



M Ű E G Y E T E M 1 7 8 2

Budapesti Műszaki és Gazdaságtudományi Egyetem
Mechatronika, Optika és Gépészeti Informatika Tanszék

Digitális szervohajtások - (BMEGEMIMMDS)
Első négy mérés

Név: Szőke Kálmán Benjamin
Neptun: SLZ0UE
Dátum: 2015. március 27.

Tartalomjegyzék

1. Mérési feladatok	1
1.1. A PCI-1720 D/A kártya használata – Motion control/Exercise 1 menüpont alatt található laboratóriumi mérési gyakorlat	1
1.2. Valós idejű óra használata a PCI 1720 D/A kártyával – Motion control/Exercise 2 menüpont alatt található laboratóriumi mérési gyakorlat	1
1.2.1. Egy szinuszos feszültség-idő függvény megjelenítése a PCI 1720 A/D kártya kimenetén	1
1.3. A PCI 1784 számláló használata – Motion control/Exercise 3 menüpont alatt található laboratóriumi mérési gyakorlat	2
1.4. Nyílt szabályozó kör mérése – Motion control/Exercise 4 menüpont alatt található laboratóriumi mérési gyakorlat	4
1.4.1. Konstans nyomaték kiadása	4
1.4.2. Konstans nyomaték kiadása és a fordulatszám szűrése	6
1.4.3. A motor válasza szinuszos nyomatéokra és a szervoerősítő offset kompenzációja	14
1.4.4. Digitális szűrő	19
1.4.5. Tapadási és csúszási súrlódás	22
2. Melléklet	26

Ábrák jegyzéke

1. Egy szinuszos feszültség-idő függvény megjelenítése a PCI 1720 A/D kártya kimenetén	2
2. A PCI 1784 számláló használata	3
3. Konstans nyomaték kiadása, ResultData.Torque = 0.1;	5
4. Konstans nyomaték kiadása, ResultData.Torque = 0.5;	5
5. Konstans nyomaték kiadása, ResultData.Torque = 1.0;	6
6. Konstans nyomaték kiadása és a fordulatszám szűrése, ResultData.Torque = 0.1;	8
7. Konstans nyomaték kiadása és a fordulatszám szűrése, ResultData.Torque = 0.5;	9
8. Konstans nyomaték kiadása és a fordulatszám szűrése, ResultData.Torque = 1.0;	9
9. A szervoerősítő offset kompenzációja, Offset = 0.0	15
10. A szervoerősítő offset kompenzációja, Offset = -0.22	16
11. A szervoerősítő offset kompenzációja, Offset = -0.25	17
12. A szervoerősítő offset kompenzációja, Offset = -0.28	18
13. Digitális szűrő, Period = 0.1	20
14. Digitális szűrő, Period = 0.4	20
15. Tapadási és csúszási súrlódás, nyomaték = 0.2 mNm, periódus = 1.0 s, Offset = -0.21	23

1. Mérési feladatok

1.1. A PCI-1720 D/A kártya használata – Motion control/Exercise 1 menüpont alatt található laboratóriumi mérési gyakorlat

A feladat a kártya 3-as csatornájának 5 Volt értékre való beállítása. Tulajdonképpen ez az 5 Volt teszi lehetővé a szervohajtás működését. Szükséges lépések:

- Hasonlítsuk össze az Adventech specifikus változókat a keretrendszer programban és a mintaprogramban. Találjuk meg a megváltozott változókat.
- Állítsuk be a kártyát (vágjuk ki és illesszük be a 3. és 4. lépést a mintaprogramból).
- Módosítsuk a megváltozott változókat.
- Távolítsuk el az érdektelen részeket printf(); és getch(); utasításokat, amelyek szöveg kiírására és karakter beolvasásra szolgálnak).
- Állítsuk be a (kimeneti) feszültséget 5 Voltra a 3-as csatornán.

```
// A mérés dátuma és időpontja: 2015. 03. 26. 17:08 CET
// IP Address (IPv4): 86.59.219.9
motorDA = 3;
new_voltage = 5.0;
dwErrCde = DRV_DeviceOpen(lDevNumDA, &lDriverHandleDA);
if (dwErrCde != SUCCESS)
{
    ErrorHandler(dwErrCde);
    exit(1);
}
tAOVoltageOut.chan = motorDA;
tAOVoltageOut.OutputValue = new_voltage;
dwErrCde = DRV_AOVoltageOut(lDriverHandleDA, &tAOVoltageOut);
if (dwErrCde != SUCCESS)
{
    ErrorStop(&lDriverHandleDA, dwErrCde);
    return;
}
```

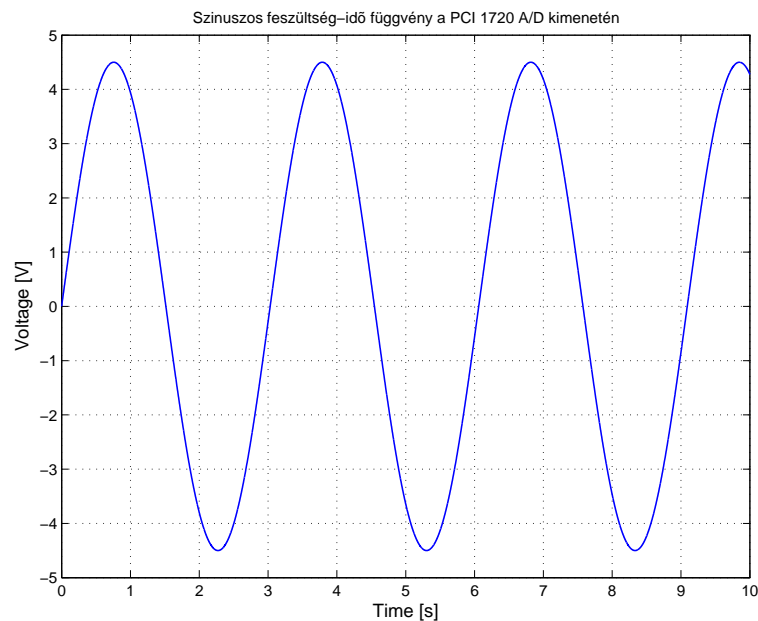
1.2. Valós idejű óra használata a PCI 1720 D/A kártyával – Motion control/Exercise 2 menüpont alatt található laboratóriumi mérési gyakorlat

1.2.1. Egy szinuszos feszültség-idő függvény megjelenítése a PCI 1720 A/D kártya kimenetén

Meg kell határozni az összefüggést a mintavételi idő és a szinusz hullám között (3 Voltos amplitúdóval 1 HZ-es frekvencián). A mintavételezés 0-tól kezdődik és a Current-Time változón keresztül érhető el a programból. Ezt a változót a time array[tickCount]

változóval lehet elérni, ami 100 nanoszekundumos egységű adatot ad vissza. Szükséges lépések:

- Áttanulmányozni a megadott keret programot;
- Beírni a hiányzó kódokat;
 - Deklarálni a paramétereket és változókat;
 - $\text{Out} = \text{Amplitude} * \sin(\text{Angular Frequency} * \text{time})$ (megjegyzés: ez nem egy C kód, és a time-ot a TimerHandler függvénnyel kell beolvasni).



1. ábra. Egy szinuszos feszültség-idő függvény megjelenítése a PCI 1720 A/D kártya kimenetén

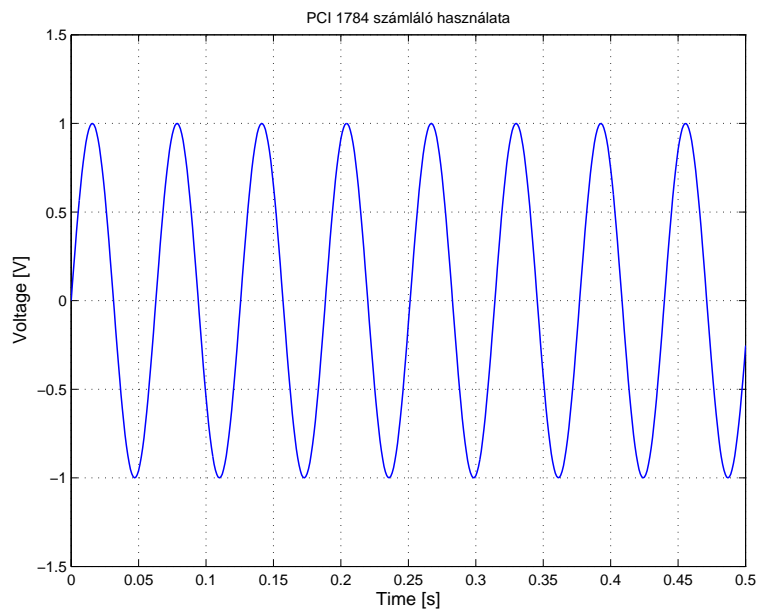
```
// A mérés dátuma és időpontja: 2015. 03. 26. 17:19 CET
// IP Address (IPv4): 86.59.219.9
//Block 1. Enter your variable declarations here:
float sin_amp = 4.5;
float sin_freq = 2.0*PI*0.33;

//Block 2. Enter your code here:
new_voltage = sin_amp * sin(sin_freq * (time_array[tickCount] / 10000000.0));
```

1.3. A PCI 1784 számláló használata – Motion control/Exercise 3 menüpont alatt található laboratóriumi mérési gyakorlat

A feladat a számláló értékének beolvasása a PCI 1784-es kártya 3-as csatornájáról. Szükséges lépések:

- Kártya inicializálása.
- Számláló értékek visszaállítása.
- A számláló művelet elindítása.
- A számláló érték beolvasása a művelet elején.
- A számláló értékének minden egyes órajelre való beolvasása.



2. ábra. A PCI 1784 számláló használata

```
// A mérés dátuma és időpontja: 2015. 03. 26. 17:31 CET
// IP Address (IPv4): 86.59.219.9
//Initialization:
wChannelCounter = 3;
//Step 1: Open device
dwErrCde = DRV_DeviceOpen(lDevNumCounter, &lDriverHandleCounter);
if (dwErrCde != SUCCESS)
{
ErrorHandler(dwErrCde);
exit(1);
}
// Step 2: Reset counter by DRV_CounterReset
dwErrCde = DRV_CounterReset(lDriverHandleCounter, wChannelCounter);
if (dwErrCde != SUCCESS)
{
ErrorHandler(dwErrCde);
exit(1);
}
// Step 3: Start counter operation by DRV_CounterEventStart
```

```

tCounterEventStart.counter = wChannelCounter;
dwErrCde = DRV_CounterEventStart(lDriverHandleCounter, &tCounterEventStart);
if (dwErrCde != SUCCESS)
{
ErrorHandler(dwErrCde);
exit(1);
}
// Step 4: Read counter value by DRV_CounterEventRead
tCounterEventRead.counter = wChannelCounter;
tCounterEventRead.overflow = &wOverflow;
tCounterEventRead.count = &dwReading;
dwErrCde = DRV_CounterEventRead(lDriverHandleCounter, &tCounterEventRead);
if (dwErrCde != SUCCESS)
{
ErrorStop(&lDriverHandleCounter, dwErrCde);
return;
}

//Loop:
dwErrCde = DRV_CounterEventRead(lDriverHandleCounter, &tCounterEventRead);
if (dwErrCde != SUCCESS)
{
ErrorStop(&lDriverHandleCounter, dwErrCde);
return;
}

