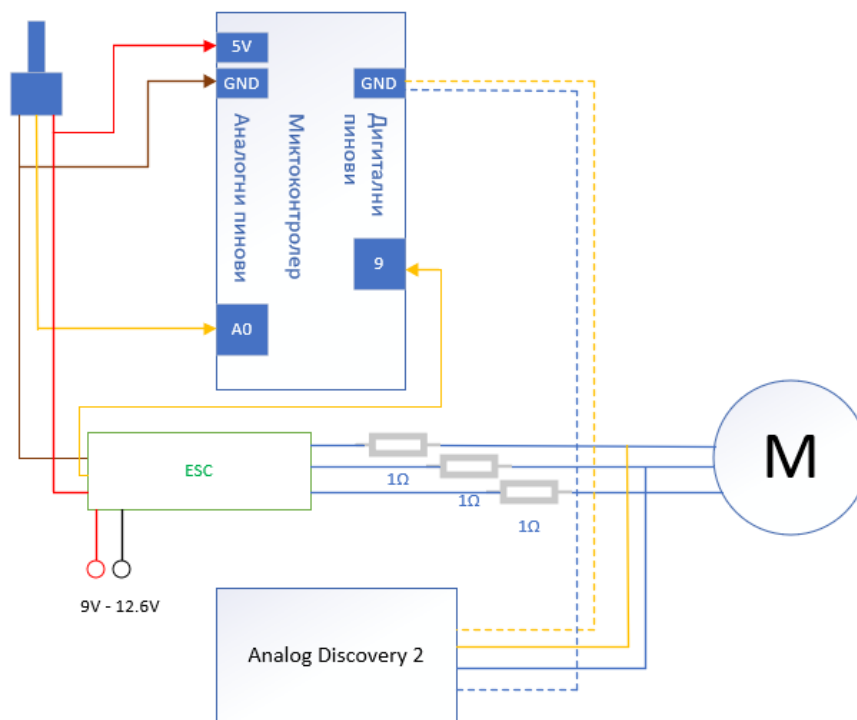
Апсолутна резолуција: $0.32mV$

Напонски offset: $\pm 5V$

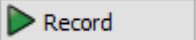
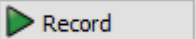


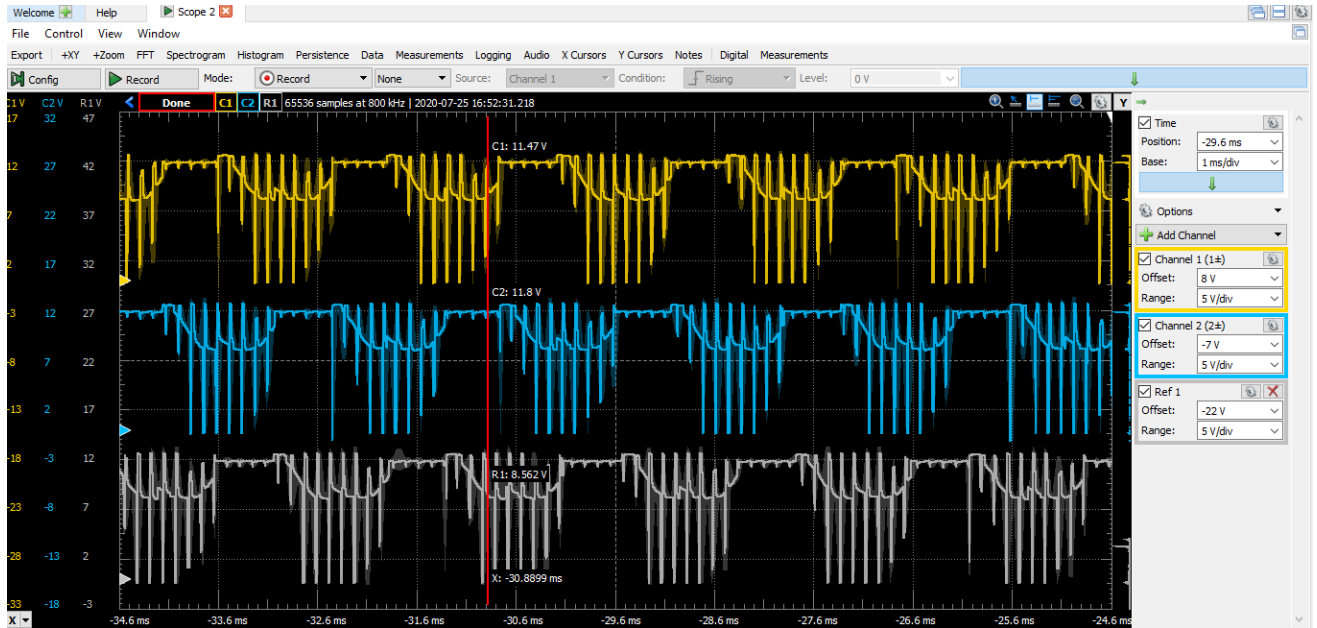
1. Снимање напонског сигнала у све три фазе

