

Matlab Practice 02

Discrete Signal and Systems

Sangkeun Lee

Unit impulse Signal

● 단위 임펄스 신호 $\delta[n]$

- 단위 샘플 신호 또는 크로네커 델타Kronecker delta 라고도 함

$$\delta[n] = \begin{cases} 1, & n = 0 \\ 0, & n \neq 0 \end{cases}$$

- 이산 신호는 임펄스 신호의 가중합 형태로 나타낼 수 있음

$$x[n] = \cdots + x[-1]\delta[n+1] + x[0]\delta[n] + \cdots + x[k]\delta[n-k] + \cdots$$

$$= \sum_{k=-\infty}^{\infty} x[k]\delta[n-k]$$

- 임펄스 함수의 체 거르기(sifting) 또는 샘플링 성질
신호 값 중에서 특정 시각(n)의 신호 값($x[n]$)만 취하는 것으로 해석

Unit Function

● 단위 임펄스 신호 $u[n]$

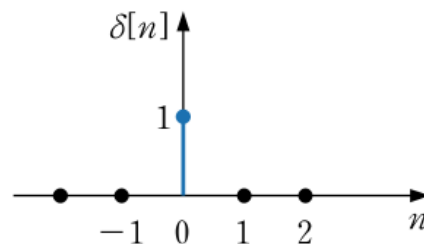
- $n \geq 0$ 에서 값이 일정(=1)한 직류 신호

$$u[n] = \begin{cases} 1, & n \geq 0 \\ 0, & n < 0 \end{cases}$$

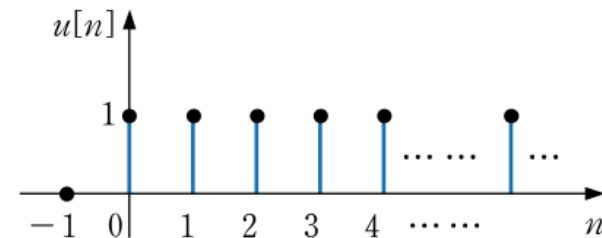
- 다른 신호에 대한 on-off 스위치 역할 → 신호의 표현에 이용
- $\delta[n]$ 과 $u[n]$ 의 관계