```

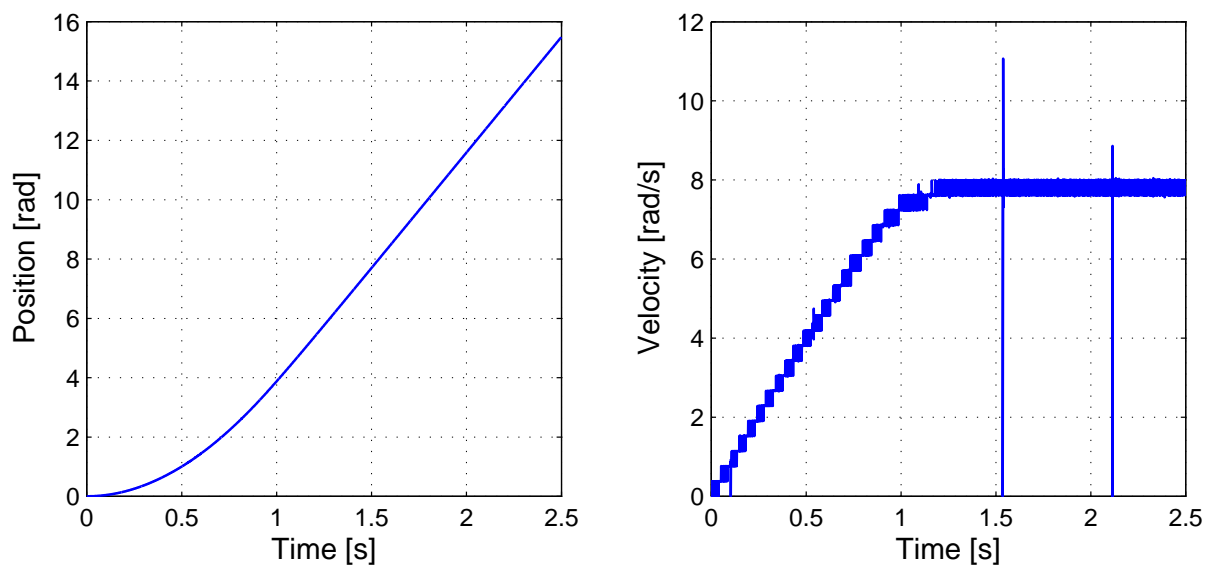
1.4. Nyílt szabályozó kör mérése – Motion control/Exercise 4 menüpont alatt található laboratóriumi mérési gyakorlat

1.4.1. Konstans nyomaték kiadása

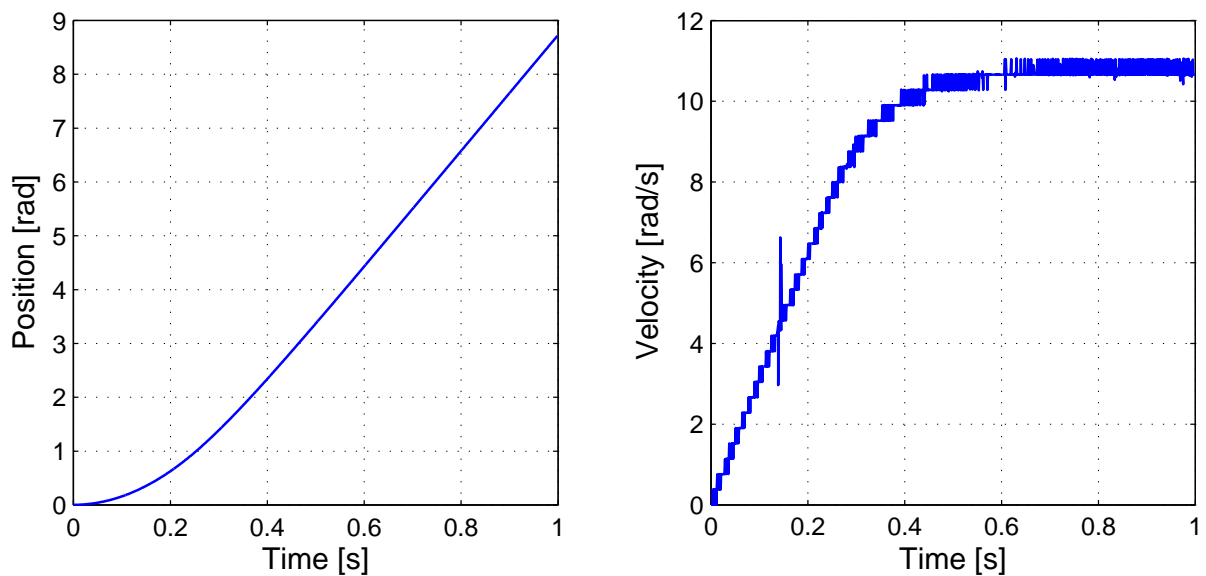
Az első feladatban az elektromos nyomaték értéke:

- ResultData.Torque = 0.1;
- ResultData.Torque = 0.5;
- ResultData.Torque = 1.0;

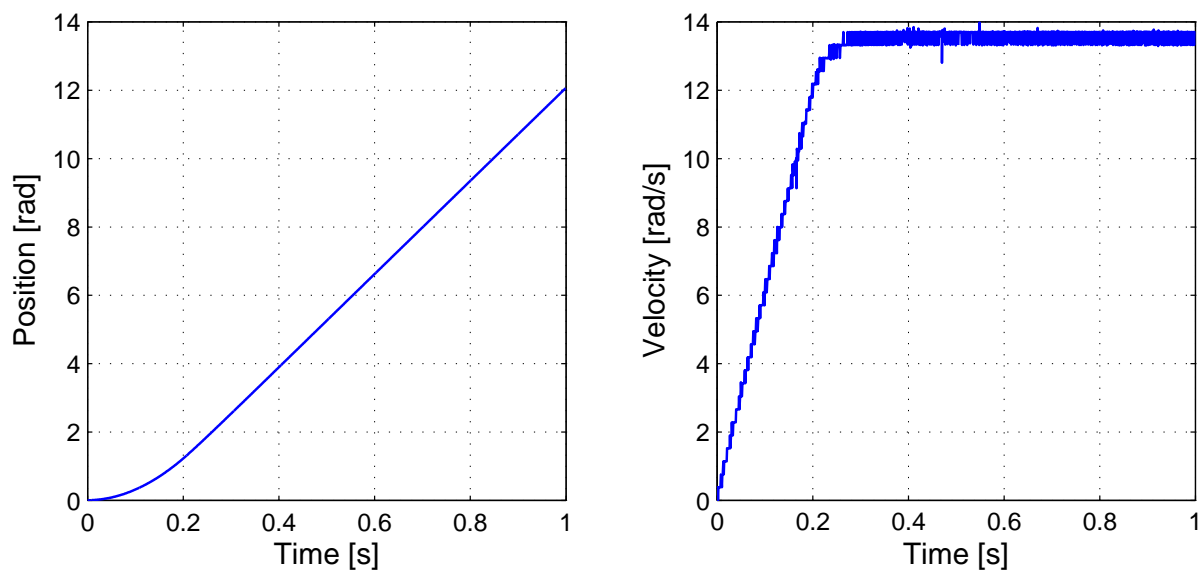
Ezúton a különböző motor-változók különböző nyomatékjel lépés-változásaira kapott lépés válaszai kaphatók meg. Az eredmény fájlok, amelyek a mérés végén letölthetők, Matlabban kiszámíthatók. Erre a feladatra már van egy program. A program futtatásával az eredményeket megkapjuk tengelysebesség-idő, feszültség-idő és pozíció-idő diagramokon. Ezeket a diagramokat .jpg formátumban lehet elmenteni a további dokumentáció érdekében.



3. ábra. Konstans nyomaték kiadása, ResultData.Torque = 0.1;



4. ábra. Konstans nyomaték kiadása, ResultData.Torque = 0.5;



5. ábra. Konstans nyomaték kiadása, ResultData.Torque = 1.0;

```
// A mérés dátuma és időpontja: 2015. 03. 26. 17:48 CET
// IP Address (IPv4): 86.59.185.137
//Task 1. Open loop:
//Measurement length in milliseconds: 2000
//Torque: 0.2 Nm
//The state variable names:
//1. time (given)
//2. position (given)
//3. velocity (given)
//4. torque (given)
//The code:
ResultData.Torque = 0.1;
//ResultData.Torque = 0.5;
//ResultData.Torque = 1.0;
```

1.4.2. Konstans nyomaték kiadása és a fordulatszám szűrése

Az előző feladathoz hasonlóan a nyomaték legyen a következő:

- ResultData.Torque = 0.1;
- ResultData.Torque = 0.5;
- ResultData.Torque = 1.0;

Alkalmazza az leírt digitális szűrőket. Legyen a töréspont helye:

$$T_c = 0.0032$$

$$T_c = 0.032$$


```

function [ Ad,Bd ] = IIR_filter_ord3( Ts, Tc )
%Ts = 0.001;
%Tc = 1/(2*pi*1/(2*Ts)/10);
A = [0, 1, 0;
      0, 0, 1;
      -1/Tc^3, -3/Tc^2, -3/Tc];
B = [0;0;1/Tc^3];
[ Ad,Bd ] = c2d(A,B,Ts);

for i = 1:3
    for j = 1:3
        disp(['float ad',num2str(i),num2str(j),' = ', num2str(Ad(i,j)),';']);
    end
end

for i = 1:3
    disp(['float bd',num2str(i),' = ', num2str(Bd(i)),';']);
end

end

```

```

function [ Ad,Bd ] = IIR_filter_ord3_Bessel( Ts, Tc )
%Ts = 0.001;
%Tc = 1/(2*pi*1/(2*Ts)/10);
A = [0, 1, 0;
      0, 0, 1;
      -15/Tc^3, -15/Tc^2, -6/Tc];
B = [0;0;15/Tc^3];
[ Ad,Bd ] = c2d(A,B,Ts);

for i = 1:3
    for j = 1:3
        disp(['float ad',num2str(i),num2str(j),' = ', num2str(Ad(i,j)),';'])
    end
end

for i = 1:3
    disp(['float bd',num2str(i),' = ', num2str(Bd(i)),';']);
end

end

```

```

function [ Ad,Bd ] = IIR_filter_ord5_Bessel( Ts, Tc )
%Ts = 0.001;
%Tc = 1/(2*pi*1/(2*Ts)/10);
A = [0, 1, 0, 0, 0;
      0, 0, 1, 0, 0;
      0, 0, 0, 1, 0;
      0, 0, 0, 0, 1;
      -945/Tc^5, -945/Tc^4, -420/Tc^3, -105/Tc^2, -15/Tc];
B = [0;0;0;0;945/Tc^5];
[Ad,Bd] = c2d(A,B,Ts);