Електрична шема и начин повезивања



Поступак извођења вежбе:

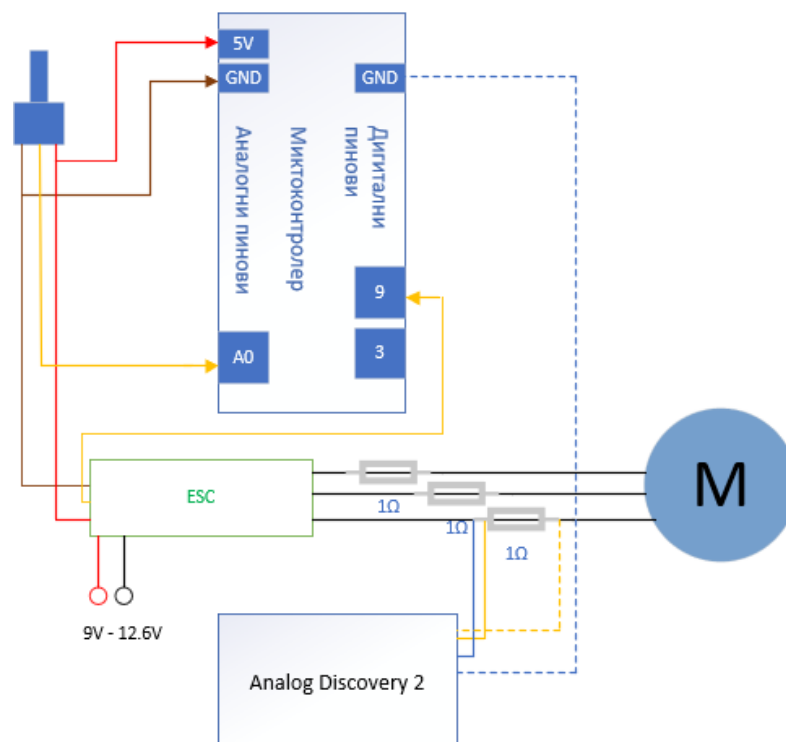
- 1) Повезати елементе са шеме
- 2) Улазе осцилоскопа повезати на прву и другу фазу
- 3) Употребити Arduino код из Прилога 1
- 4) Покренути софтвер WaveForms (Scope)
- 5) Извршити следећа подешавања софтвера :
 Trigger:
 Source:Channel 2;
 Type: Edge;
 Condition: Falling;
 Mode: Record.
- 6) Задати жељену брзину мотора, помоћу потенциометра
- 7) Пустити мотор у рад
- 8) Кликом на  извршити снимање
- 9) Сачувати добијене вредности у једној од фаза (првој)
- 10) Искључити напајање и променити улазе осцилоскопа на другу и трећу фазу
- 11) Извршити следећа подешавања софтвера:
 Trigger:
 Source:Channel 1;
 Type: Edge;
 Condition: Rising;
 Mode: Record
- 12) Пустити мотор у рад
- 13) Кликом на  извршити снимање
- 14) Сачувати добијене вредности и анализирати изглед добијених сигнала




