Rádióamatőr tanfolyamot segítő jegyzet, egyelőre kidolgozás alatt
2021 ősz

Szabó Áron HA1FLX, Kiss Ádám HA8KDA,
Bazsó Márton HA7BM, Keresztes Botond

Modulációs gyakorlat

Tartalom

1. Bevezető **2**

2. Előkészületek **2**

3. Elméleti összefoglaló **3**
3.1. Keverés . . . 3
3.1.1. Tükörfrekvencia . . . 3
3.2. Keverés matematikája . . . 4
3.3. Egyszerű modulációk konstellációs ábrája . . . 5

4. Feladatok **7**
4.1. Szinusz jel generálása, ábrázolása . . . 7
4.2. Keverés . . . 10
4.3. IQ moduláció . . . 11
4.3.1. Fázismoduláció . . . 11
4.4. Újramintavételezés . . . 13

1. Bevezető

A gyakorlat célja bemutatni a különböző modulációs eljárások idő- és frekvenciatartománybeli viselkedését, illetve matematikai leírásukat. Ezzel megalapozzuk a szoftverrádiózáshoz szükséges tudást.

2. Előkészületek

A gyakorlat Octave-ban és Matlab-ban is elvégezhető, a használt parancsok mindkettőben működőképesek.

Octave telepítés:

- **GNU Octave telepítése**
  `https://wiki.octave.org/Category:Installation`
- **Signal package telepítése**
  A jelfeldolgozó könyvtárak a Matlab-ban a telepítéssel érkeznek, Octave esetén csomagként érhetők el, és használat előtt be kell tölteni őket. A `signal` könyvtár a `control`-ra is támaszkodik, így a megfelelő sorrendben telepítsük őket.
  Írjuk be az Octave parancssorába:
  ```
  pkg install -forge control
  pkg install -forge signal
  ```
  Az Octave indulásakor ez nem töltődik be automatikusan, mielőtt jelfeldolgozási függvényeket használnánk, töltsük be a `signal` comagot:
  ```
  pkg load signal
  ```
  A könyvtárat csomagkezelőből is telepítheted:
  ```
  sudo apt-get install octave-signal #Ubuntu
  sudo pacman -S octave-signal #Arch vagy Manjaro
  sudo zypper install octave-forge-signal #RPM (OpenSUSE)
  ```

3. Elméleti összefoglaló

3.1. Keverés

Az elektromágneses spektrum egy véges erőforrás, így szeretnénk hatékonyan kihasználni. „Emberi fogyasztásra alkalmas” jeleinket az alapsávban (hangátvitel esetén kb. 20 Hz - 20 kHz között) állítjuk elő, de értelemszerűen ha mindannyian ebben a tartományban forgalmaznánk, akkor nagyon kevés adást tudnánk sugározni, mert mindig csak egymást zavarnánk. Gondoljunk csak arra, amikor mindenki egyszerre beszél az étkezőasztalnál, nagy hangzavar keletkezik, és semmit sem érteni. Épp ezért, az adásainkat szeretnénk az elektromágneses (EM) spekturm különböző sávjaiban elhelyezni, így jobban kihasználva azt.

Erre eszközünk a keverés, mellyel jeleinket el tudjuk tolni frekvenciatartományban, vagyis az EM spektrumban. Ez matematikailag két jel összeszorzását jelenti. A pontos összefüggést a modulációs tétel írja le:

$$\cos(\omega_1 t) \cdot \cos(\omega_2 t) = \frac{1}{2}\cos((\omega_1 + \omega_2)\, t) + \frac{1}{2}\cos((\omega_1 - \omega_2)\, t) \tag{1}$$

Ezt a két jelet vivőnek és moduláló (alapsávi) jelnek szokás hívni. A tételben szereplő jelek szinuszosak, de a gyakorlatban a moduláló jelünk tetszőleges lehet. A tétel továbbra is alkalmazható lesz, ugyanis bármilyen jel felbontható szinuszos jelek összegére, és ezek mindegyikére egyenként alkalmazhatjuk a modulációs tételt. Ha ezt a módszert önmagában alkalmazzuk, azt amplitúdómodulációnak hívjuk (AM). A frekvencia- és fázismodulációkóról a tanfolyamon már volt szó, de lentebb átismételjük őket.