```

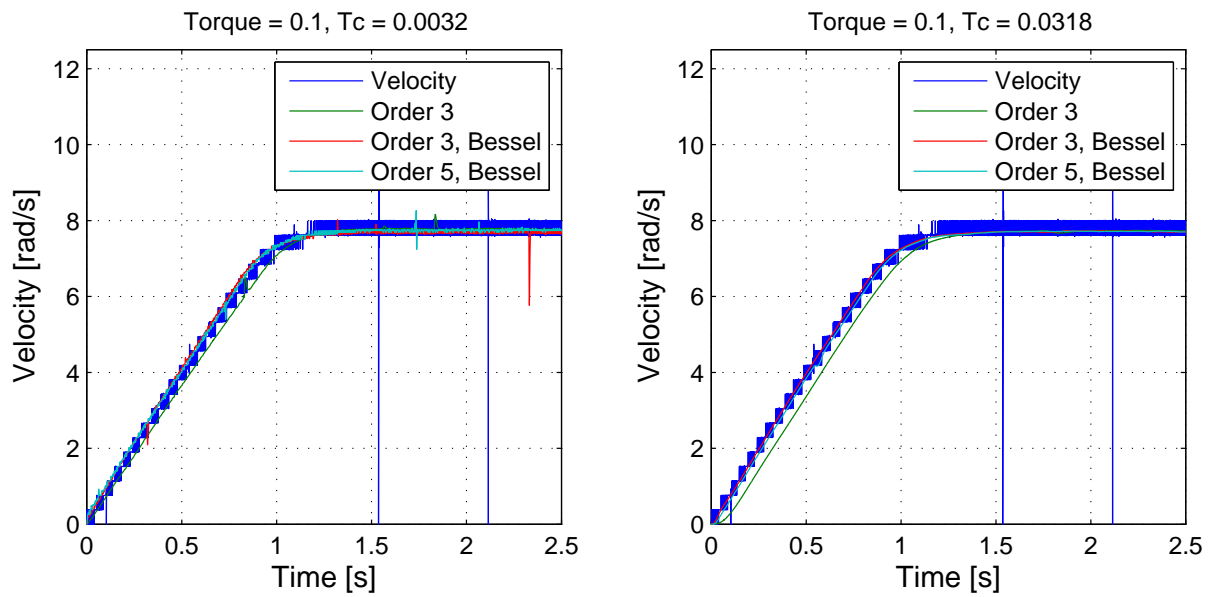
```

for i = 1:5
    for j = 1:5
        disp(['float ad', num2str(i), num2str(j), ' = ', num2str(Ad(i, j)), ';'])
    end
end

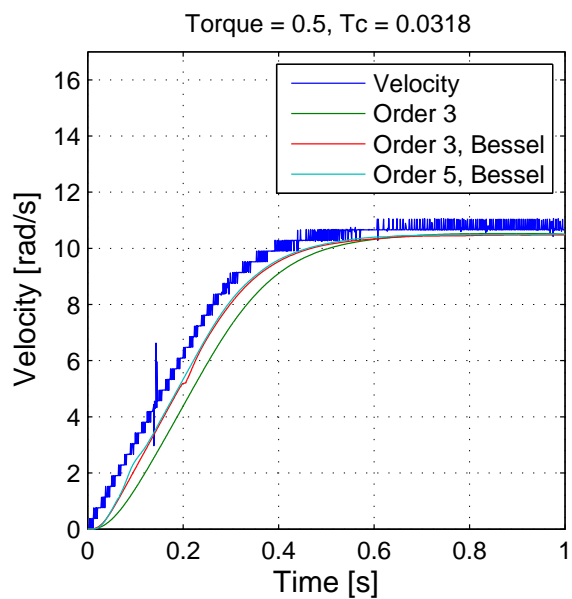
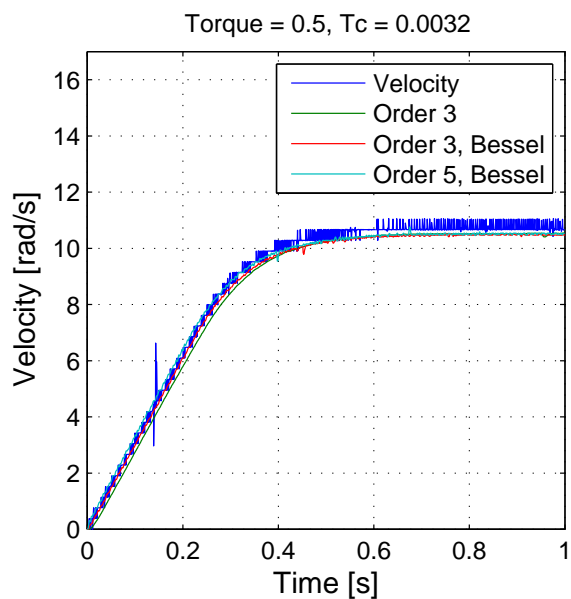
for i = 1:5
    disp(['float bd', num2str(i), ' = ', num2str(Bd(i)), ';']);
end

end

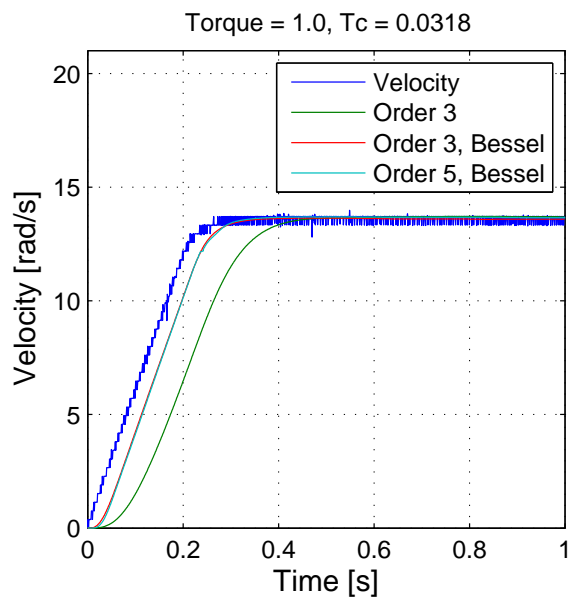
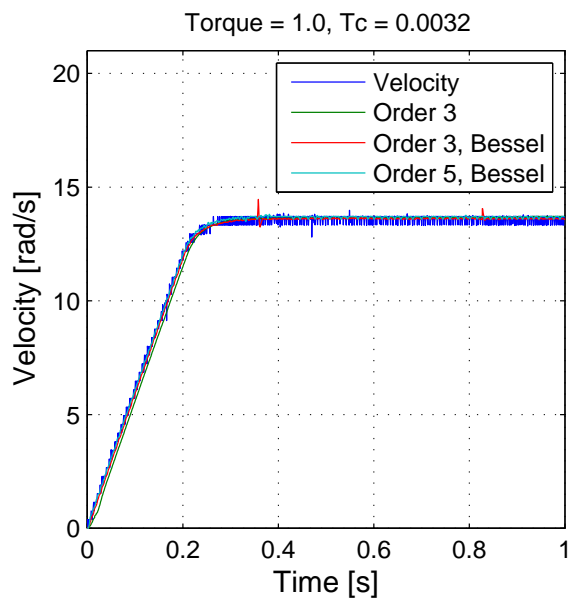
```



6. ábra. Konstans nyomaték kiadása és a fordulatszám szűrése, ResultData.Torque = 0.1;



7. ábra. Konstans nyomaték kiadása és a fordulatszám szűrése, ResultData.Torque = 0.5;



8. ábra. Konstans nyomaték kiadása és a fordulatszám szűrése, ResultData.Torque = 1.0;

```

// A mérés dátuma és időpontja: 2015. 03. 26. 20:00 CET
// IP Address (IPv4): 86.59.219.9
//Task 2 Digital Filter:
//Measurement length in milliseconds: 2000
//Torque: 0.1 Nm, 0.5 Nm, 1.0 Nm
//The state variable names:
//1. time (given)
//2. position (given)
//3. velocity (given)
//4. torque (given)

// IIR filter order 3, Tc=0.0032
//Declaration:
/* velocity filter variables */
static float z_1=0.0, z_2=0.0, z_3=0.0;
static float ztmp_1=0.0, ztmp_2=0.0;
/* velocity filter parameters */
/* Tsample=1e-3 and Tc=0.0032 */
float ad11 = 0.99591;
float ad12 = 0.00095987;
float ad13 = 3.652e-07;
float ad21 = -11.3235;
float ad22 = 0.88778;
float ad23 = 0.00061567;
float ad31 = -19089.6748;
float ad32 = -193.6165;
float ad33 = 0.30752;
float bd1 = 0.0040906;
float bd2 = 11.3235;
float bd3 = 19089.6748;

//Controller:
/* Velocity filter */
ztmp_1=ad11* z_1+ad12* z_2+ad13* z_3 + bd1* ResultData.Velocity;
ztmp_2=ad21* z_1+ad22* z_2+ad23* z_3 + bd2* ResultData.Velocity;
z_3=ad31* z_1+ad32* z_2+ad33* z_3 + bd3* ResultData.Velocity;
z_1 = ztmp_1;
z_2 = ztmp_2;
ResultData.Velocity =z_1;
ResultData.Torque = 0.1;
// ResultData.Torque = 0.5;
// ResultData.Torque =1.0;

// IIR filter order 3, Tc=0.0318
//Declaration:
/* velocity filter variables */
static float z_1=0.0, z_2=0.0, z_3=0.0;
static float ztmp_1=0.0, ztmp_2=0.0;
/* velocity filter parameters */
/* Tsample=1e-3 and Tc=0.0318 */
float ad11 = 0.99999;
float ad12 = 0.00099952;
float ad13 = 4.8454e-07;

```

```

float ad21 = -0.015024;
float ad22 = 0.99856;
float ad23 = 0.00095385;
float ad31 = -29.5753;
float ad32 = -2.8393;
float ad33 = 0.90866;
float bd1 = 5.0475e-06;
float bd2 = 0.015024;
float bd3 = 29.5753;

//Controller:
/* Velocity filter */
ztmp_1=ad11* z_1+ad12* z_2+ad13* z_3 + bd1* ResultData.Velocity;
ztmp_2=ad21* z_1+ad22* z_2+ad23* z_3 + bd2* ResultData.Velocity;
z_3=ad31* z_1+ad32* z_2+ad33* z_3 + bd3* ResultData.Velocity;
z_1 = ztmp_1;
z_2 = ztmp_2;
ResultData.Velocity =z_1;
ResultData.Torque = 0.1;
// ResultData.Torque = 0.5;
// ResultData.Torque =1.0;

// IIR filter order 3 Bessel, Tc=0.0032
//Declaration:
/* velocity filter variables */
static float z_1=0.0, z_2=0.0, z_3=0.0;
static float ztmp_1=0.0, ztmp_2=0.0;
/* velocity filter parameters */
/* Tsample=1e-3 and Tc=0.0032 */
float ad11 = 0.95193;
float ad12 = 0.00083371;
float ad13 = 2.6009e-07;
float ad21 = -120.9669;
float ad22 = 0.56688;
float ad23 = 0.00034345;
float ad31 = -159737.8997;
float ad32 = -629.4284;
float ad33 = -0.080513;
float bd1 = 0.04807;
float bd2 = 120.9669;
float bd3 = 159737.8997;

//Controller:
/* Velocity filter */
ztmp_1=ad11* z_1+ad12* z_2+ad13* z_3 + bd1* ResultData.Velocity;
ztmp_2=ad21* z_1+ad22* z_2+ad23* z_3 + bd2* ResultData.Velocity;
z_3=ad31* z_1+ad32* z_2+ad33* z_3 + bd3* ResultData.Velocity;
z_1 = ztmp_1;
z_2 = ztmp_2;
ResultData.Velocity =z_1;
ResultData.Torque = 0.1;
// ResultData.Torque = 0.5;
// ResultData.Torque =1.0;

```

```

// IIR filter order 3 Bessel, Tc=0.0318
//Declaration:
/* velocity filter variables */
static float z_1=0.0, z_2=0.0, z_3=0.0;
static float ztmp_1=0.0, ztmp_2=0.0;
/* velocity filter parameters */
/* Tsample=1e-3 and Tc=0.0318 */
float ad11 = 0.99993;
float ad12 = 0.00099763;
float ad13 = 4.6943e-07;
float ad21 = -0.21833;
float ad22 = 0.99298;
float ad23 = 0.00090914;
float ad31 = -422.8363;
float ad32 = -13.6776;
float ad33 = 0.82161;
float bd1 = 7.3942e-05;
float bd2 = 0.21833;
float bd3 = 422.8363;

//Controller:
/* Velocity filter */
ztmp_1=ad11* z_1+ad12* z_2+ad13* z_3 + bd1* ResultData.Velocity;
ztmp_2=ad21* z_1+ad22* z_2+ad23* z_3 + bd2* ResultData.Velocity;
z_3=ad31* z_1+ad32* z_2+ad33* z_3 + bd3* ResultData.Velocity;
z_1 = ztmp_1;
z_2 = ztmp_2;
ResultData.Velocity =z_1;
ResultData.Torque = 0.1;
// ResultData.Torque = 0.5;
// ResultData.Torque =1.0;

// IIR filter order 5 Bessel, Tc=0.0032
//Declaration:
/* velocity filter variables */
static float z_1=0.0, z_2=0.0, z_3=0.0, z_4=0.0, z_5=0.0;
static float ztmp_1=0.0, ztmp_2=0.0, ztmp_3=0.0, ztmp_4=0.0;
/* velocity filter parameters */
/* Tsample=1e-3 and Tc=0.0032 */
float ad11 = 0.98945;
float ad12 = 0.00096416;
float ad13 = 4.46e-07;
float ad14 = 1.1837e-10;
float ad15 = 1.5377e-14;
float ad21 = -44.5849;
float ad22 = 0.84738;
float ad23 = 0.00076312;
float ad24 = 2.8527e-07;
float ad25 = 4.5502e-11;
float ad31 = -131587.7641;
float ad32 = -463.4275;
float ad33 = 0.25481;
float ad34 = 0.00029161;