Резултати напонског сигнала у све три фазе BLDC мотора

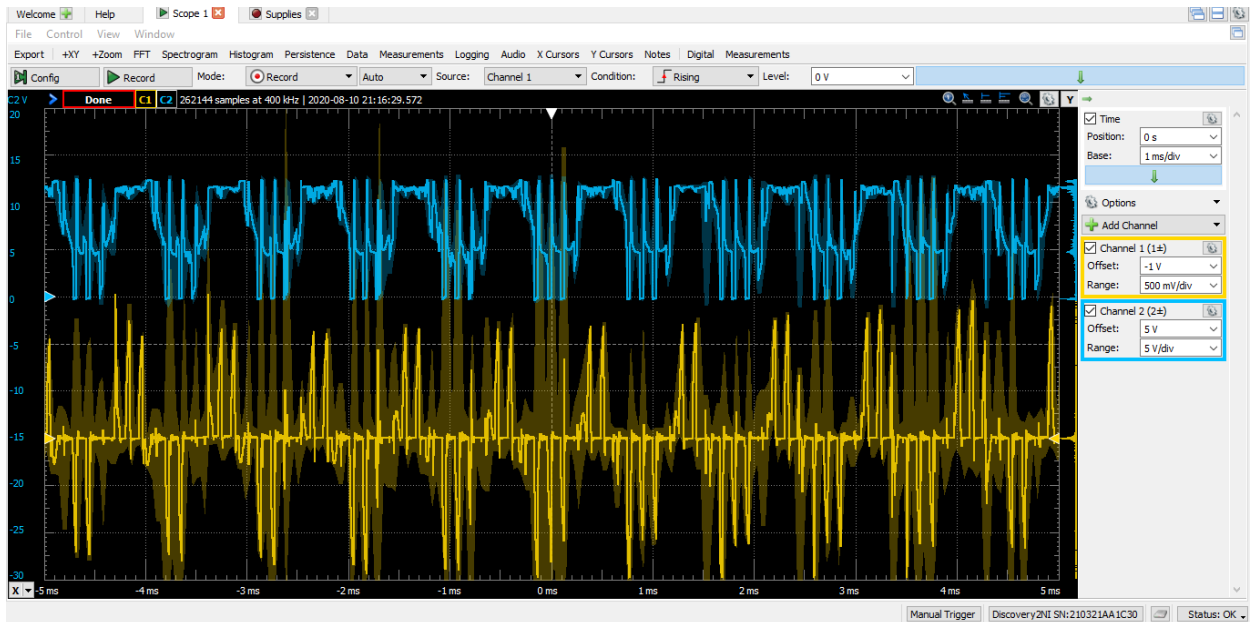
2. Снимање напонског и струјног сигнала једне фазе у празном ходу

Електрична шема и начин повезивања



Поступак извођења вежбе:

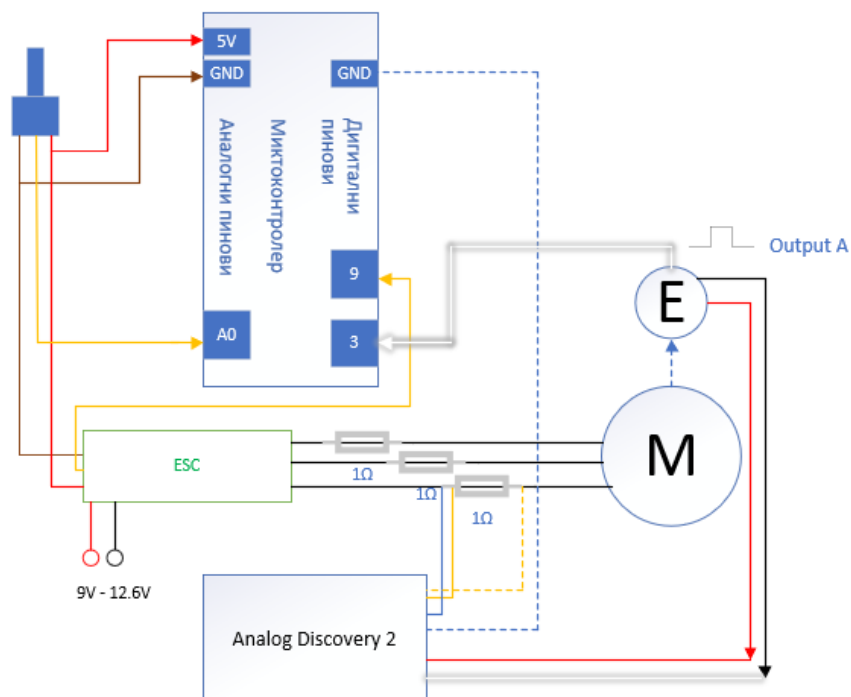
- 1) Повезати елементе са шеме
- 2) Употребити код за Arduino из Прилога 2
- 3) Покренути софтвер WaveForms (Scope)
- 4) Извршити следећа подешавања софтвера:
Trigger:
Source: Channel 1
Condition: Rising
Mode: Record
- 5) Задати брзину мотора помоћу потенциометра
- 6) Пустити мотор у рад
- 7) Кликом на  извршити снимање за неведене вредности
- 8) Сачувати добијене вредности и анализирати изглед добијених сигнала