3.1.1. Tükörfrekvencia

Létezik úgynevezett tükörfrekvencia. Ez az a sáv, amit a keverő ugyanarra a frekvenciára kever, mint a hasznos jelet. Vegyünk például egy 100 MHz-es adást, amit 10 MHz-re szeretnénk lekeverni. A modulációs tétel (1) értelmében ezt például úgy érhetjük el, hogy a 100 MHz-es adást megszorozzuk egy 90 MHz-es oszcillátorral (a másik lehetőség egy 110 MHz-es helyi oszcillátor jel lenne), így az 10 MHz-en megjelenik mint különbségi frekvencia. Viszont ekkor a 80 MHz-en lévő spektrumrész is bekeverődik 10 MHz-re, mint összegfrekvencia. Az így fellépő, de oda nem illő részt hívjuk tükörátlapolódásnak.

Ezt az eredeti, ide lapolódó spektrumrészt ki kell szűrni a keverés előtt, különben áthallás lesz két adó között. A keverő kimenetét középfrekvenciának szokták hívni, az ezt követő szűrőt KF szűrőnek. (Angolul intermediate freq, és IF filter).

Az IQ moduláció segítségével a tükörfrekvenciából adódó problémák kivédhetők.

3.2. Keverés matematikája

Vizsgáljuk csupán egy szinuszjelet, ennél összetettebb jel felfogható több szinusz összegeként, a lentiek igazak erre az esetre is, csak több példányban értelmezve.

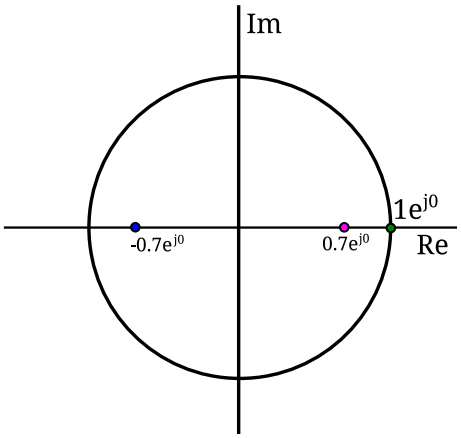
Vegyünk egy vektort, ami forog körbe körbe. Vegyük ennek a vektor csücskének az egyik tengelyre vett árnyékát. A tengelyen a szinusz függvény szerint fog mozogni a pont. Ha ezt a tengelyen mozgó pontot egy ceruzára cseréljük, és a papírt elkezdjük a mozgására merőlegesen lehúzni egyenletes sebességgel, akkor egy ideális szinuszt fogunk kapni. Bizonyítást lásd Euler-tétel.

A keverés során semmi mást nem teszünk, mint próbáljuk felvenni a sebességet a jellel. Azaz a vektor továbbra is forog úgy, ahogy forog, viszont a koordináta rendszert elkezdjük forgatni hozzá. Ha ugyan azon irányba, akkor különbségi keverést csinálunk, ha ellentétesbe, akkor összegzőt. (A valóságban persze a kettőt csak együtt tudjuk csinálni. Valós szinusszal keverünk, ami az Euler-tétel miatt két részre bontható, az egyik az egyik, a másik iránynak felel meg.) Ekkor elképzelhető, hogy a kapott szinuszunk lassul vagy gyorsul. Le tudjuk azonos sebesség esetén nullára is lassítani a látszólagos sebességét. Ekkor a szinusz és a koordináta-rendszerünk fázisától függően egy valamilyen álló fázisban látjuk a vektort.