```

```

float ad35 = 7.086e-08;
float ad41 = -204919351.0995;
float ad42 = -783824.2356;
float ad43 = -1386.2332;
float ad44 = -0.47942;
float ad45 = -4.2271e-05;
float ad51 = 122243481346.0161;
float ad52 = 184259271.6258;
float ad53 = -233373.4536;
float ad54 = -948.0155;
float ad55 = -0.28016;
float bd1 = 0.010549;
float bd2 = 44.5849;
float bd3 = 131587.7641;
float bd4 = 204919351.117;
float bd5 = -122243481318.8252;

//Controller:
/* Velocity filter */
ztmp_1= ad11*z_1 + ad12*z_2 + ad13*z_3 + ad14*z_4 + ad15*z_5 + bd1*ResultData.Velocity;
ztmp_2=ad21*z_1 + ad22*z_2 + ad23*z_3 + ad24*z_4 + ad25*z_5 + bd2*ResultData.Velocity;
ztmp_3= ad31*z_1 + ad32*z_2 + ad33*z_3 + ad34*z_4 + ad35*z_5 + bd3*ResultData.Velocity;
ztmp_4= ad41*z_1 + ad42*z_2 + ad43*z_3 + ad44*z_4 + ad45*z_5 + bd4*ResultData.Velocity;
z_5= ad51*z_1 + ad52*z_2 + ad53*z_3 + ad54*z_4 + ad55*z_5 + bd5* ResultData.Velocity;
z_1 = ztmp_1;
z_2 = ztmp_2;
z_3 = ztmp_3;
z_4 = ztmp_4;
ResultData.Velocity =z_1;
ResultData.Torque = 0.1;
// ResultData.Torque = 0.5;
// ResultData.Torque =1.0;

// IIR filter order 5 Bessel, Tc=0.0318
//Declaration:
/* velocity filter variables */
static float z_1=0.0, z_2=0.0, z_3=0.0, z_4=0.0, z_5=0.0;
static float ztmp_1=0.0, ztmp_2=0.0, ztmp_3=0.0, ztmp_4=0.0;
/* velocity filter parameters */
/* Tsample=1e-3 and Tc=0.0318 */
float ad11 = 1;
float ad12 = 0.00099999;
float ad13 = 4.999e-07;
float ad14 = 1.6585e-10;
float ad15 = 3.7886e-14;
float ad21 = -0.001096;
float ad22 = 0.99996;
float ad23 = 0.0009995;
float ad24 = 4.9597e-07;
float ad25 = 1.4799e-10;
float ad31 = -4.2797;
float ad32 = -0.13732;
float ad33 = 0.99804;
float ad34 = 0.00098416;

```

```

float ad35 = 4.2623e-07;
float ad41 = -12326.1315;
float ad42 = -396.6327;
float ad43 = -5.688;
float ad44 = 0.95387;
float ad45 = 0.00078331;
float ad51 = -22652315.0414;
float ad52 = -733371.7135;
float ad53 = -10597.341;
float ad54 = -86.8626;
float ad55 = 0.58474;
float bd1 = 2.2355e-07;
float bd2 = 0.001096;
float bd3 = 4.2797;
float bd4 = 12326.1314;
float bd5 = 22652315.0286;

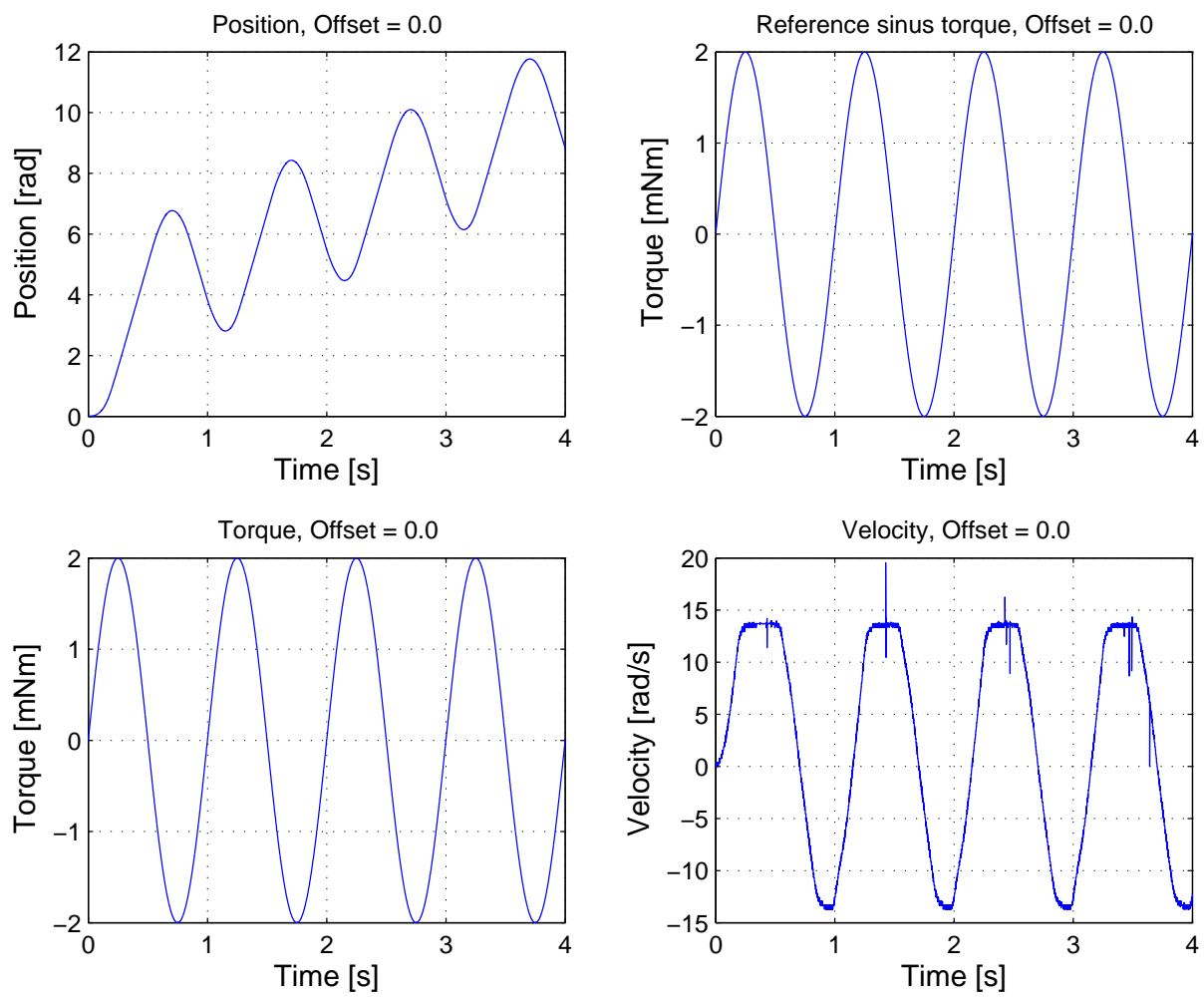
//Controller:
/* Velocity filter */
ztmp_1= ad11*z_1 + ad12*z_2 + ad13*z_3 + ad14*z_4 + ad15*z_5 + bd1*ResultData.Velocity;
ztmp_2=ad21*z_1 + ad22*z_2 + ad23*z_3 + ad24*z_4 + ad25*z_5 + bd2*ResultData.Velocity;
ztmp_3= ad31*z_1 + ad32*z_2 + ad33*z_3 + ad34*z_4 + ad35*z_5 + bd3*ResultData.Velocity;
ztmp_4= ad41*z_1 + ad42*z_2 + ad43*z_3 + ad44*z_4 + ad45*z_5 + bd4*ResultData.Velocity;
z_5= ad51*z_1 + ad52*z_2 + ad53*z_3 + ad54*z_4 + ad55*z_5 + bd5* ResultData.Velocity;
z_1 = ztmp_1;
z_2 = ztmp_2;
z_3 = ztmp_3;
z_4 = ztmp_4;
ResultData.Velocity =z_1;
ResultData.Torque = 0.1;
// ResultData.Torque = 0.5;
// ResultData.Torque =1.0;

```

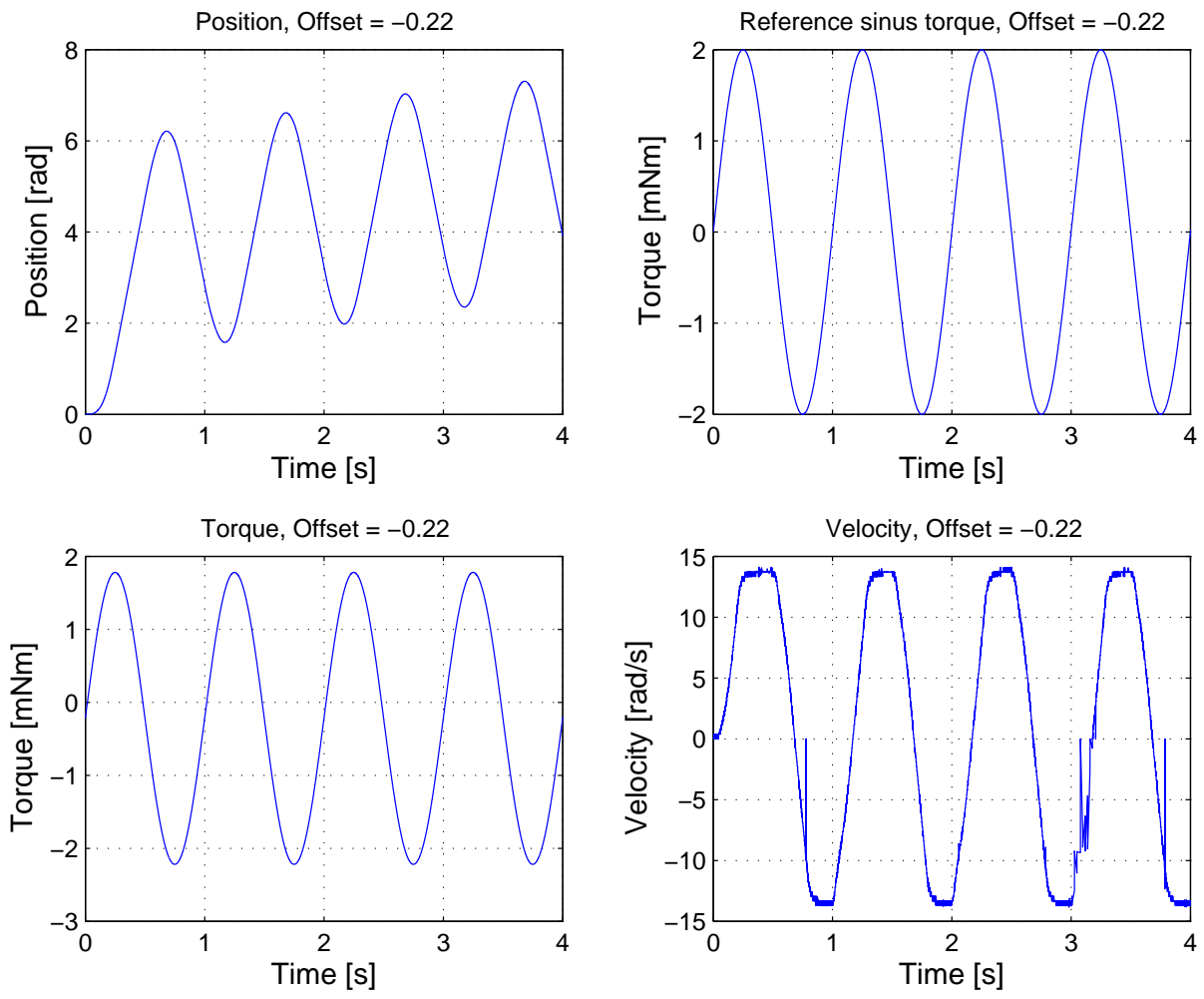
1.4.3. A motor válasza szinuszos nyomatékra és a szervoerősítő offset kompenzációja

A szervoerősítőnek (mint minden műveleti erősítőnek) van egy offset feszültsége. Ez akkor a legszembetűnőbb, ha szinuszos feszültséget alkalmazunk. Ebben az esetben a motortól azt várjuk, hogy szinuszos legyen a tengely szögsebessége és ebből adódóan, szinuszos pozícióváltozása. Az offset feszültséget célszerű hardveresen kompenzálni, de kompenzálhatjuk szoftveresen is.