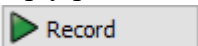
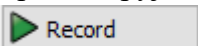
Изглед напонског и струјног сигнала за задату вредност брзине

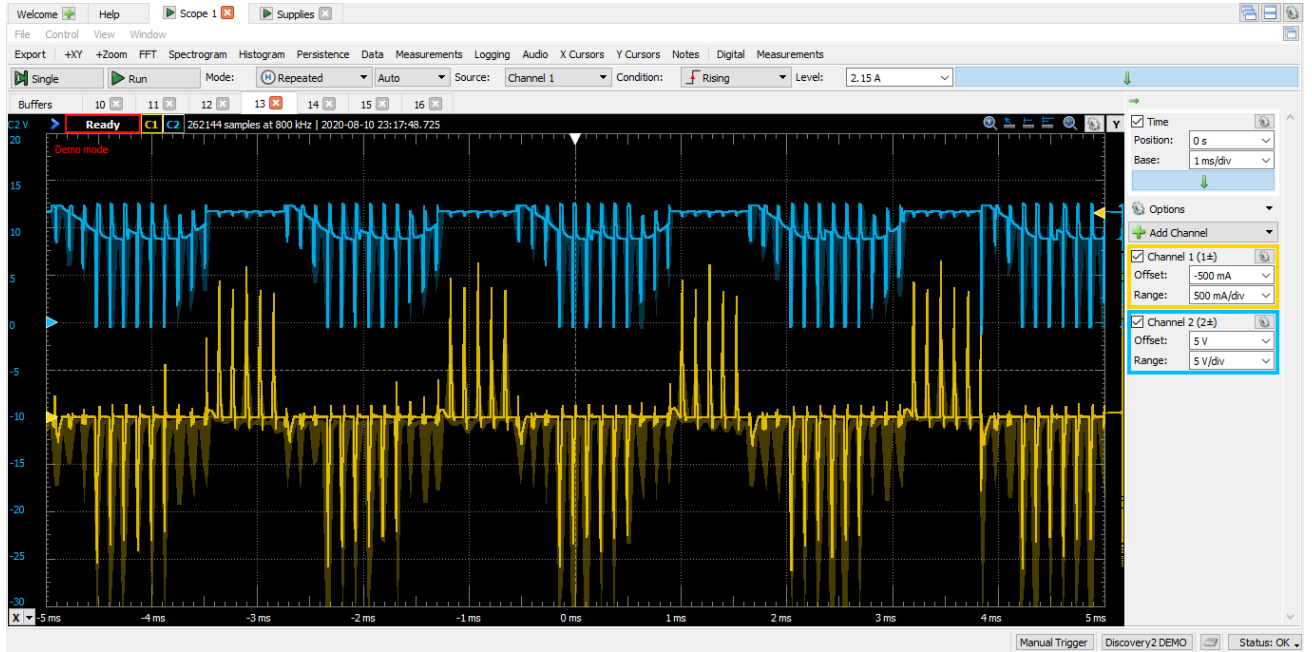
3. Снимање напонског и струјног сигнала једне фазе при оптерећењу