A koordináta-rendszert, amiben ábrázoljuk a lelassított szinusz, konstellációs diagramnak nevezzük. A vett jel (tfh. szinusz) vektorának a végpontját pedig konstellációs pontnak. Szokás a különböző konstellációs térnegyedekhez vagy ennél is finomabb felbontásokhoz digitális szimbólumokat is rendelni, amiket később (miután rászinkronizált a vevő az adó bitkibocsájtási ütemére) bitfolyamként tud értelmezni egy következő fokozat.

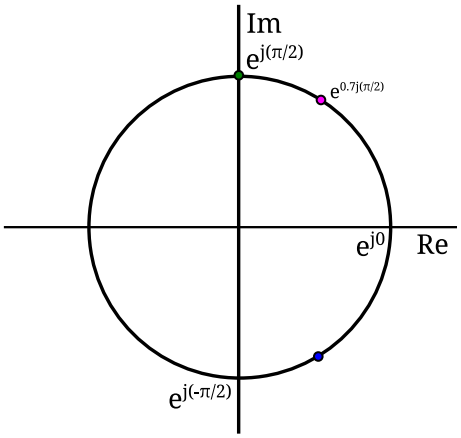
3.3. Egyszerű modulációk konstellációs ábrája

AM Amplitúdó modulált esetben a konstellációs pont szöge áll, azonban az amplitúdója változik.



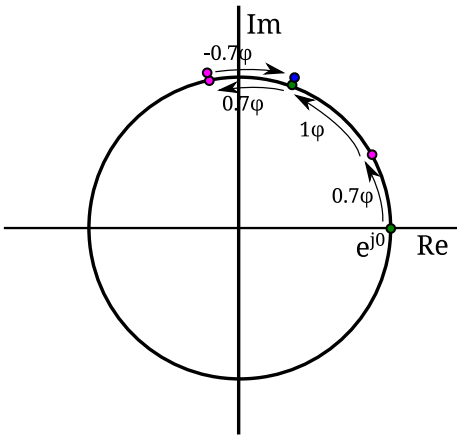
1. ábra. AM konstellációs ábrája

PM Az alapsávi jel amplitúdóját a modulált jel fázisával reprezentáljuk. A jel 0 feletti értéke a nulla fokhoz képest pozitív, a 0 alatti értéke negatív szöget jelent. Az amplitúdó arányát a szöghöz képest fázislöketnek nevezzük. A fázislöket túl nagy értékénél körbeérhetünk a körön, túl kicsi értéknél viszont nehezen észlelhető a fázisváltozás.



2. ábra. PM konstellációs ábrája

FM A frekvenciamoduláció nagyban hasonlít a fázismodulációra, azonban itt a fázis változásának mértéke hordozza az információt (tehát a szögsebessége, a fázis deriváltja). Minél nagyobb a fáziskülönbség két egymást követő pont között, annál nagyobb az alapsávi jel amplitúdója. A moduláló jel amplitúdója és a fázisváltozás közti szorzót itt is löketnek nevezzük.



3. ábra. FM konstellációs ábrája

4. Feladatok

4.1. Szinusz jel generálása, ábrázolása

Első feladatunk egy szinusz jel generálása lesz, melyet a *sin()* függvénnyel tehetünk meg. Fontos, hogy ez a függvény egy adott érték szinuszát adja vissza, nekünk viszont egy időtartománybeli függvényre van szükségünk.

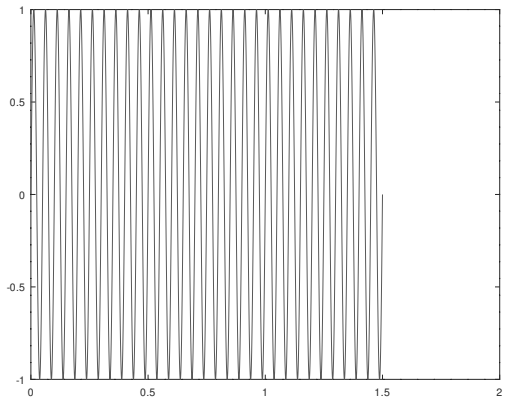
Létre kell hoznunk tehát egy időtengelyt, aminek pontjaiban kiszámoljuk a függvény értékét. Ebből a szempontból fontos meghatározni a mintavételi frekvenciát (F_s), hiszen ebből az értékből számolható, hogy két minta közt mennyi idő fog eltelni. Válasszuk a mintavételi frekvenciát *44 100 Hz*-re, ami a digitális audió jelfeldolgozásban használatos frekvencia, és a számítógép hangkártyáján könnyen lejátszható.