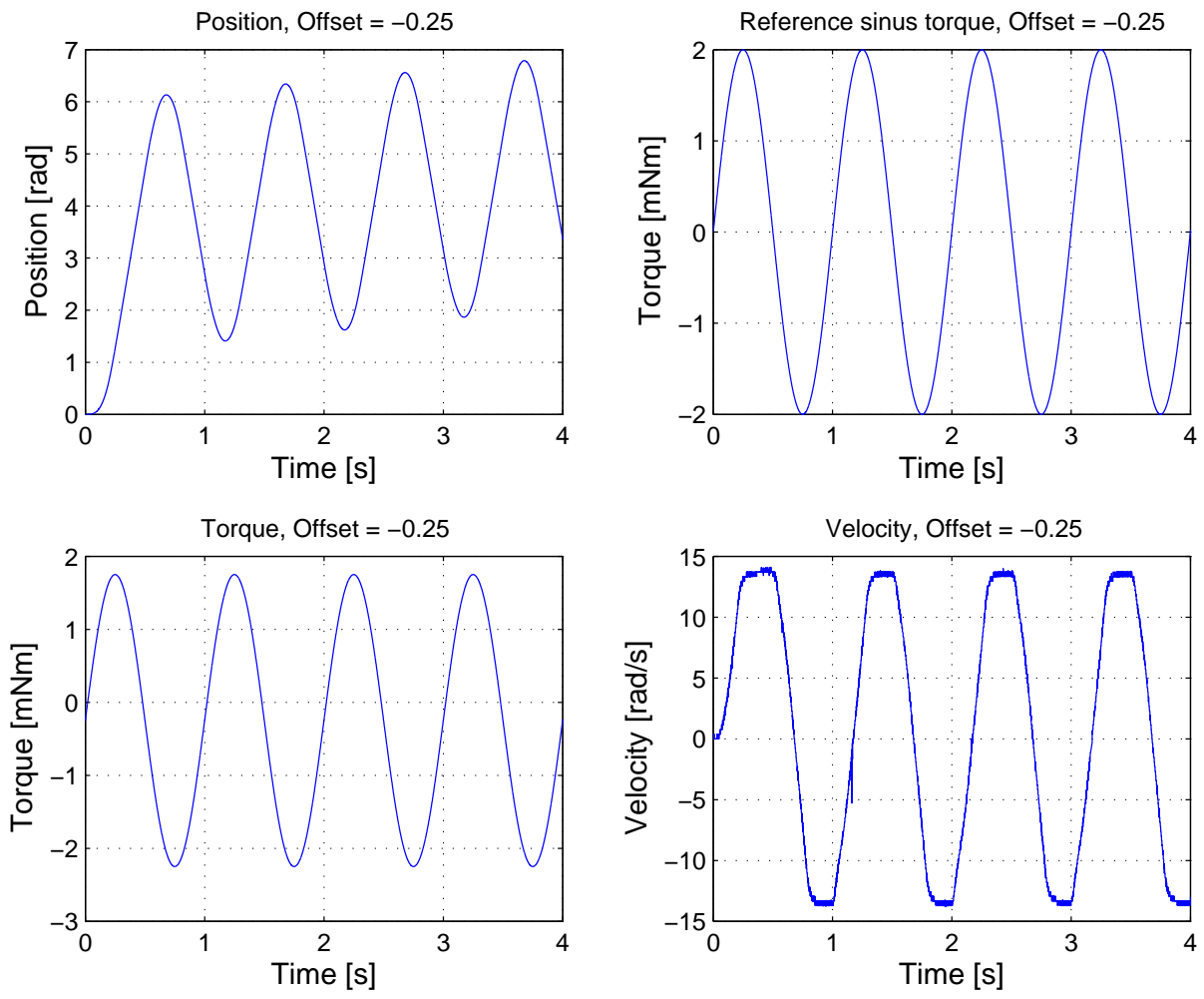
- Változtassa a motor nyomatékát szinuszosan. A szinuszos nyomatékjel amplitúdója legyen 0.6 mNm és a periódusideje legyen 0.1 s és 0.5 s. Vizsgálja meg a motor fordulatszámát és pozícióját.
- Változtassa meg a szinuszos nyomatékjel offszetjét úgy, hogy a pozíciójel középértéke is nulla legyen.



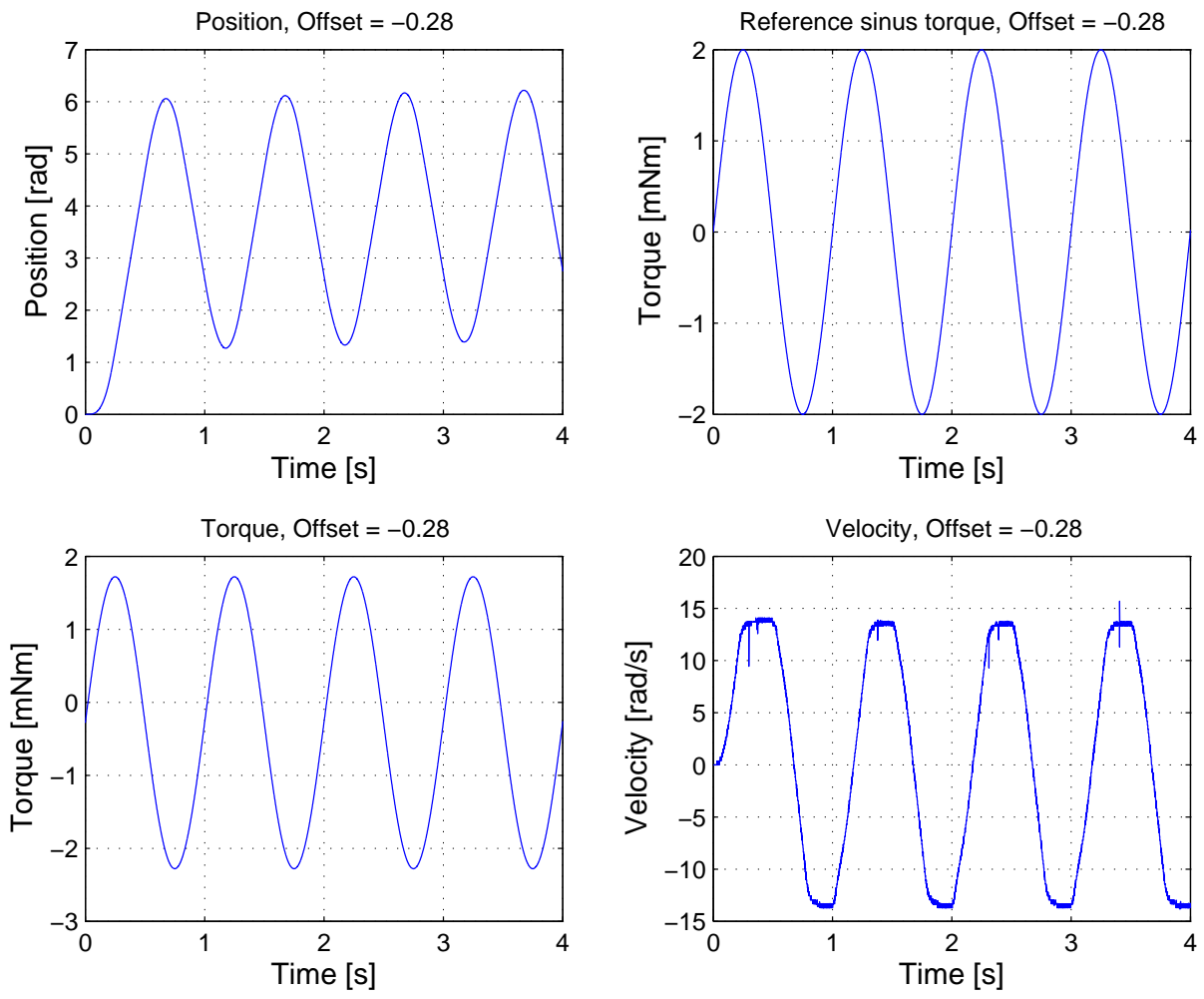
9. ábra. A szervoerősítő offset kompenzációja, Offset = 0.0