$$\delta[n] = u[n] - u[n-1]$$

$$u[n] = \sum_{k=-\infty}^n \delta[k]$$

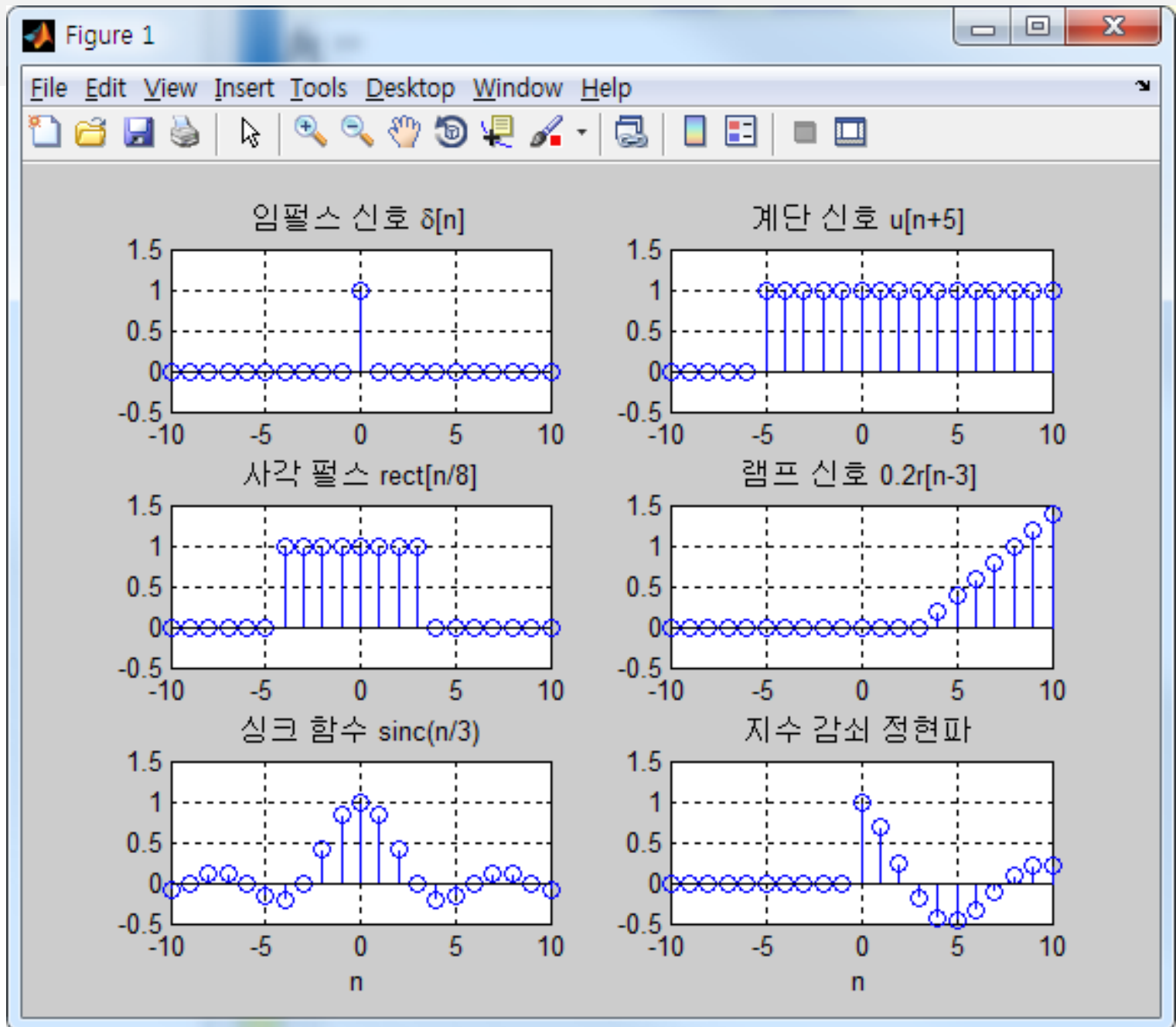


[그림 2-2] 이산 단위 임펄스 신호



[그림 2-3] 이산 단위 계단 신호

Result



code

```

n=[-10:1:10];
xa=(n==0);
xb=stepfun(n,-5);
xc=rectpuls(n/8);
xd=0.2*(n-3).*(n>=3);
xe=sinc(n/3);
xf=exp(-0.15*n).*cos(0.2*pi*n).*(n>=0);

subplot(3,2,1);
stem(n,xa);
axis([-10 10 -0.5 1.5]); grid on;
title('임펄스 신호 #delta[n]');
subplot(3,2,2);
stem(n,xb);
axis([-10 10 -0.5 1.5]); grid on;
title('계단 신호 u[n+5]');
subplot(3,2,3);
stem(n,xc);
axis([-10 10 -0.5 1.5]); grid on;
title('사각 펄스 rect[n/8]');
subplot(3,2,4);
stem(n,xd);
axis([-10 10 -0.5 1.5]); grid on;
title('램프 신호 0.2r[n-3]');
subplot(3,2,5);
stem(n,xe);
axis([-10 10 -0.5 1.5]); grid on;
title('싱크 함수 sinc(n/3)');
xlabel('n');
subplot(3,2,6);
stem(n,xf);
axis([-10 10 -0.5 1.5]); grid on;
title('지수 감쇠 정현파');
xlabel('n');

```

% 시간축 설정(n은 -10에서 10)까지 1씩 증가)
 % n=0일 때 1, 즉 임펄스
 % 계단 신호
 % 사각 펄스
 % 램프 함수
 % 싱크 함수
 % 지수 감쇠 정현파