Ha a minta hosszát (l) másodpercben szeretnénk megadni, a minták számának (N) meghatározásához egyszerűen szorozzuk meg a másodperc értéket a mintavételi frekvenciával. A másodperc időtengelyt (t) úgy kaphatjuk meg, ha az `1:N` sorozatot elemenként elosztjuk a mintavételi frekvenciával (a transzponálásra azért van szükség, hogy oszlopvektort kapjunk). Ez után a szinusz jel egyszerűen kiszámolható: mivel a szinusz egy periódusa 2π, a szinuszjel pillanatnyi értékét $2\pi \cdot f \cdot t$-re kell választani, ahol f a szinusz frekvenciája.

Kód 1. Mintavételi frekvencia beállítása és szinuszjel előállítása

```
Fs = 44100; % mintaveteli freki
l = 1.5; % minta hossza masodpercben
N = l * Fs, % mintak szama
f = 440; % 440 Hz-es lesz a jel
t = ( 1:N )' ./ Fs; % Ido oszlopvektora
y = sin( 2 * pi * f * t );
plot( t, y );
xlim([t(1), t(end)]) %grafikon x-tengely hatarainak beallitasa
```

Ábrázoljuk a Plot függvénnyel időtartományban:



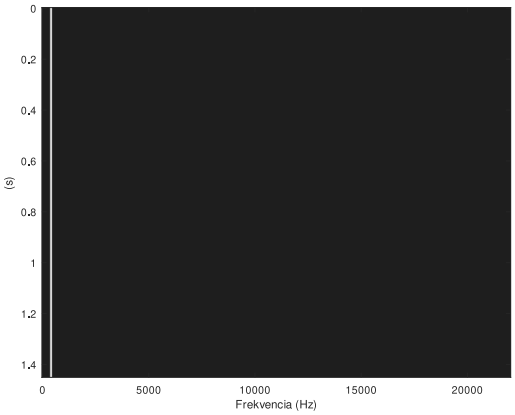
4. ábra. 20 Hz-es szinusz függvény ábrázolása időtartományban

A 20 Hz-es jelet könnyen ábrázoltuk, de a hallható tartomány magasabb frekvencián van. Változtassuk a jel frekvenciáját 440 Hz-re (zenei A hang), és játsszuk ki a számítógép hangkártyáján a jelet:

```
soundsc(y, Fs); %Hang lejatszasa Fs mintaveteli frekvenciaval
```

Itt már több információt kapnánk a jelről, ha frekvenciatartományban, vízesés diagramon ábrázolnánk a jelet. A HA5KFU Octave gyakorlathoz csomagolt *wf(y, Fs)* függvénnyel ezt a rádiózásban megszokott formában rajzolhatjuk ki.

```
wf(y, Fs); %Vizeses diagram kirajzolasa
```