10. ábra. A szervoerősítő offset kompenzációja, Offset = -0.22



11. ábra. A szervoerősítő offset kompenzációja, Offset = -0.25



12. ábra. A szervoerősítő offset kompenzációja, Offset = -0.28

```

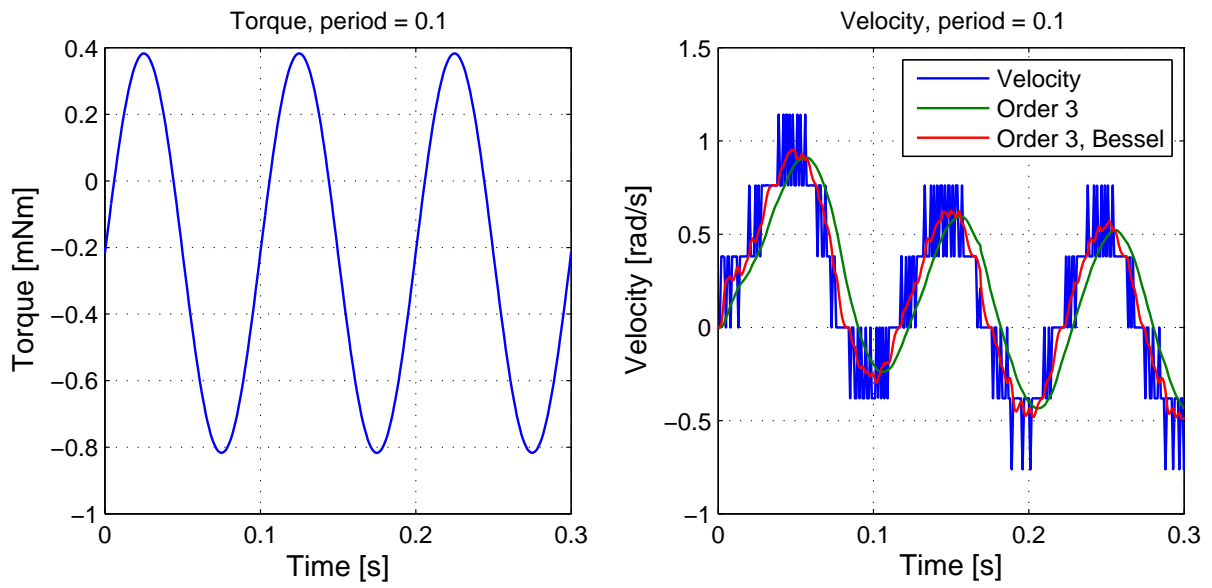
// A mérés dátuma és időpontja: 2015. 03. 26. 21:20 CET
// IP Address (IPv4): 86.59.219.9
//Task 3
// A mérés hossza milliszekundumban: 4000
// Az állapotváltozók nevei:
// time (adott) – idő
// position (adott)– pozíció
// velocity (adott) – sebesség
// torque (adott) – nyomaték
// sin.torque (választott) – sin.nyomaték
//Deklaration:
double param = 0.0;
double sinperiod = 1.0;
double sinamplitude = 2.0;
double offset = 0.0;
// double offset = -0.28;
// double offset = -0.25;
// double offset = -0.22;
// változtassuk az offset tartományát -0.3 és -0.2 közt

// Szabályozó kód:
// Pillanatnyi idő milliszekundumban
param = CurrentTime;
// Pillanatnyi idő másodpercben
param /= 1000.0;
// Eredmény
param = param * 2 * PI / sinperiod;
//
// Szabályozó kezdete
//
// Sinuszos feszültség
ResultData.Torque = sinamplitude * sin(param) + offset;
ResultData.StateVariable_5 = sinamplitude * sin(param);
//
// Szabályozó vége

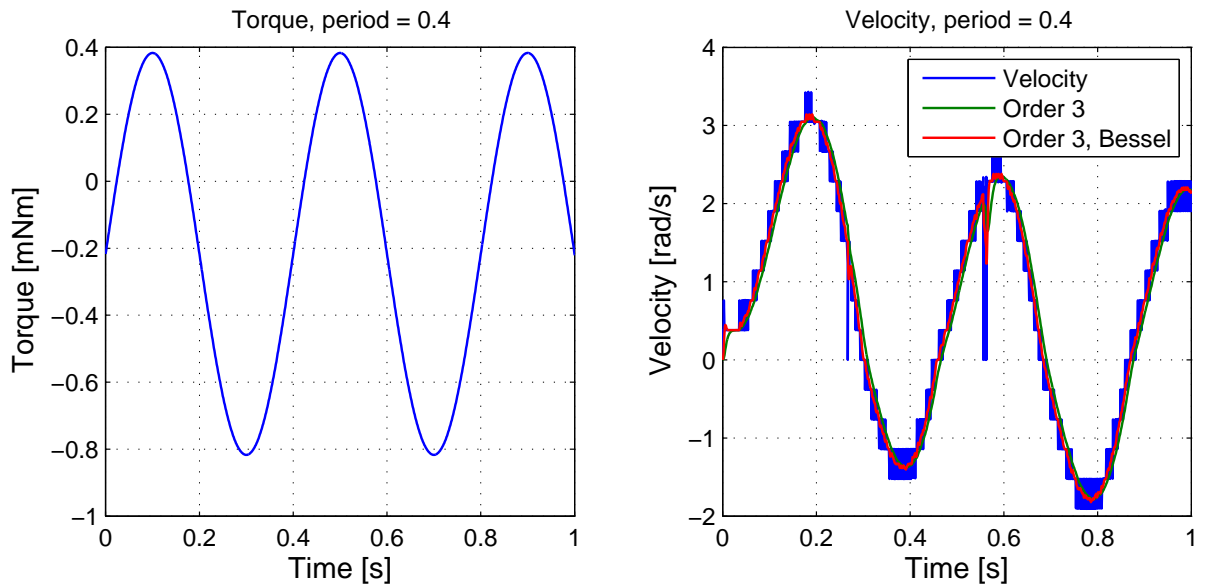
```

1.4.4. Digitális szűrő

- Hajtsa végre az 5.13 mérési feladatot a 5.9.5 pontban leírt digitális szűrők alkalmazásával. Próbálja ki erősebb (kisebb vágási körfrekvenciájú) és gyengébb (nagyobb vágási körfrekvenciájú) szűrőket.
- Vizsgálja meg a fordulatszám jel spektrumát különböző szűrők esetén. Legyen $T_c = 0.032$ és $T_c = 0.0032$.



13. ábra. Digitális szűrő, Period = 0.1



14. ábra. Digitális szűrő, Period = 0.4

```

// A mérés dátuma és időpontja: 2015. 03. 26. 21:50 CET
// IP Address (IPv4): 86.59.219.9
// Task 4
// A mérés hossza milliszekundumban: 1000
// Az állapotváltozók nevei:
// 1. time (adott)
// 2. position (adott)
// 3. velocity (adott)
// 4. torque (adott)
// 5. ref
// 6. filt
// 7. filtb

// Deklarációk:
double param = 0.0;
double sinperiod = 0.4;
// double sinperiod = 0.1;
double sinamplitude = 0.6;
double offset = -0.217;
/* sebesseg szuro valtozoi */
static float z_1=0.0, z_2=0.0, z_3=0.0;
static float ztmp_1=0.0, ztmp_2=0.0;
/* Tsampl=1e-3 Tc=0.0032*/
float ad11 = 0.99591;
float ad12 = 0.00095987;
float ad13 = 3.652e-007;
float ad21 = -11.3235;
float ad22 = 0.88778;
float ad23 = 0.00061567;
float ad31 = -19089.6748;
float ad32 = -193.6165;
float ad33 = 0.30752;
float bd1 = 0.0040906;
float bd2 = 11.3235;
float bd3 = 19089.6748;
/* Bessel sebesseg szuro valtozoi */
static float z_1b=0.0, z_2b=0.0, z_3b=0.0;
static float ztmp_1b=0.0, ztmp_2b=0.0;
/* Bessel Tsampl=1e-3 Tc=0.0032*/
float ad11b = 0.95193;
float ad12b = 0.00083371;
float ad13b = 2.6009e-007;
float ad21b = -120.9668;
float ad22b = 0.56688;
float ad23b = 0.00034345;
float ad31b = -159737.83;
float ad32b = -629.4281;
float ad33b = -0.080513;
float bd1b = 0.048071;
float bd2b = 120.9668;
float bd3b = 159737.83;

// Szabályozó kód:
// Current time
param = CurrentTime;

```

```

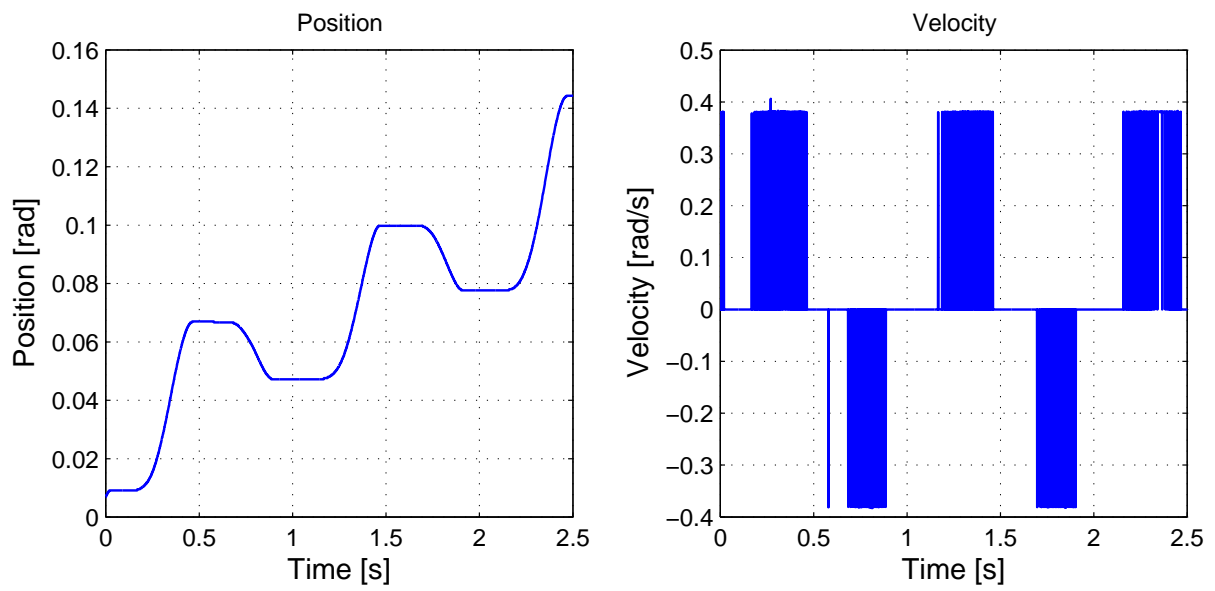
// Only miliseconds
param /= 1000.0;
// Result
param = param * 2 * PI / sinperiod;
//
// Controller part begin
//
// Sinusoidal voltage
ResultData.Torque = sinamplitude * sin(param) + offset;
ResultData.StateVariable_5 = sinamplitude * sin(param);
/* Velocity filter */
ztmp_1=ad11* z_1+ad12* z_2+ad13* z_3 + bd1* ResultData.Velocity;
ztmp_2=ad21* z_1+ad22* z_2+ad23* z_3 + bd2* ResultData.Velocity;
z_3=ad31* z_1+ad32* z_2+ad33* z_3 + bd3* ResultData.Velocity;
z_1 = ztmp_1;
z_2 = ztmp_2;
ResultData.StateVariable_6=z_1;
/* Bessel velocity filter */
ztmp_1b=ad11b* z_1b+ad12b* z_2b+ad13b* z_3b + bd1b*
ResultData.Velocity;
ztmp_2b=ad21b* z_1b+ad22b* z_2b+ad23b* z_3b + bd2b*
ResultData.Velocity;
z_3b=ad31b* z_1b+ad32b* z_2b+ad33b* z_3b + bd3b*
ResultData.Velocity;
z_1b = ztmp_1b;
z_2b = ztmp_2b;
ResultData.StateVariable_7=z_1b;

```

1.4.5. Tapadási és csúszási súrlódás

Ha relatívan kis amplitúdóval szinuszosan változtatjuk a motor referencia nyomatékát, akkor a motor a nullátmenet környékén beragad. Az offset feszültséget az előző mérésben meghatározott értékkel szoftveresen kompenzáljuk.

- Változtassa a motor nyomatékát szinuszosan. A szinuszos nyomatékjel amplitúdója legyen 0.2 mNm és a periódusideje legyen 0.5 s. Az offset legyen a 5.13. feladatban meghatározott érték (-0.21). Vizsgálja meg a motor fordulatszámát és pozícióját.



