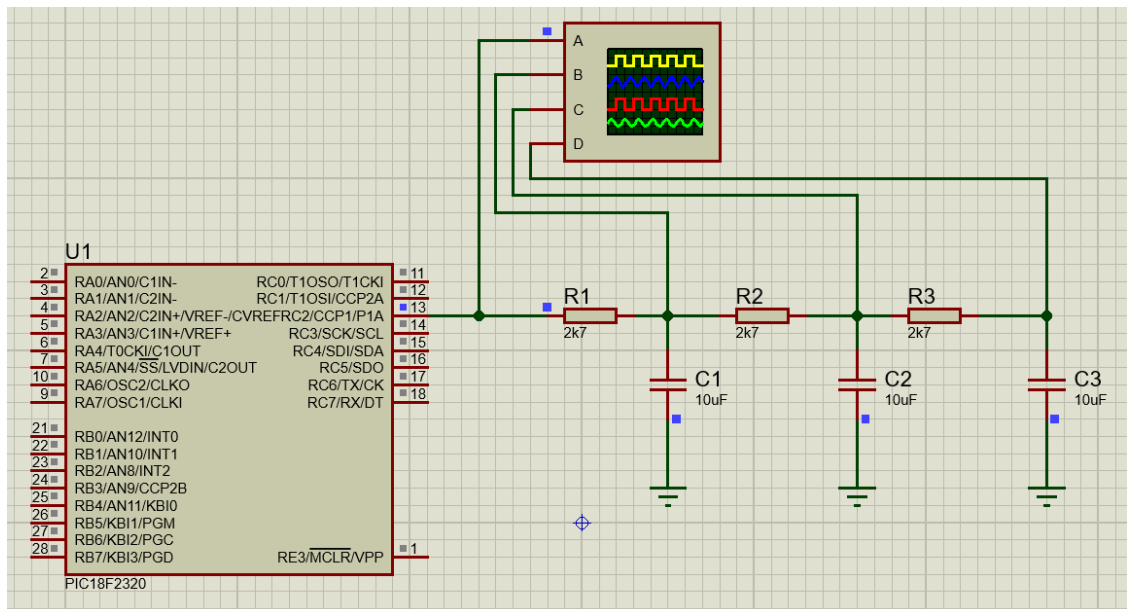


U zadatku su u sklopu globalne lookup tablice upisane izračunate vrijednosti signala u 25 trenutaka u periodi. Formula simulirane sinusoide je

$$\sin(i) = \sin\left(\frac{2 * \pi * i}{25}\right) * 25 + 25, i \in [0, 24]$$

U main funkciji inicijalizira se PWM uz PR2 vrijednost 49, kako su vrijednosti sinusa u intervalu [0, 50]. Uključuje se timer2 s preddjelilom 4 kako bi frekvencija naredbi bila 250kHz. Program ulazi u beskonačnu petlju. Unutra se postavlja se 10-bitna popunjenost kanala 1 ovisno o varijabli i . Pomoću nje se slijedno uzimaju vrijednosti sinusa iz lookup tablice. Varijabla i se povećava i počinje čekanje do iduće promjene. Iza petlje se isključuju PWM i timer2.



Slika 1. Proteus - shema projekta

Programski kod:

```
#include <p18f2320.h>
```

```
#include <timers.h>
```

```
#include <pwm.h>
```

```
#include <delays.h>
```

```
#pragma config WDT = OFF
```

```
#pragma config PWRT= ON
```

```
#pragma config MCLRE=OFF
```

```
#define N 25; // N i A = 25
```

```
unsigned int sinus[] = { 25, 31, 37, 42, 46, 48, 49, 49, 47, 44, 39, 34, 28, 21, 15, 10, 5, 2, 0, 0, 1, 3, 7, 12, 18 }; // Lookup tablica vrijednosti sinusa
```

```
unsigned int i = 0; // Vrijednost popunjenosti
```

```
void main(void) {
```

```
OpenPWM1( 49 ); // PR2 = 49 = 2*A - 1
```

```
OpenTimer2(TIMER_INT_OFF & T2_PS_1_4); // Ukljuci timer2, preddjelilo = 4
```

```
while(1) {
```

```
    SetDCPWM1(sinus[i]<<2); // Postavi popunjenost(velicina 10b !)
```

```
    i = (i + 1)%N; // Pomakni indeks u tablici popunjenosti (cirkularno)
```

```
    Delay100TCYx(96); // cekanje 9600 taktova
```

```
    Delay10TCYx(5); // cekanje 50 taktova
```

```
}
```