5. ábra. 440 Hz-es szinusz függvény ábrázolása frekvenciatartományban

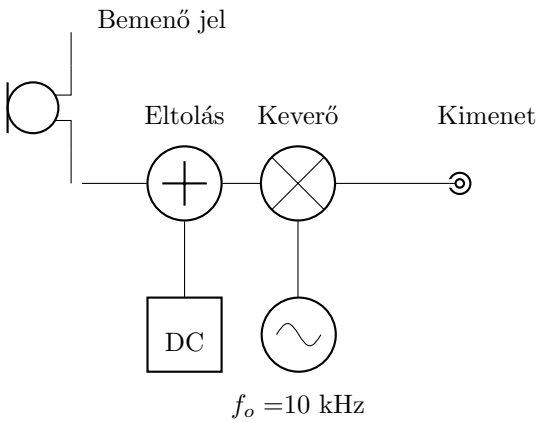
A vízesés diagram 0-tól a mintavételi frekvencia feléig ábrázolja a frekvenciakomponenseket, az ábrán láthatjuk a konstans szinusz jel képét 440 Hz-nél. Próbálkozzunk meg a jel manipulálásával, hallgassuk meg, ábrázoljuk frekvenciatartományban!

```
y = ( 1 - ( 1 / l )* t ).* sin( 2 * pi * f * t ); %Fokozatosan halkulo jel
y = sin( t .* ( f * t )* 2 * pi ); %Valtozo frekvenciaju szinusz
y = sin( 2 * pi * f * t )+ sin( 2 * pi * (1.5 * f)* t ); %Tobb szinusz osszege
```

Megpróbálhatunk akár fehér zajt generálni, vagy hangfájlt betölteni.

```
y = rand( N, 1 ); %N elemu zaj
[y,Fs] = audioread( 'akkord.wav'); N = length( y ); l=N/Fs; %WAV file beolvasasa
```

4.2. Keverés



6. ábra. A keverő blokkvázlata (megjegyzés: a fenti elrendezés digitális jelfeldolgozásnál működhet, analóg keverő nem kaphat DC eltolást!)

Amikor egy jelet egy szinusz oszcillátor jelével keverünk (szorzunk), a jel megjelenik a frekvenciatartományban a két frekvencia összegeként és különbségeként. Ha pl a két jel egy ω_1 frekvenciájú koszinusz vivő, és egy ω_2 frekvenciájú koszinusz moduláló jel, a modulációs tétel alapján:

$$\cos(\omega_1 t) \cdot \cos(\omega_2 t) = \frac{1}{2}\cos(\omega_1 t + \omega_2 t) + \frac{1}{2}\cos(\omega_1 t - \omega_2 t) \tag{2}$$

Nézzük meg a gyakorlatban, próbáljuk meg az *akkord.wav* hangfilet keverni, például egy 10 kHz-es szinusz jellel!

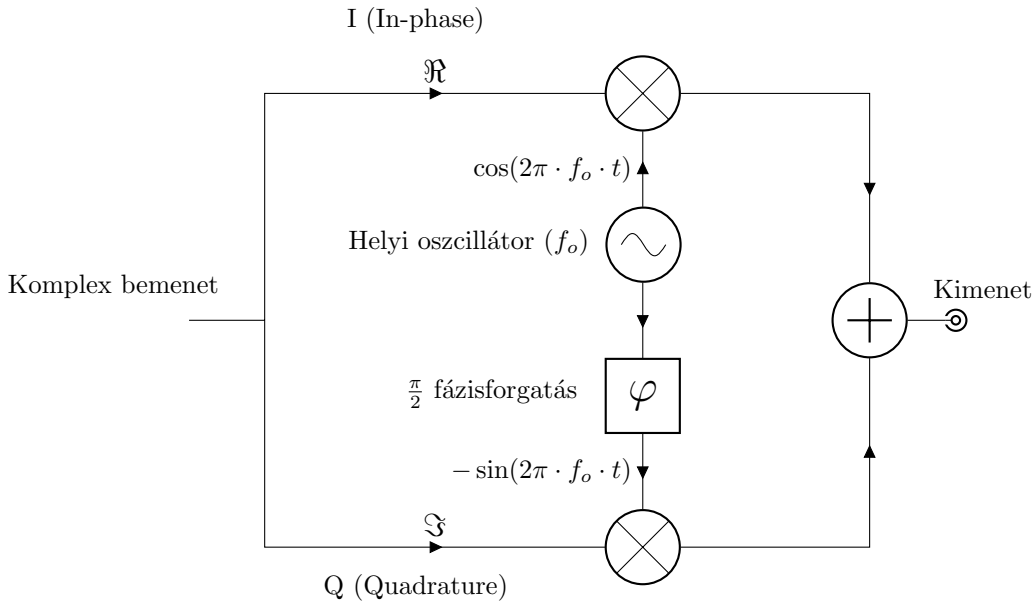
Kód 2. Keverés

```
[a,Fs] = audioread('akkord.wav'); N=length(a);
f = 10000; % 10 kHz-es lesz a vivo jel
t = ( 1:N )' ./ Fs;
b = sin( 2 * pi * f * t );

m = 0.75; % modulacios index
dc_offset = (1 + 1-m) * max(a);
a_dc = a + dc_offset; % dc eltolas

y = a_dc .* b; % keveres (a .* b)

wf(y, Fs);
```