15. ábra. Tapadási és csúszási súrlódás, nyomaték = 0.2 mNm, periódus = 1.0 s, Offset = -0.21

```

// A mérés dátuma és időpontja: 2015. 03. 26. 23:22 CET
// IP Address (IPv4): 86.59.219.9
// Task 5
// A mérés hossza milliszekundumban: 4000
// Az állapotváltozók nevei:
// 1. time (adott) – idő
// 2. position (adott)– pozíció
// 3. velocity (adott) – sebesség
// 4. torque (adott) – nyomaték
// 5. ref (választott)
// 6. velocity_filtered (választott)
// 7. velocity_bessel (választott)

//Deklaráció:
double param = 0.0;
double sinperiod = 1.0;
double sinamplitude = 0.2;
double offset = -0.21;
/* variables for filter */
static float z_1=0.0, z_2=0.0, z_3=0.0;
static float ztmp_1=0.0, ztmp_2=0.0;
/* Tsample=1e-3 Tc=0.0032*/
float ad11 = 0.99591;
float ad12 = 0.00095987;
float ad13 = 3.652e-007;
float ad21 = -11.3235;
float ad22 = 0.88778;
float ad23 = 0.00061567;
float ad31 = -19089.6748;
float ad32 = -193.6165;
float ad33 = 0.30752;
float bd1 = 0.0040906;
float bd2 = 11.3235;
float bd3 = 19089.6748;
/* Variables for Bessel filter */
static float z_1b=0.0, z_2b=0.0, z_3b=0.0;
static float ztmp_1b=0.0, ztmp_2b=0.0;
/* Bessel Tsample=1e-3 Tc=0.0032*/
float ad11b = 0.95193;
float ad12b = 0.00083371;
float ad13b = 2.6009e-007;
float ad21b = -120.9668;
float ad22b = 0.56688;
float ad23b = 0.00034345;
float ad31b = -159737.83;
float ad32b = -629.4281;
float ad33b = -0.080513;
float bd1b = 0.048071;
float bd2b = 120.9668;
float bd3b = 159737.83;

//Szabályozó kód:
// Current time
param = CurrentTime;
// Only milliseconds

```

```

param /= 1000.0;
// Result
param = param * 2 * PI / sinperiod;
//
// Controller part begin
//
// Sinusoidal voltage
ResultData.Torque = sinamplitude * sin(param) + offset;
ResultData.StateVariable_5 = sinamplitude * sin(param);
/* Velocity filter */
ztmp_1=ad11* z_1+ad12* z_2+ad13* z_3 + bd1* ResultData.Velocity;
ztmp_2=ad21* z_1+ad22* z_2+ad23* z_3 + bd2* ResultData.Velocity;
z_3=ad31* z_1+ad32* z_2+ad33* z_3 + bd3* ResultData.Velocity;
z_1 = ztmp_1;
z_2 = ztmp_2;
ResultData.StateVariable_6=z_1;
/* Bessel velocity filter */
ztmp_1b=ad11b* z_1b+ad12b* z_2b+ad13b* z_3b + bd1b*
ResultData.Velocity;
ztmp_2b=ad21b* z_1b+ad22b* z_2b+ad23b* z_3b + bd2b*
ResultData.Velocity;
z_3b=ad31b* z_1b+ad32b* z_2b+ad33b* z_3b + bd3b*
ResultData.Velocity;
z_1b = ztmp_1b;
z_2b = ztmp_2b;
ResultData.StateVariable_7=z_1b;
// End of controller

```

2. Melléklet

Ábrák Matlab kódja

```
close all
clear all
current_dir = cd;

%% Exercise 2
cd('Exercise-2');
time_2      % t vector
volt_2      % control vector

Plot_exercise2 = figure;
plot( (t./10000000.0), control, 'b', 'LineWidth', 1)
title('Szinuszos feszültség-idő függvény a PCI 1720 A/D kimenetén','FontSize',10)
xlabel('Time [s]','FontSize',12)
ylabel('Voltage [V]','FontSize',12)
xlim([0 10]);
grid on
cd(current_dir);
print(Plot_exercise2,'latex\fig\exercise-2','-depsc')

%% Exercise 3
cd('Exercise-3');
%pos_3      % position vector
time_3      % t vector
%vel_3      % omega vector
volt_3      % control vector

Plot_exercise3Volts = figure;
plot( (t./10000000.0), control, 'b', 'LineWidth', 1)
title('PCI 1784 számláló használata','FontSize',10)
xlabel('Time [s]','FontSize',12)
ylabel('Voltage [V]','FontSize',12)
xlim([0 0.5]);
ylim([-1.5 1.5]);
grid on

cd(current_dir);
print(Plot_exercise3Volts,'latex\fig\exercise-3-volts','-depsc')

%% Exercise 4, Task1
cd('Exercise-4\task-1\ResultData.Torque0.1');
time_01     % t vector
velocity_01 % velocity vector
position_01  % position vector

Plot_ex4_task1_01 = figure('Position',[50 250 1024 450]);
subplot(1,2,1)
```

```

plot( (time./1000), position, 'b', 'LineWidth', 1)
%title('Torque = 0.1','FontSize',10)
xlabel('Time [s]','FontSize',12)
ylabel('Position [rad]','FontSize',12)
%xlim([0 0.5]);
%ylim([-1.5 1.5]);
axis square
grid on

subplot(1,2,2)
plot( (time./1000), velocity, 'b', 'LineWidth', 1)
xlabel('Time [s]','FontSize',12)
ylabel('Velocity [rad/s]','FontSize',12)
%xlim([0 0.5]);
%ylim([-1.5 1.5]);
axis square
grid on

vel_01 = velocity;
t_01 = time./1000;

cd(current_dir);
print(Plot_ex4_task1_01,'latex\fig\ex4-task1-01','-depsc')

cd('Exercise-4\task-1\ResultData.Torque0.5');
time_05      % t vector
velocity_05  % velocity vector
position_05  % position vector

Plot_ex4_task1_05 = figure('Position',[50 250 1024 450]);
subplot(1,2,1)
plot( (time./1000), position, 'b', 'LineWidth', 1)
xlabel('Time [s]','FontSize',12)
ylabel('Position [rad]','FontSize',12)
xlim([0 1.0]);
%ylim([-1.5 1.5]);
axis square
grid on

subplot(1,2,2)
plot( (time./1000), velocity, 'b', 'LineWidth', 1)
xlabel('Time [s]','FontSize',12)
ylabel('Velocity [rad/s]','FontSize',12)
xlim([0 1.0]);
%ylim([-1.5 1.5]);
axis square
grid on

vel_05 = velocity;
t_05 = time./1000;

cd(current_dir);
print(Plot_ex4_task1_05,'latex\fig\ex4-task1-05','-depsc')

```

```

cd('Exercise-4\task-1\ResultData.Torque1.0');
time_1      % t vector
velocity_1  % velocity vector
position_1  % position vector

Plot_ex4_task1_1 = figure('Position',[50 250 1024 450]);
subplot(1,2,1)
plot( (time./1000), position, 'b', 'LineWidth', 1)
xlabel('Time [s]','FontSize',12)
ylabel('Position [rad]','FontSize',12)
%xlim([0 0.5]);
%ylim([-1.5 1.5]);
axis square
grid on

subplot(1,2,2)
plot( (time./1000), velocity, 'b', 'LineWidth', 1)
xlabel('Time [s]','FontSize',12)
ylabel('Velocity [rad/s]','FontSize',12)
%xlim([0 0.5]);
%ylim([-1.5 1.5]);
axis square
grid on

vel_10 = velocity;
t_10 = time./1000;

cd(current_dir);
print(Plot_ex4_task1_1,'latex\fig\ex4-task1-1','-depsc')

%% Exercise 4, Task2
%Torque = 0.1
cd('Exercise-4\task-2\Torque0.1-Tc0.0032');
time_bessel3
t0032_bessel3 = time./1000;
time_bessel5
t0032_bessel5 = time./1000;
time_ord3
t0032_ord3 = time./1000;
velocity_bessel3
v0032_bessel3 = velocity;
velocity_bessel5
v0032_bessel5 = velocity;
velocity_ord3
v0032_ord3 = velocity;

cd(current_dir);
cd('Exercise-4\task-2\Torque0.1-Tc0.0318');
time_bessel3
t0318_bessel3 = time./1000;
time_bessel5

```

```

t0318_bessel5 = time./1000;
time_ord3
t0318_ord3 = time./1000;
velocity_bessel3
v0318_bessel3 = velocity;
velocity_bessel5
v0318_bessel5 = velocity;
velocity_ord3
v0318_ord3 = velocity;

Plot_ex4_task2_01 = figure('Position',[50 250 1024 450]);
subplot(1,2,1)
plot( t_01, vel_01, t0032_ord3, v0032_ord3, t0032_bessel3, v0032_bessel3, t0032_bessel5,
title('Torque = 0.1, Tc = 0.0032','FontSize',10)
legend('Velocity','Order 3','Order 3, Bessel','Order 5, Bessel');
xlabel('Time [s]','FontSize',12)
ylabel('Velocity [rad/s]','FontSize',12)
xlim([0 2.5]);
ylim([0 12.5]);
axis square
grid on

subplot(1,2,2)
plot( t_01, vel_01, t0318_ord3, v0318_ord3, t0318_bessel3, v0318_bessel3, t0318_bessel5,
title('Torque = 0.1, Tc = 0.0318','FontSize',10)
legend('Velocity','Order 3','Order 3, Bessel','Order 5, Bessel');
xlabel('Time [s]','FontSize',12)
ylabel('Velocity [rad/s]','FontSize',12)
xlim([0 2.5]);
ylim([0 12.5]);
axis square
grid on

cd(current_dir);
print(Plot_ex4_task2_01,'latex\fig\ex4-task2-01','-depsc')

%Torque = 0.5
cd('Exercise-4\task-2\Torque0.5-Tc0.0032');
time_bessel3
t0032_bessel3 = time./1000;
time_bessel5
t0032_bessel5 = time./1000;
time_ord3
t0032_ord3 = time./1000;
velocity_bessel3
v0032_bessel3 = velocity;
velocity_bessel5
v0032_bessel5 = velocity;
velocity_ord3
v0032_ord3 = velocity;

cd(current_dir);
cd('Exercise-4\task-2\Torque0.5-Tc0.0318');
time_bessel3
t0318_bessel3 = time./1000;

```

```

time_bessel5
t0318_bessel5 = time./1000;
time_ord3
t0318_ord3 = time./1000;
velocity_bessel3
v0318_bessel3 = velocity;
velocity_bessel5
v0318_bessel5 = velocity;
velocity_ord3
v0318_ord3 = velocity;

Plot_ex4_task2_05 = figure('Position',[50 250 1024 450]);
subplot(1,2,1)
plot( t_05, vel_05, t0032_ord3, v0032_ord3, t0032_bessel3, v0032_bessel3, t0032_bessel5,
title('Torque = 0.5, Tc = 0.0032','FontSize',10)
legend('Velocity','Order 3','Order 3, Bessel','Order 5, Bessel');
xlabel('Time [s]','FontSize',12)
ylabel('Velocity [rad/s]','FontSize',12)
xlim([0 1.0]);
ylim([0 17.0]);
axis square
grid on

subplot(1,2,2)
plot( t_05, vel_05, t0318_ord3, v0318_ord3, t0318_bessel3, v0318_bessel3, t0318_bessel5,
title('Torque = 0.5, Tc = 0.0318','FontSize',10)
legend('Velocity','Order 3','Order 3, Bessel','Order 5, Bessel');
xlabel('Time [s]','FontSize',12)
ylabel('Velocity [rad/s]','FontSize',12)
xlim([0 1.0]);
ylim([0 17.0]);
axis square
grid on

cd(current_dir);
print(Plot_ex4_task2_05,'latex\fig\ex4-task2-05','-depsc')

%Torque = 1.0
cd('Exercise-4\task-2\Torque1.0-Tc0.0032');
time_bessel3
t0032_bessel3 = time./1000;
time_bessel5
t0032_bessel5 = time./1000;
time_ord3
t0032_ord3 = time./1000;
velocity_bessel3
v0032_bessel3 = velocity;
velocity_bessel5
v0032_bessel5 = velocity;
velocity_ord3
v0032_ord3 = velocity;

cd(current_dir);
cd('Exercise-4\task-2\Torque1.0-Tc0.0318');
time_bessel3