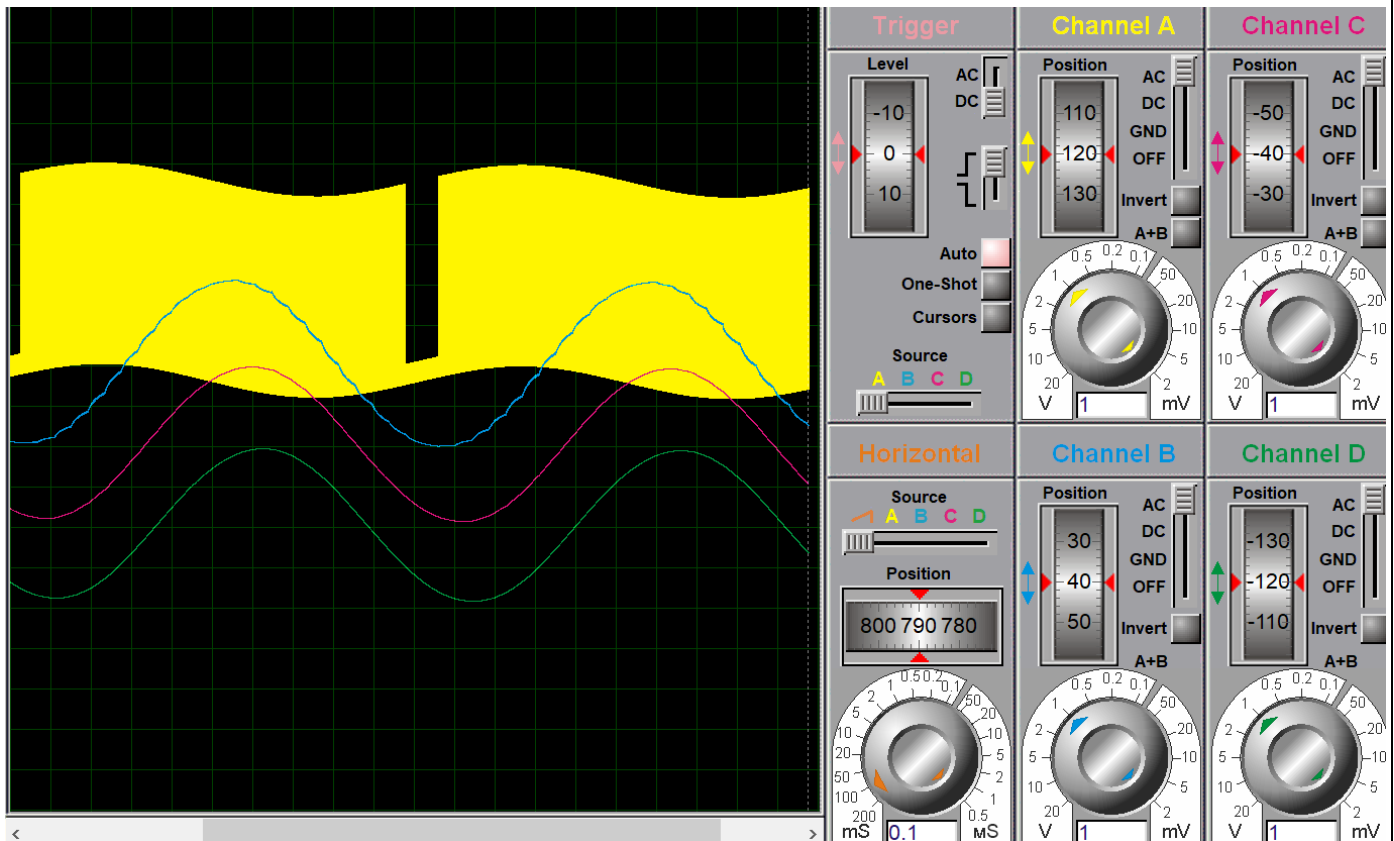
```
ClosePWM1(); // Iskljuci PWM
```

```
CloseTimer2(); // Iskljuci Timer2
```

```
}
```