Fontos, hogy eltoljuk az eredeti jelet úgy, hogy a moduláló jel ne menjen 0 szint alá, különben túlmoduláljuk a vivő jelet. A vivő és moduláló jel amplitúdójának kapcsolatát a modulációs mélységgel jellemezzük.

Vajon hogyan kapjuk vissza a kevert jelből az eredeti információt?

4.3. IQ moduláció



7. ábra. Az IQ modulátor blokkvázlata

Az IQ modulátornál egy komplex jellel moduláljuk a vivő frekvenciát. Ezzel a módszerrel az alapsávi jelet egy komplex fazorként ábrázolhatjuk, így – kellő ügyességgel – tetszőleges moduláció létrehozható.

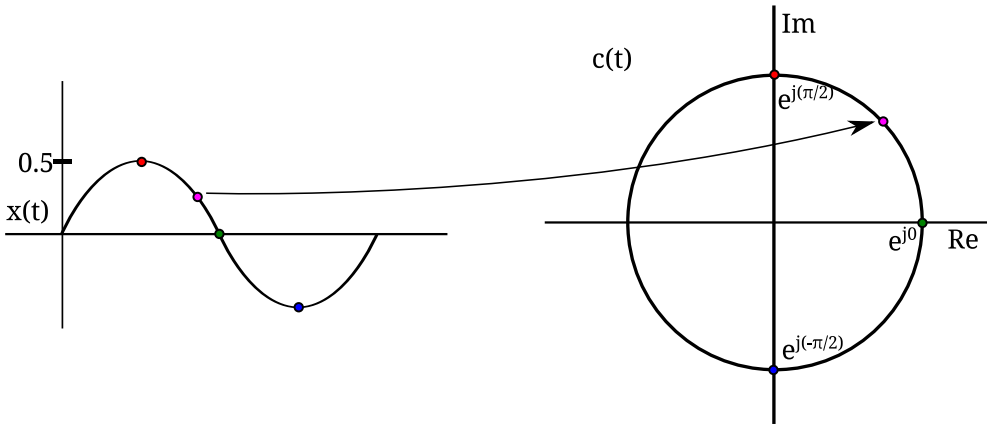
4.3.1. Fázismoduláció

Vegyük át újra hogy is néz ki egy szinusz.

$$y(t) = A \cdot \sin(2\pi \cdot f \cdot t + \phi) \tag{3}$$

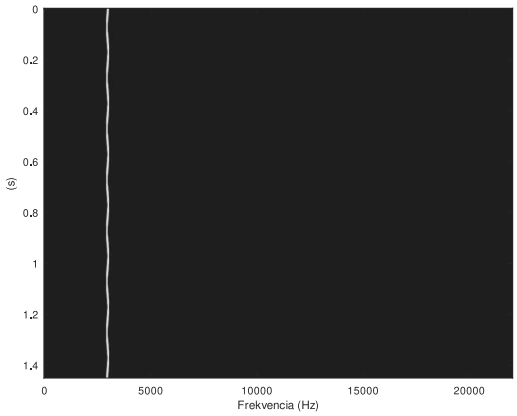
A képletben lévő amplitúdót amplitúdómodulációban, a frekvenciát frekvenciamodulációban használjuk, a fázist pedig fázismodulációban. Talán a fázismodulációt a legnehezebb megfogni élettapasztalatból, mivel a fülünkkel nem halljuk ki a jelek fázisát – furcsa is lenne ha egy méterrel távolabbról másnak hallanánk mások hangját. A fázis jelen esetben is nehezen értelmezhető, mivel a kiadott jel úgymond elveszik, nincs mihez hasonlítani első közelítésben majd vevő oldalon a fázist. A probléma feloldása az, hogy egyrészt eltárolhatjuk a jel mintáit majd vevőoldalon, illetve a fázist tudja majd a vevő a saját, azonos frekvencián futó oszcillátorához képest mérni. Különböző frekvenciákhoz nincs értelme fázist mérni általában, mert a különböző frekvencia miatt fázisban úgyis elmásznak egymástól.