```



```

t0318_bessel3 = time./1000;
time_bessel5
t0318_bessel5 = time./1000;
time_ord3
t0318_ord3 = time./1000;
velocity_bessel3
v0318_bessel3 = velocity;
velocity_bessel5
v0318_bessel5 = velocity;
velocity_ord3
v0318_ord3 = velocity;

Plot_ex4_task2_1 = figure('Position',[50 250 1024 450]);
subplot(1,2,1)
plot( t_10, vel_10, t0032_ord3, v0032_ord3, t0032_bessel3, v0032_bessel3, t0032_bessel5,
title('Torque = 1.0, Tc = 0.0032','FontSize',10)
legend('Velocity','Order 3','Order 3, Bessel','Order 5, Bessel');
xlabel('Time [s]','FontSize',12)
ylabel('Velocity [rad/s]','FontSize',12)
xlim([0 1.0]);
ylim([0 21.0]);
axis square
grid on

subplot(1,2,2)
plot( t_10, vel_10, t0318_ord3, v0318_ord3, t0318_bessel3, v0318_bessel3, t0318_bessel5,
title('Torque = 1.0, Tc = 0.0318','FontSize',10)
legend('Velocity','Order 3','Order 3, Bessel','Order 5, Bessel');
xlabel('Time [s]','FontSize',12)
ylabel('Velocity [rad/s]','FontSize',12)
xlim([0 1.0]);
ylim([0 21.0]);
axis square
grid on

cd(current_dir);
print(Plot_ex4_task2_1,'latex\fig\ex4-task2-10','-depsc')

%% Exercise 4, Task3
%Offset = 0.0
cd('Exercise-4\task-3\Sin-T1.0-Offset-0.0');
position_off00
sin_torque_off00
time_off00
torque_off00
velocity_off00

Plot_ex4_task3_offset0 = figure('Position',[50 250 1024 500]);
subplot(2,2,1)
plot( (time./1000), position, 'LineWidth', 0.5)
title('Position, Offset = 0.0','FontSize',10)
xlabel('Time [s]','FontSize',12)
ylabel('Position [rad]','FontSize',12)

```

```

xlim([0 4.0]);
%ylim([0 18.5]);
%axis square
grid on

subplot(2,2,2)
plot( (time./1000), sin_torque, 'LineWidth', 0.5)
title('Reference sinus torque, Offset = 0.0', 'FontSize',10)
xlabel('Time [s]', 'FontSize',12)
ylabel('Torque [mNm]', 'FontSize',12)
xlim([0 4.0]);
%ylim([0 18.5]);
%axis square
grid on

subplot(2,2,3)
plot( (time./1000), torque, 'LineWidth', 0.5)
title('Torque, Offset = 0.0', 'FontSize',10)
xlabel('Time [s]', 'FontSize',12)
ylabel('Torque [mNm]', 'FontSize',12)
xlim([0 4.0]);
%ylim([0 18.5]);
%axis square
grid on

subplot(2,2,4)
plot( (time./1000), velocity, 'LineWidth', 0.5)
title('Velocity, Offset = 0.0', 'FontSize',10)
xlabel('Time [s]', 'FontSize',12)
ylabel('Velocity [rad/s]', 'FontSize',12)
xlim([0 4.0]);
%ylim([0 18.5]);
%axis square
grid on

cd(current_dir);
print(Plot_ex4_task3_offset0, 'latex\fig\ex4-task3-0', '-depsc')

%Offset = -0.22
cd('Exercise-4\task-3\Sin_T1.0_Offset_-0.22');
position_off022
sin_torque_off022
time_off022
torque_off022
velocity_off022

Plot_ex4_task3_offset022 = figure('Position',[50 250 1024 500]);
subplot(2,2,1)
plot( (time./1000), position, 'LineWidth', 0.5)
title('Position, Offset = -0.22', 'FontSize',10)
xlabel('Time [s]', 'FontSize',12)
ylabel('Position [rad]', 'FontSize',12)
xlim([0 4.0]);
%ylim([0 18.5]);
%axis square

```

```

grid on

subplot(2,2,2)
plot( (time./1000), sin_torque, 'LineWidth', 0.5)
title('Reference sinus torque, Offset = -0.22', 'FontSize',10)
xlabel('Time [s]', 'FontSize',12)
ylabel('Torque [mNm]', 'FontSize',12)
xlim([0 4.0]);
%ylim([0 18.5]);
%axis square
grid on

subplot(2,2,3)
plot( (time./1000), torque, 'LineWidth', 0.5)
title('Torque, Offset = -0.22', 'FontSize',10)
xlabel('Time [s]', 'FontSize',12)
ylabel('Torque [mNm]', 'FontSize',12)
xlim([0 4.0]);
%ylim([0 18.5]);
%axis square
grid on

subplot(2,2,4)
plot( (time./1000), velocity, 'LineWidth', 0.5)
title('Velocity, Offset = -0.22', 'FontSize',10)
xlabel('Time [s]', 'FontSize',12)
ylabel('Velocity [rad/s]', 'FontSize',12)
xlim([0 4.0]);
%ylim([0 18.5]);
%axis square
grid on

cd(current_dir);
print(Plot_ex4_task3_offset022, 'latex\fig\ex4-task3-022', '-depsc')

%Offset = -0.25
cd('Exercise-4\task-3\Sin-T1.0.Offset_-0.25');
position_off025
sin_torque_off025
time_off025
torque_off025
velocity_off025

Plot_ex4_task3_offset025 = figure('Position',[50 250 1024 500]);
subplot(2,2,1)
plot( (time./1000), position, 'LineWidth', 0.5)
title('Position, Offset = -0.25', 'FontSize',10)
xlabel('Time [s]', 'FontSize',12)
ylabel('Position [rad]', 'FontSize',12)
xlim([0 4.0]);
%ylim([0 18.5]);
%axis square
grid on

subplot(2,2,2)

```

```

plot( (time./1000), sin_torque, 'LineWidth', 0.5)
title('Reference sinus torque, Offset = -0.25', 'FontSize',10)
xlabel('Time [s]', 'FontSize',12)
ylabel('Torque [mNm]', 'FontSize',12)
xlim([0 4.0]);
%ylim([0 18.5]);
%axis square
grid on

subplot(2,2,3)
plot( (time./1000), torque, 'LineWidth', 0.5)
title('Torque, Offset = -0.25', 'FontSize',10)
xlabel('Time [s]', 'FontSize',12)
ylabel('Torque [mNm]', 'FontSize',12)
xlim([0 4.0]);
%ylim([0 18.5]);
%axis square
grid on

subplot(2,2,4)
plot( (time./1000), velocity, 'LineWidth', 0.5)
title('Velocity, Offset = -0.25', 'FontSize',10)
xlabel('Time [s]', 'FontSize',12)
ylabel('Velocity [rad/s]', 'FontSize',12)
xlim([0 4.0]);
%ylim([0 18.5]);
%axis square
grid on

cd(current_dir);
print(Plot_ex4_task3_offset025, 'latex\fig\ex4-task3-025', '-depsc')

%Offset = -0.28
cd('Exercise-4\task-3\Sin-T1.0_Offset_-0.28');
position_off028
sin_torque_off028
time_off028
torque_off028
velocity_off028

Plot_ex4_task3_offset028 = figure('Position',[50 250 1024 500]);
subplot(2,2,1)
plot( (time./1000), position, 'LineWidth', 0.5)
title('Position, Offset = -0.28', 'FontSize',10)
xlabel('Time [s]', 'FontSize',12)
ylabel('Position [rad]', 'FontSize',12)
xlim([0 4.0]);
%ylim([0 18.5]);
%axis square
grid on

subplot(2,2,2)
plot( (time./1000), sin_torque, 'LineWidth', 0.5)
title('Reference sinus torque, Offset = -0.28', 'FontSize',10)
xlabel('Time [s]', 'FontSize',12)

```

```

ylabel('Torque [mNm]', 'FontSize', 12)
xlim([0 4.0]);
%ylim([0 18.5]);
%axis square
grid on

subplot(2,2,3)
plot( (time./1000), torque, 'LineWidth', 0.5)
title('Torque, Offset = -0.28', 'FontSize', 10)
xlabel('Time [s]', 'FontSize', 12)
ylabel('Torque [mNm]', 'FontSize', 12)
xlim([0 4.0]);
%ylim([0 18.5]);
%axis square
grid on

subplot(2,2,4)
plot( (time./1000), velocity, 'LineWidth', 0.5)
title('Velocity, Offset = -0.28', 'FontSize', 10)
xlabel('Time [s]', 'FontSize', 12)
ylabel('Velocity [rad/s]', 'FontSize', 12)
xlim([0 4.0]);
%ylim([0 18.5]);
%axis square
grid on

cd(current_dir);
print(Plot_ex4_task3_offset028, 'latex\fig\ex4-task3-028', '-depsc')

%% Exercise 4, Task4
%period = 0.1
cd('Exercise-4\task-4\Filter_period0.1');
filt_p01
filtb_p01
time_p01
torque_p01
velocity_p01
t = time./1000;

Plot_ex4_task4_01 = figure('Position', [50 250 1024 450]);
subplot(1,2,1)
plot( t, torque, 'b', 'LineWidth', 1)
title('Torque, period = 0.1', 'FontSize', 10)
xlabel('Time [s]', 'FontSize', 12)
ylabel('Torque [mNm]', 'FontSize', 12)
xlim([0 0.3]);
%ylim([-1.5 1.5]);
axis square
grid on

subplot(1,2,2)
plot( t, velocity, t, filt, t, filtb, 'LineWidth', 1)

```

```

legend('Velocity','Order 3','Order 3, Bessel');
title('Velocity, period = 0.1','FontSize',10)
xlabel('Time [s]','FontSize',12)
ylabel('Velocity [rad/s]','FontSize',12)
xlim([0 0.3]);
%ylim([-1.5 1.5]);
axis square
grid on

cd(current_dir);
print(Plot_ex4_task4_01,'latex\fig\ex4-task4-01','-depsc')

%period = 0.4
cd('Exercise-4\task-4\Filter_period0.4');
filt_p04
filtb_p04
time_p04
torque_p04
velocity_p04
t = time./1000;

Plot_ex4_task4_04 = figure('Position',[50 250 1024 450]);
subplot(1,2,1)
plot(t, torque, 'b', 'LineWidth', 1)
title('Torque, period = 0.4','FontSize',10)
xlabel('Time [s]','FontSize',12)
ylabel('Torque [mNm]','FontSize',12)
xlim([0 1.0]);
%ylim([-1.5 1.5]);
axis square
grid on

subplot(1,2,2)
plot(t, velocity, t, filt, t, filtb, 'LineWidth', 1)
legend('Velocity','Order 3','Order 3, Bessel');
title('Velocity, period = 0.4','FontSize',10)
xlabel('Time [s]','FontSize',12)
ylabel('Velocity [rad/s]','FontSize',12)
xlim([0 1.0]);
%ylim([-1.5 1.5]);
axis square
grid on

cd(current_dir);
print(Plot_ex4_task4_04,'latex\fig\ex4-task4-04','-depsc')

%% Exercise 4, Task5
cd('Exercise-4\task-5\Csuszo_period1.0');
position_p01
velocity_p01
time_p01

```

```

t = time./1000;

Plot_ex4_task5 = figure('Position',[50 250 1024 450]);
subplot(1,2,1)
plot(t, position, 'b', 'LineWidth', 1)
title('Position', 'FontSize', 10)
xlabel('Time [s]', 'FontSize', 12)
ylabel('Position [rad]', 'FontSize', 12)
xlim([0 2.5]);
ylim([-1.5 1.5]);
axis square
grid on

subplot(1,2,2)
plot(t, velocity, 'b', 'LineWidth', 1)
%legend('Velocity', 'Order 3', 'Order 3, Bessel');
title('Velocity', 'FontSize', 10)
xlabel('Time [s]', 'FontSize', 12)
ylabel('Velocity [rad/s]', 'FontSize', 12)
xlim([0 2.5]);
ylim([-1.5 1.5]);
axis square
grid on

cd(current_dir);
print(Plot_ex4_task5, 'latex\fig\ex4-task5', '-depsc')

```