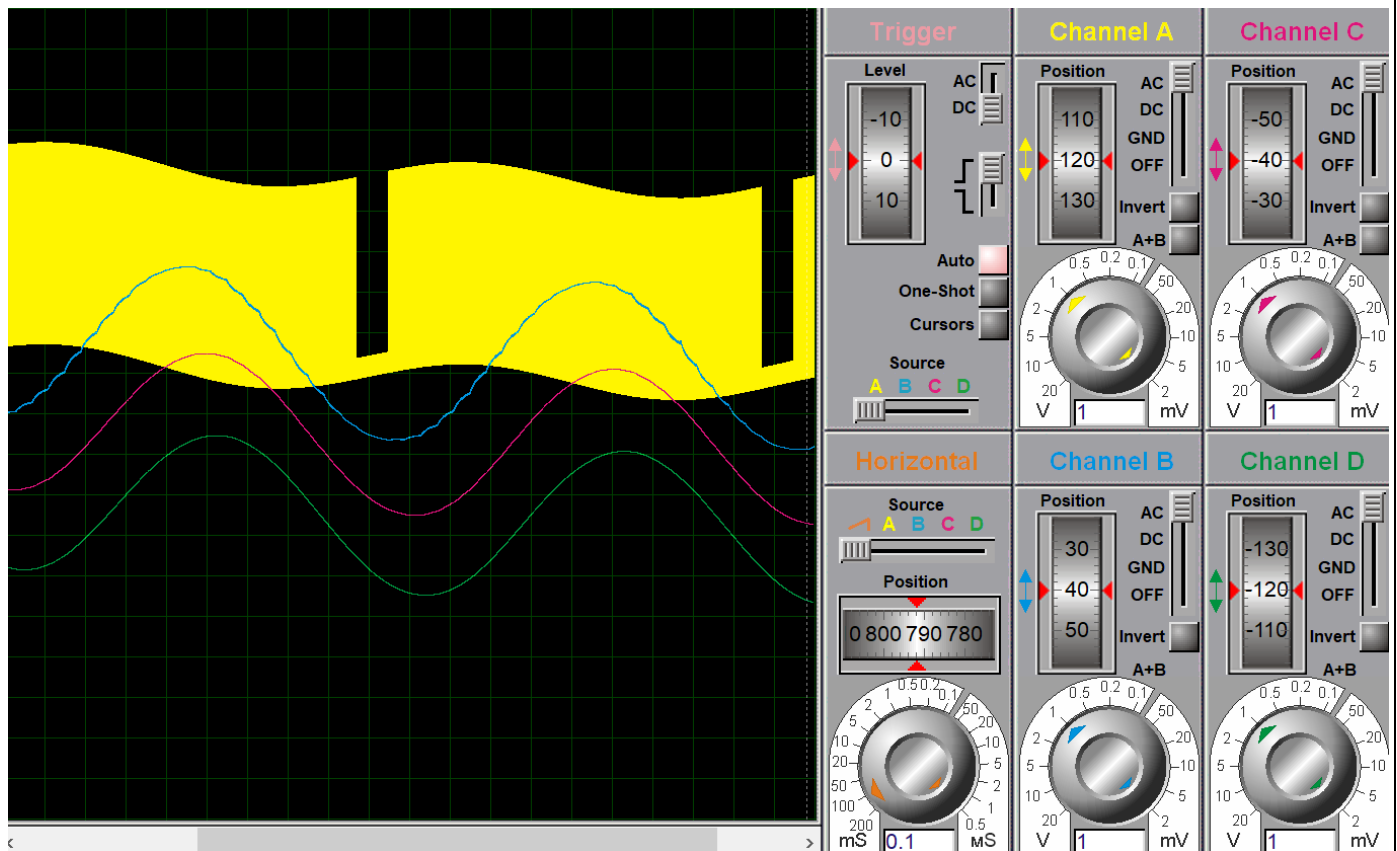
1. ZADATAK

a) Iscrtava se sinusoida frekvencije $f_{\sin} = 1\text{Hz}$. Slijedi da je period sinusoide $T_{\sin} = \frac{1}{1\text{Hz}} = 1\text{s}$. Uzima se uzorkovanje od 25 točki, pa trajanje svakog koraka (svake od 25 razina) treba trajati $\frac{1\text{s}}{25} = 40\text{ms}$. Mikrokontroler s frekvencijom naredbi 250kHz ima period naredbe dužine $\frac{1}{250\text{kHz}} = 4\mu\text{s}$. Za čekanje od 40ms potrebno je $\frac{40\text{ms}}{4\mu\text{s}} = 10000$ taktova. Ako se taj broj upotrijebi u funkciji Delay (`Delay100TCYx(100);`) sinusoida će imati predug period. Uz čekanje 10000 taktova dobiva se sinusoida prikazana slikom 2.