Készítsünk fázismodulátort IQ moduláció segítségével! Legyen az átvinni kívánt (moduláló) jel $x(t)$ (ügyelve arra, hogy $|x(t)| < 1$)! Ekkor a $c(t)$ komplex vektor hossza $|c(t)| = 1$, fázisa pedig a modulált jel pillanatnyi értéke ($\arg(c(t)) = x(t) \cdot \pi$). Így a jel 1 értéke a π fázishoz kerül. A $c(t)$ jelet az Euler alakkal könnyen előállíthatjuk: $c(t) = e^{j \cdot \pi \cdot x(t)}$.



8. ábra. Fázisok előállítása a moduláló jelből

Az IQ keveréshez ennek a komplex jelnek a valós részét az in-phase $\cos(2\pi \cdot f_o \cdot t)$, a képzetes részét a quadrature $-\sin(2\pi \cdot f_o \cdot t)$ oszcillátor jellel moduláljuk (szorozzuk), majd összeadjuk. Kész is a modulált jel!



9. ábra. Az előállított fázismodulált jel

Kód 3. Fázismoduláció IQ modulátorral

```
Fs = 44100; % mintaveteli freki
l = 1.5; % minta hossza masodpercben
N = l * Fs; % mintak szama
t = ( 1:N )' ./ Fs; % Ido oszlopvektora

f = 5; % eredeti jel frek
x =  sin( 2 * pi * f * t ) .* 0.5; % ez lesz az atvinni kivant jel (1/2
    amplitudo)

c = ones(N,1); % N meretu vektor letrehozasa es feltoltese 1-essel
for i=1:N
  c(i) = exp( j * pi * x(i) ); % fazismodulalt jel letrehozasa
  % e^(2pi*j*x/2), mert nem szeretnenk a korbeforgassal atlapolodni
end

fo = 3000; % oszcillator frek
oi =  cos( 2 * pi * fo * t ); % in-phase oszcillator
oq = -sin( 2 * pi * fo * t ); % quadrature oszcillator

y = oq .* imag(c) + oi .* real(c); % IQ keveres

figure(1);
plot(t, y);
figure(2);
wf(y, Fs);
```

Feladat: Ez alapján készítsünk el egy FM modulátort! Nagyon hasonlít az előzőhöz, csak itt a komplex vektor forgásának frekvenciája változik.

4.4. Újramintavételezés

Próbáljuk ki hangfileokkal! Ügyeljünk arra, hogy a FM sávszélessége nagy lesz, ezért nagyobb helyre lesz szükségünk a frekvenciatartományban. Ha megváltoztatjuk a mintavételi frekvenciát és a hangot újramintavételezzük ez alapján, ki tudjuk terjeszteni a frekvenciatartomány méretét.
Vigyázzunk, hogy n-szeres mintavételi frekvenciánál a minták száma is n-szeresére változik, és ez nagyon megnövelheti a számítási időt.
Az újramintavételezés a *resample(x, p, q)* függvénnyel tehető meg, ami $\frac{p}{q}$-szorosára változtatja a mintavételi frekvenciát (ehhez be kell tölteni a signal csomagot). Ne felejtsük a F_s változót is beállítani!

Kód 4. Újramintavételezés

```
[x,Fs] = audioread('voice.wav');
% Ujramintavetelezes -> nagyobb frekitartomany
Fs = Fs * 4;
x = resample(x, 4, 1);
N = length( x ); l = N / Fs;
```