

Esercitazione n° 3 – Decimazione e sovracampionamento

Scelta della frequenza di campionamento.

- 1) Visualizzare una forma d'onda sinusoidale di frequenza 1 kHz con una frequenza di campionamento di 20 kHz tra 0 e 20 ms. (20 punti per periodo).
- 2) Provare a diminuire la frequenza di campionamento e visualizzare la forma d'onda corrispondente. (Es. 10, 6, 4 punti per periodo --- $f_s=10$ kHz, 6 kHz, 4kHz),
- 3) Provare a spingere la frequenza di campionamento sotto il limite di Nyquist (Es. $f_s=1200$)

Aliasing nel dominio del tempo

- 4) Visualizzare contemporaneamente il grafico per $f_s=20$ kHz e $f_s=1.2$ kHz.

Decimazione e sovracampionamento

- 5) Scrivere una funzione MATLAB che effettua una decimazione per un fattore N.
`function y=decimate(x,N)`
- 6) Scrivere una funzione MATLAB che effettua un sovracampionamento per un fattore N.
(Inserire N-1 zeri tra ogni coppia di campioni;
`function y=upsample(x,N)`)

Aliasing nel dominio della frequenza

- 7) Costruire un segnale a banda limitata con risposta in frequenza triangolare. ($f_s=20$ kHz, $T_{oss} = 20$ ms, frequenza centrata su 5 kHz e banda 2 kHz).
- 8) Diminuire la frequenza di campionamento ($f_s=14$ kHz, $f_s = 12$ kHz);

Decimazione e sovracampionamento nel dominio della frequenza.

- 9) Applicare la decimazione al segnale a banda limitata ($f_s=50$ kHz, $N=2,3,4$)
- 10) Applicare il sovracampionamento al segnale a banda limitata ($f_s=20$ kHz, $N=2,3$);

Soluzioni

1)

```
t=linspace(0,0.02,400); % N=fs*Toss=400
x=sin(2*pi*1000*t);
plot(t,x);
```

2)

```
t=linspace(0,0.02,200); % N = 20 (periodi) * 10 punti
x=sin(2*pi*1000*t);
plot(t,x)
```

```
t=linspace(0,0.02,120); % N = 20 (periodi) * 6 punti
x=sin(2*pi*1000*t);
plot(t,x)
```

```
t=linspace(0,0.02,80); % N = 20 (periodi) * 4 punti
x=sin(2*pi*1000*t);
plot(t,x)
```

3)

```
t=linspace(0,0.02,24); % N = fs*Toss=24
x=sin(2*pi*1000*t);
plot(t,x)
```

4)

```
t=linspace(0,0.02,400);
x=sin(2*pi*1000*t);
plot(t,x);
hold on
t=linspace(0,0.02,24);
x=sin(2*pi*1000*t);
plot(t,x);
hold on
```

5)

```
function y=decimate(x,N)
y=x(1:N:length(x));
```

6)

```
function y=upsample(x,N)
y=zeros(1,length(x)*N);
y(1:N:length(y))=x;
```

7)

```
t=linspace(0,0.02,400);
x=(sinc(2000*(t-0.01)).^2).*sin(2*pi*5000*t);
f=linspace(0,20,length(x));
plot(f,abs(fft(x)));
```

8)

```
t=linspace(0,0.02,280);
```

```
x=(sinc(2000*(t-0.01)).^2).*sin(2*pi*5000*t);
f=linspace(0,14,length(x));
plot(f,abs(fft(x)));
```

```
t=linspace(0,0.02,260);
x=(sinc(2000*(t-0.01)).^2).*sin(2*pi*5000*t);
f=linspace(0,13,length(x));
plot(f,abs(fft(x)));
```

9)

```
t=linspace(0,0.02,1000);
x=(sinc(2000*(t-0.01)).^2).*sin(2*pi*5000*t);
f=linspace(0,50,length(x));
plot(f,abs(fft(x)));
```

```
y=decimate(x,2);
f2=linspace(0,50/2,length(y));
```

```
plot(f2,abs(fft(y)));
```

10)

```
t=linspace(0,0.02,400);
x=(sinc(2000*(t-0.01)).^2).*sin(2*pi*5000*t);
f=linspace(0,20,length(x));
plot(f,abs(fft(x)));
```

```
y=upsample(x,2);
f2=linspace(0,20*2,length(y));
plot(f2,abs(fft(y)));
```