Slika 2. Sinusoida dobivena uz kašnjenje 10000 taktova

b) Sinusoida traje predugo, što znači da se kašnjenje treba skratiti. U svakoj petlji instrukcije troše vrijeme, pa kašnjenje treba trajati toliko manje. Pomoću postavljanja breakpointa otkriveno je vrijeme jednog prolaska kroz petlju: 41.396ms. Dobiva se sinusoida frekvencije $41.396\text{ms} * 25 = 1.0349\text{Hz}$. Točnih 40ms postignuto je uz kašnjenje 9650 taktova. Slika 3 prikazuje sinusoidu frekvencije 1Hz.



Slika 3. Sinusoida dobivena uz kašnjenje 9650 taktova

2. ZADATAK

a) Frekvencija izlaznog PWM signala:

$$\text{period PWM} = [(PR2) + 1] * 4 * T_{osc} * TMR2 \text{ preddjelilo}$$

$$\text{period PWM} = [49 + 1] * 4 * 10^{-6}\text{s} * 4 = 0.8\text{ms}$$

$$\text{frekvencija PWM} = 1 / \text{period PWM} = 1 / 0.8\text{ms} = 1250\text{Hz}$$

b) Razlučivost signala:

$$R = \log_2 \frac{\text{period PWM}}{T_{osc} * TMR2 \text{ preddjelilo}}$$

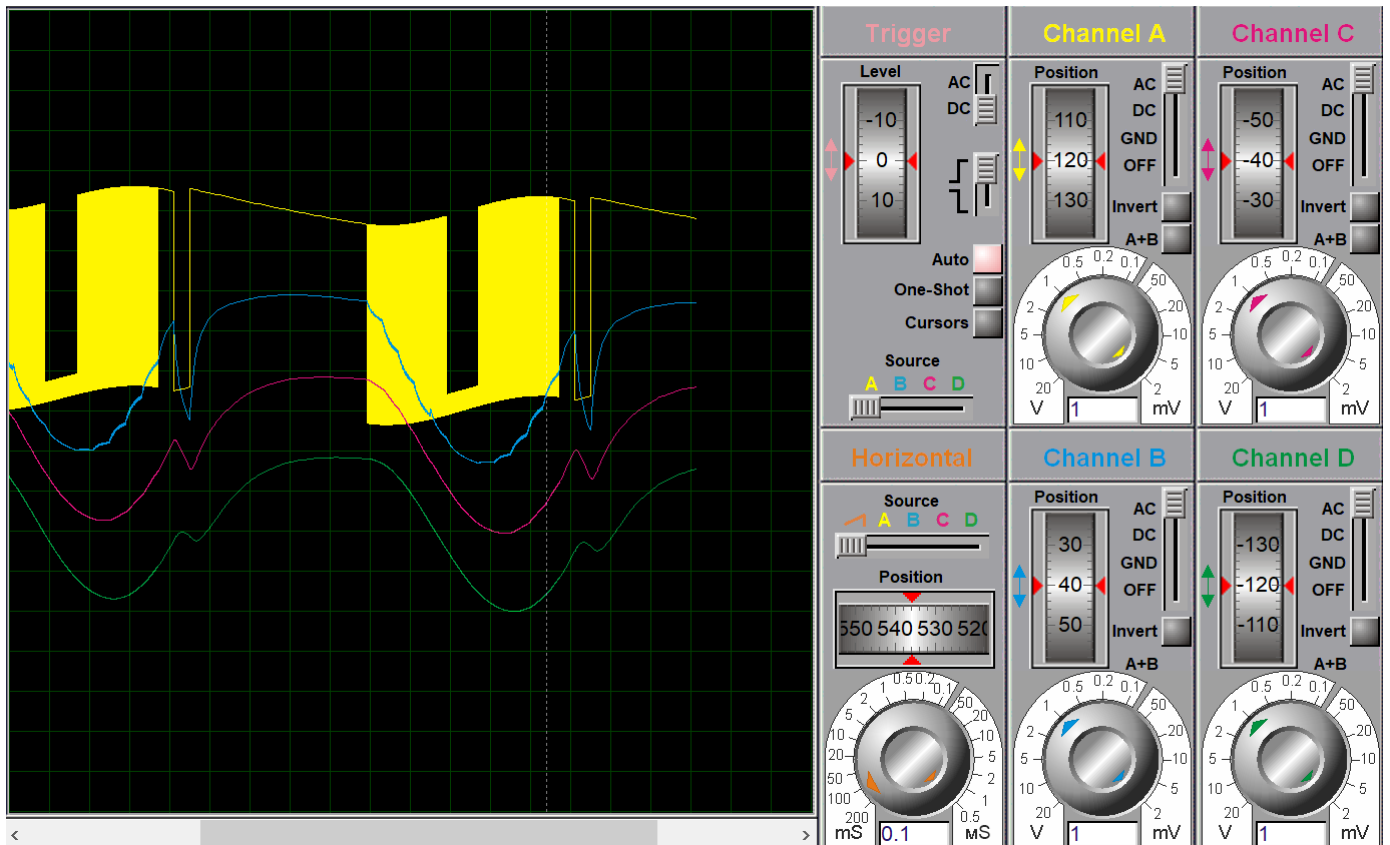
$$R = \log_2 \frac{1.6 * 10^{-3}\text{s}}{10^{-6} * 4} = 7.6439 = 8 \text{ bitova}$$