Електрична шема и начин повезивања

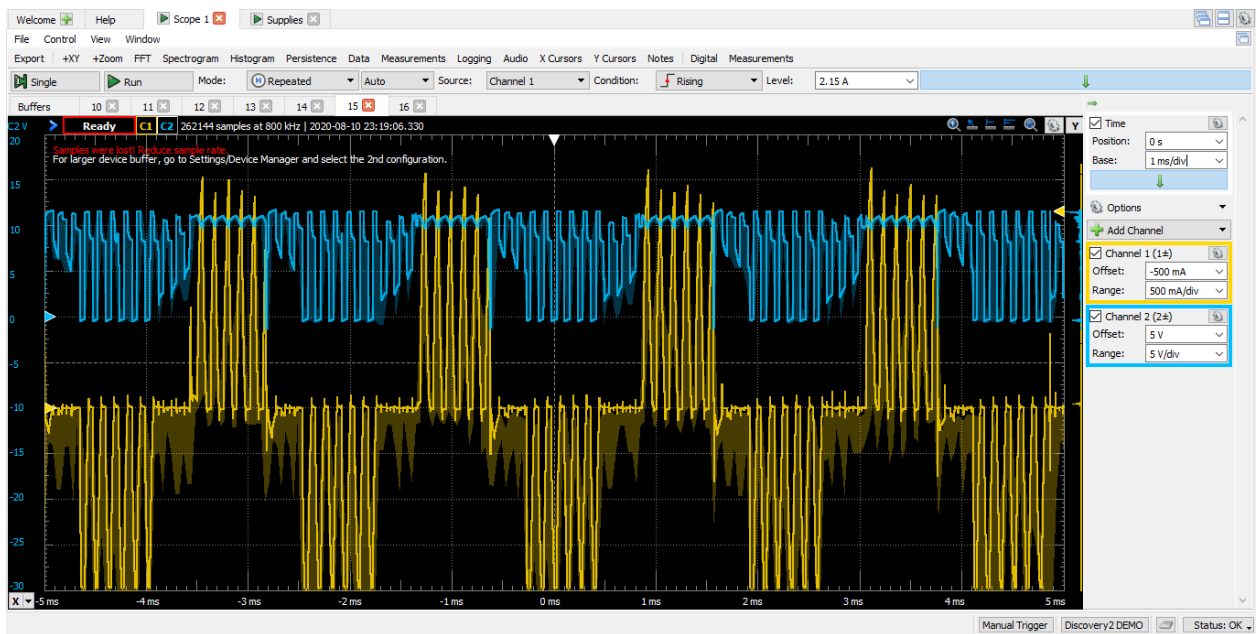


Поступак извођења вежбе:

- 1) Повезати елементе са шеме
- 2) Употребити код за Arduino из Прилога 3
- 3) Покренути софтвер WaveForms
- 4) Извршити следећа подешавања софтвера:
 - Trigger:
 - Source: Channel 1
 - Condition: Rising
 - Mode: Record
 - Supplies:
 - Positive supply (V+) on
 - Voltage: 5V
 - Negative supply (V-) on
 - Voltage: -500mV
- 5) Задати брзину мотора помоћу потенциометра
- 6) Пустити мотор у рад
- 7) Кликом на  извршити снимање сигнала
- 8) Сачувати добијене вредности сигнала мотора у празном ходу
- 9) Затим зауставити процес, задати жељено оптерећење и поново га покренути
- 10) Извршити мерење струје оптерећења помоћу амперметра
- 11) Кликом на  извршити снимање сигнала
- 12) Сачувати добијене вредности сигнала мотора при оптерећењу и упоредити их са вредностима сигнала добијеним у празном ходу



Облик сигнала у празном ходу



Облик сигнала при задатом оптерећењу

Прилози Arduino кода за реализацију лабораториских вежби**Прилог 1 (снимање напонског сигнала за све три фазе)**

```

#include <Servo.h>
Servo ESC;
int zad_vred;
float Vin;
float Vout = 0.0; // vrednost A0
void setup() {
  ESC.attach(9,1000,2000); // (pin, min sirina impulsa, max sirina impulsa u mikrosekundama)
  Serial.begin(9600);
}
void loop() {
  zad_vred = analogRead(A0); // očitava vrednost na potenciometru (između 0 i 1023)
  // skaliranje za korišćenje sa servo library (vrednost između 0 i 180)
  ESC.write(zad_vred = map(zad_vred, 0, 1023, 0, 180));
  Vin = analogRead(A0);
  Vout = (Vin * 5.0) / 1024.0;
  Serial.print("Referenca brzine: ");
  Serial.println(Vout); }