 % 3행 2열 분할 그림 창의 1번 창
 % 미산 신호 출력(n은 x축, xa는 y축)
 % x축과 y축 영역 설정, 그리드 표시
 % 그림 제목('#xx' 는 정해진 기호 표시)
 % 3행 2열 분할 그림 창의 2번 창
 % 미산 신호 출력
 % x축과 y축 영역 설정, 그리드 표시
 % 그림 제목
 % 3행 2열 분할 그림 창의 3번 창
 % 미산 신호 출력
 % x축과 y축 영역 설정, 그리드 표시
 % 그림 제목
 % 3행 2열 분할 그림 창의 4번 창
 % 미산 신호 출력
 % x축과 y축 영역 설정, 그리드 표시
 % 그림 제목
 % 3행 2열 분할 그림 창의 5번 창
 % 미산 신호 출력
 % x축과 y축 영역 설정, 그리드 표시
 % 그림 제목
 % x축 라벨
 % 3행 2열 분할 그림 창의 6번 창
 % 미산 신호 출력
 % x축과 y축 영역 설정, 그리드 표시
 % 그림 제목
 % x축 라벨

Code 1/2

n=[-10:1:10];	% 시간축 설정
x=4*stepfun(n,0)-(n-2).*(n>=2)+(n-6).*(n>=6);	% x[n] 생성
xp0=x(11:16);	% x[n]의 한 주기 파형 선택
for k=1:21	
if rem(k,6)==5	% 주기신호 xN[n] 생성()
xp(k)=xp0(1);	
else	% 주기신호 xN[n] 생성(그 외의)
xp(k)=xp0(rem(k,6)+2);	
end	
end	
xs=4*stepfun(n,-3)-(n+1).*(n>=-1)+(n-3).*(n>=3);	% x[n+3] 생성
xr=flip1r(x);	% x[-n] 생성 (y축 대칭)
xe=0.5*(x+xr);	% 우함수 대칭 성분 xe[n] 생성
xo=0.5*(x-xr);	% 기함수 대칭 성분 xo[n] 생성
Ex=sum(x.*x);	% 신호 x[n]의 에너지 계산
Pxp=sum(xp0.*xp0)/6;	% 신호 xN[n]의 전력 계산

Code 2/2

subplot(3,2,1);	% 3행 2열 분할 그림 창의 1번 창
stem(n,x);	% x[n] 그림
title('x[n]');	% 그림 제목
axis([-10 10 -1 5]);	% x축과 y축의 그림 영역 지정
subplot(3,2,2);	% 3행 2열 분할 그림 창의 2번 창
stem(n,xp);	% xN[n] 그림
title('x_N[n]');	% 그림 제목
axis([-10 10 -1 5]);	% x축과 y축의 그림 영역 지정
subplot(3,2,3);	% 3행 2열 분할 그림 창의 3번 창
stem(n,xs);	% x[n+3] 그림
title('x[n+3]');	% 그림 제목
axis([-10 10 -1 5]);	% x축과 y축의 그림 영역 지정
subplot(3,2,4);	% 3행 2열 분할 그림 창의 4번 창
stem(n,xr);	% x[-n] 그림
title('x[-n]');	% 그림 제목
axis([-10 10 0 5]);	% x축과 y축의 그림 영역 지정
subplot(3,2,5);	% 3행 2열 분할 그림 창의 5번 창
stem(n,xe);	% xe[n] 그림
title('x_e[n]');	% 그림 제목
axis([-10 10 -1 5]);	% x축과 y축의 그림 영역 지정
subplot(3,2,6);	% 3행 2열 분할 그림 창의 6번 창
stem(n,xo);	% xo[n] 그림
title('x_o[n]');	% 그림 제목
axis([-10 10 -3 3]);	% x축과 y축의 그림 영역 지정