Rezolucija od 8 bitova je dovoljna za prikaz željene sinusoide.

c) Ako je razlučivost manja od 10 bita razlučivost ovisi o registrima CCPR1L (8 viših bitova) i CCP1CON (2 najniža bita u bitovima registra 5 i 4).

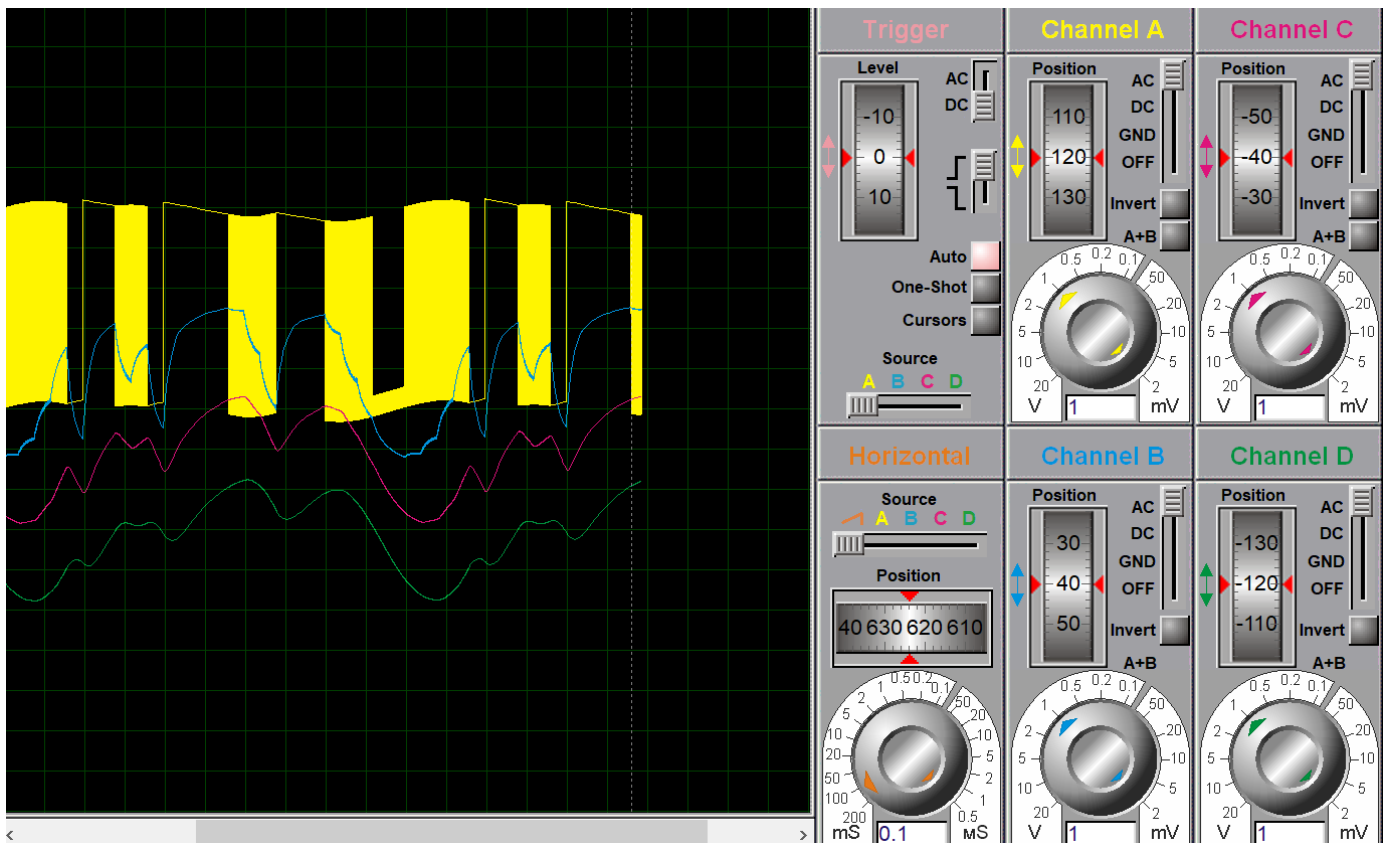
d) Kako je u ovom slučaju vrijednost PR2 jednaka 49, u CCPR1L su dopuštene vrijednosti od 0 do 50. Ako se granica prijeđe više se ne dobiva traženi oblik signala. Prvi primjer prikazan slikom