```

Прилог 2 (Снимање напонског и струјног сигнала једне фазе у празном ходу)

```

#include <Servo.h>
Servo ESC;
#define ENC_BR_IMP_O 1000
#define ENC_IN 3
int zad_vred=0;
int Vin_pot = 0;
float Vout = 0.0;
volatile long enc_br_imp = 0;
int interval = 1000;
long prethodnoMillis = 0;
long trenutnoMillis = 0;
int rpm = 0;
void setup() {
  ESC.attach(9,1000,2000);
  pinMode(ENC_IN, INPUT_PULLUP);
  Serial.begin(9600);
  attachInterrupt(digitalPinToInterrupt(ENC_IN), updateEncoder, FALLING);
  prethodnoMillis = millis();
}
void loop() {
  zad_vred = analogRead(A0);
  zad_vred = map(zad_vred, 0, 1023, 0, 180);
  ESC.write(zad_vred);
  trenutnoMillis = millis();
  if (trenutnoMillis - prethodnoMillis > interval) {
    prethodnoMillis = trenutnoMillis;
    rpm = (float)(enc_br_imp * 60 / ENC_BR_IMP_O);
    if (zad_vred > 0 || rpm > 0) {
      Serial.print("Napon:");
      Vin_pot = analogRead(A0);
      Vout = (Vin_pot * 5.0) / 1024.0;
    }
  }
}

```

```

Serial.print(Vout);
Serial.print("\t");
Serial.print(" SPEED: ");
Serial.print(rpm);
Serial.println(" RPM");
}
enc_br_imp = 0;
}
}
void updateEncoder()
{
enc_br_imp++;
}

```

Прилог 3 (Снимање напонског и студијног сигнала једне фазе при оптерећењу)

```

#include <Servo.h>
Servo ESC;
int zad_vred;
float Vin=0.0;
float Vout = 0.0;// vrednost A0
#define ENC_IN 3
#define ENC_BR_IMP_O 1000
volatile long encoder=0;
int Brzina=0;
int e_speed=0;
int e_speed_sum=0;
int e_speed_pre=0;
int pwm_pulse=0;
int pwm_pulse1=0;
int pwm_pulse2=0;
int interval = 10;
long prethodnoMillis = 0;
long trenutnoMillis = 0;

double kp=4.2555, ki=0.521999, kd=0;

void setup() {

ESC.attach(9,1000,2000); // (pin, min sirina impulsa, max sirima impulsa u mikrosekundama)
Serial.begin(9600);
pinMode(ENC_IN, INPUT_PULLUP);
attachInterrupt(digitalPinToInterrupt(ENC_IN), updateEncoder, FALLING);
}

void loop() {
Vin =(analogRead(A0));
zad_vred=map(analogRead(A0), 0, 1023, 0, 16030);

Vout= (Vin* 5.0) / 1024.0;
trenutnoMillis = millis();
if (trenutnoMillis - prethodnoMillis > interval) {
prethodnoMillis = trenutnoMillis;
Brzina = (float(encoder * 6000 / ENC_BR_IMP_O)); //proracun brzine u obr/min
encoder=0;
}
}

```

```
pwm_pulse1=map(analogRead(A0), 0, 1023, 0, 255);
ESC.write(map(pwm_pulse1,0,255,0,180));
if(abs(zad_vred - Brzina) > 0 & (Vin>0)) {

e_speed = zad_vred - Brzina;
pwm_pulse2 = e_speed*kp + e_speed_sum*ki + (e_speed - e_speed_pre)*kd;
e_speed_pre = e_speed; //cuvanje prethodne greske
e_speed_sum += e_speed; //zbir gresaka

pwm_pulse=map(pwm_pulse2,0, 16030,0, 255);
ESC.write(map(pwm_pulse,0,255,0,180));
// Serial.print("PID referenca ");
// Serial.println(float(pwm_pulse/51));
}
}

// Serial.print("Brzina: ");
Serial.print(Brzina);
Serial.print("\t");
// Serial.print("Zadata brzina: ");
Serial.println(zad_vred);
}
void updateEncoder()
{
encoder+=1; }
```