4 upisivao je prethodne vrijednosti pomnožene sa 2. Dio koji čini dol točnog je oblika, što je moguće jer su te vrijednosti od [0 – 25] prešle u [0 – 50].



Slika 4. Sinusoida s vrijednostima CCPR1L uvećanima za 2

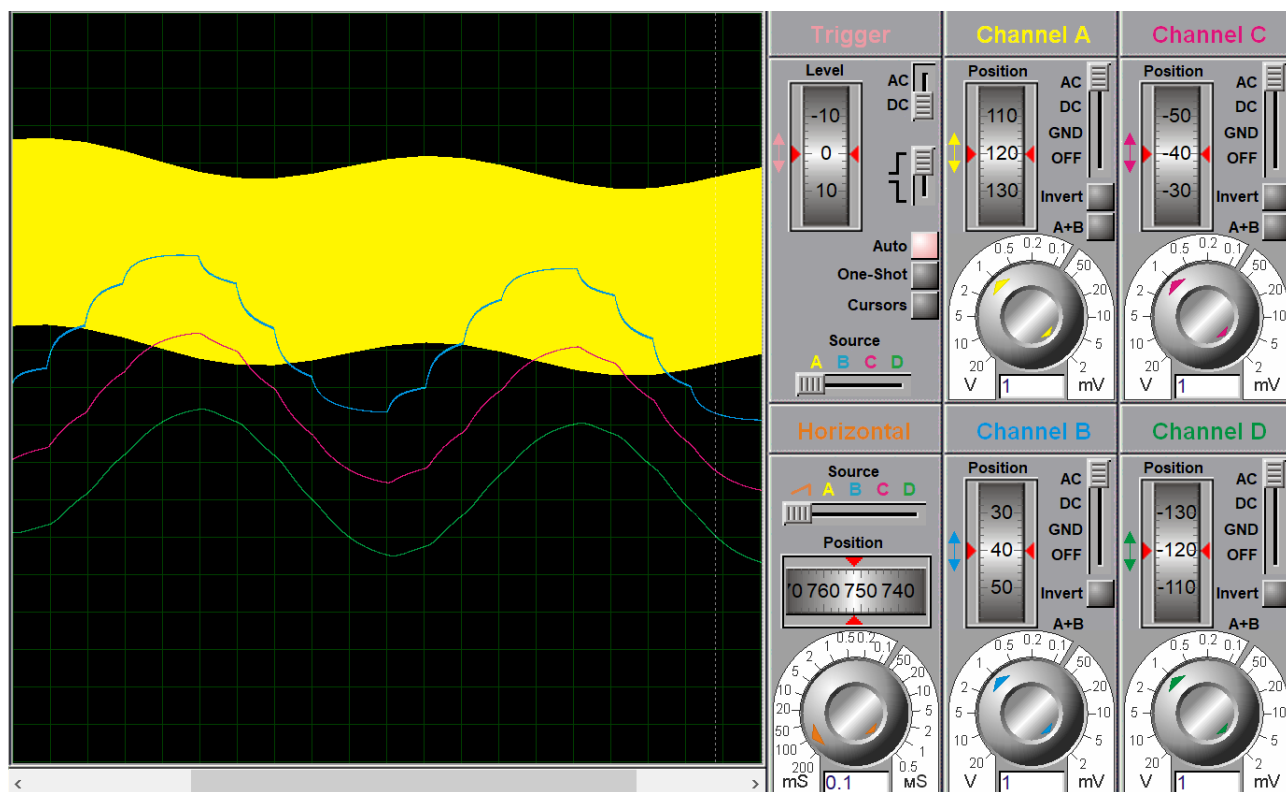
Slika 5 prikazuje slučaj kada su vrijednosti uvećane za 4. Sada je oblik još više neprepoznatljiv, sa samo najnižom polovicom dola u točnom obliku.



Slika 5. Sinusoida s vrijednostima CCPR1L uvećanima za 4

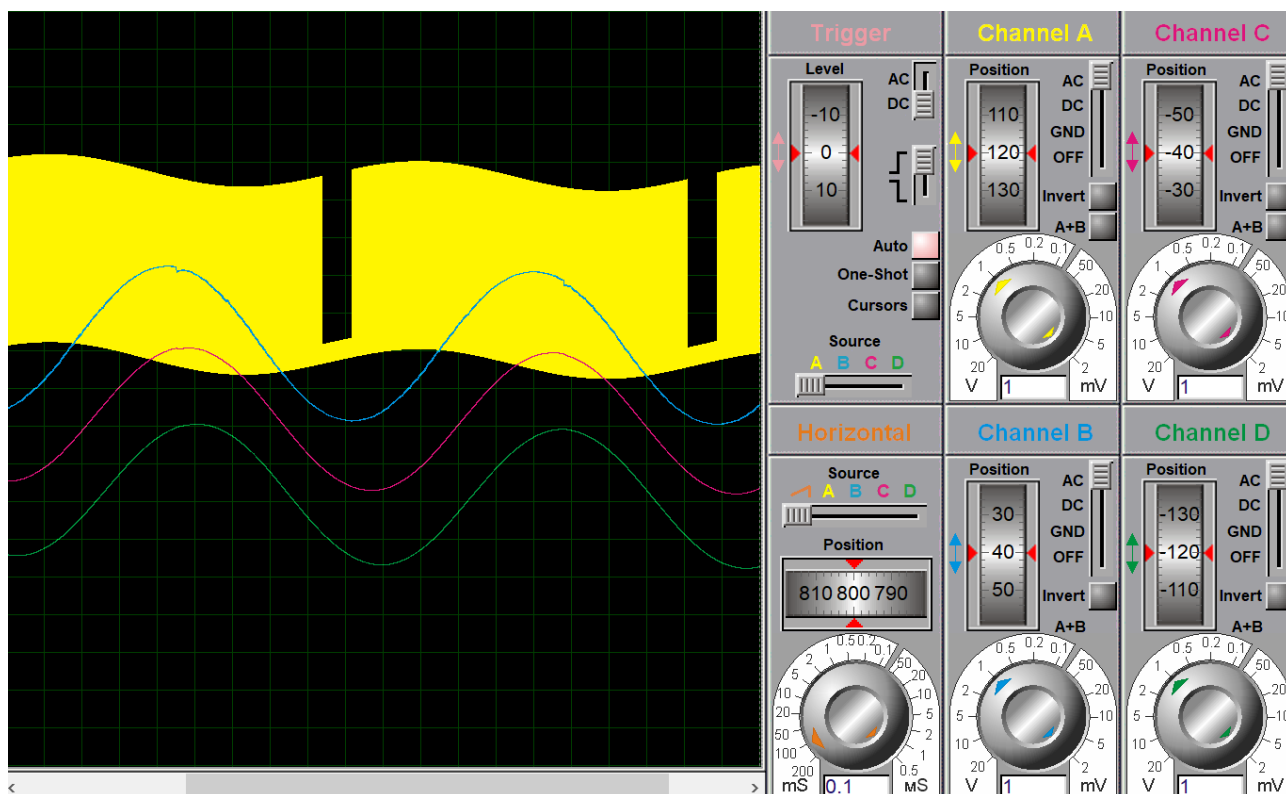
3. ZADATAK

a) U slučaju kada je vrijednost varijable N 25 sinusoida (prikazana slikom 3) je glatka, no vide se razlike između razina. Ako se broj razina smanji signal će se izobličiti. Kako bi se zadržala frekvencija, uz N se mijenja i vrijeme kašnjenja. Ako se N smanji na 10 sinusoida je još uvijek vidljiva, ali nije toliko glatka. Ovaj slučaj je prikazan slikom 6.



Slika 6. Sinusoida sa vrijednosti N = 10

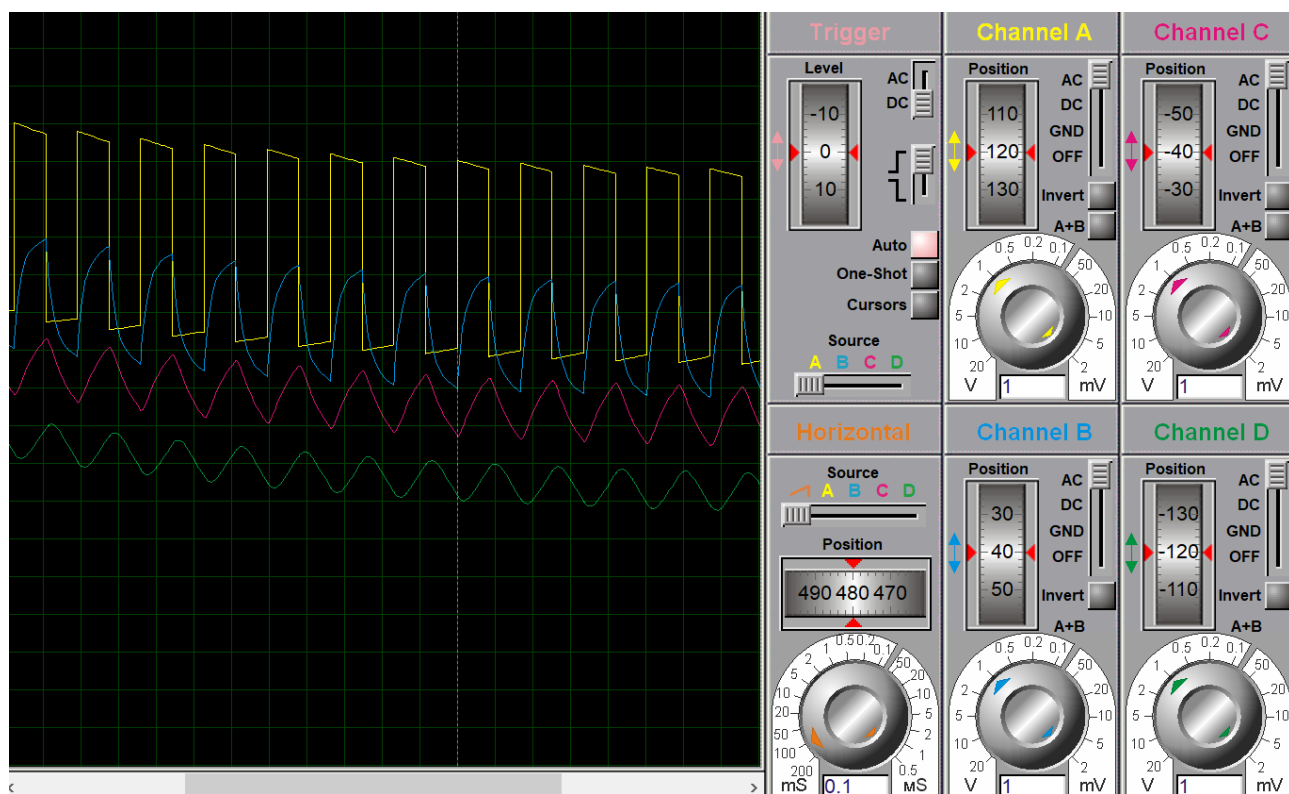
Što više razina postoji, sinusoida će biti glađa i slijedni koraci manje primjetni. Slika 7 prikazuje sinusoidu s brojem razina N 50. Signal je glatki i gotovo se ne raspoznaju trenutci promjene razine.



Slika 7. Sinusoida sa vrijednosti N = 50

b) Smanjenje frekvencije izlaznog PWM signala dovodi do izobličenja signala. Prestaje biti glatka i gubi izgled sinusoide. Izlazna frekvencija se smanjuje (izlazni period se povećava) povećanjem

perioda oscilatora. To se postiže smanjivanjem frekvencije mikrokontrolera. Rezolucija ostaje ista jer se promijenjena vrijednost poništava u brojniku i nazivniku formule. Prijelazne pojave dolaze do izražaja približavanjem sinusoide granici od 6Hz. Za frekvenciju od 5.95Hz, vrijednost na granici prigušenosti, potrebna je frekvencija mikrokontrolera 4.76kHz. Uz ovu vrijednost signal rezultira oblikom sa slike 8. Do izražaja dolaze RC prijelazne pojave.



Slika 8. Sinusoida sa vrijednosti $f_{osc} = 4.76\text{kHz}$