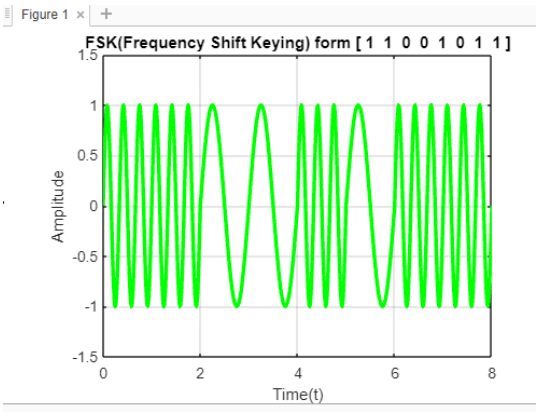
1. Frequency Shift Keying .

```
%Frequency Shift Keying matlab code
clear all;
close all;
clc;

f1=1;
f2=3;
fs=100;
t=0: 1/fs: 1;
bit=[1 1 0 0 1 0 1 1];
time=[];
FSK_Signal=[];
for i=1:1: length(bit)
   FSK_Signal=[FSK_Signal (bit(i)==0)*sin(2*pi*f1*t)...
      + (bit(i)==1)*sin(2*pi*f2*t)];
   time=[time t];
   t=t +1;
end
plot(time,FSK_Signal,'green','LineWidth',2.5);
axis([0 time(end) -1.5 1.5]);
xlabel('Time(t)');
title(['FSK(Frequency Shift Keying) form [ ' num2str(bit) ' ]']);
ylabel('Amplitude');
grid on;
```

output:

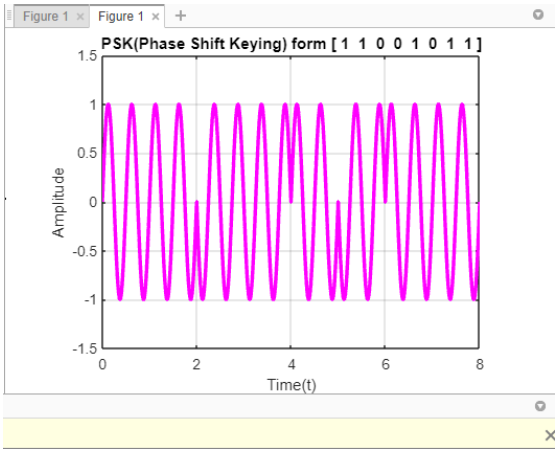


2. Phase Shift Keying .

code:

```
clear all;
close all;
clc;

f1=2;
fs=100;
t=0: 1/fs: 1;
bit=[1 1 0 0 1 0 1 1];
time=[];
psk_Signal=[];
for i=1:1: length(bit);
   psk_Signal=[psk_Signal (bit(i)==0)*-sin(2*pi*f1*t)...
      + (bit(i)==1)*sin(2*pi*f1*t)];
   time=[time t];
   t=t +1;
end
plot(time,psk_Signal,'magenta','LineWidth',2.5);
axis([0 time(end) -1.5 1.5]);
xlabel('Time(t)');
title(['PSK(Phase Shift Keying) form [ ' num2str(bit) ' ]']);
ylabel('Amplitude');
grid on;
```

output:



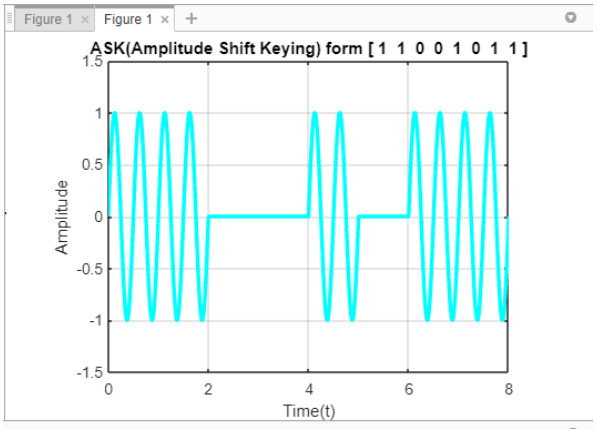
3. Amplitude Shift Keying.

code:

```
clear all;

close all;

clc;


f1=2;

fs=100;

t=0: 1/fs: 1;

bit=[1 1 0 0 1 0 1 1];

time=[];

ask_Signal=[];

for i=1:1: length(bit);

   ask_Signal=[ask_Signal (bit(i)==0)*...

      zeros(1,length(t))+ (bit(i)==1)*sin(2*pi*f1*t)];

   time=[time t];
```

```
    t=t +1;
end
plot(time,ask_Signal,'cyan','LineWidth',2.5);
axis([0 time(end) -1.5 1.5]);
xlabel('Time(t)');
title(['ASK(Amplitude Shift Keying) form [ ' num2str(bit) ' ]']);
ylabel('Amplitude');
grid on;
```

output:



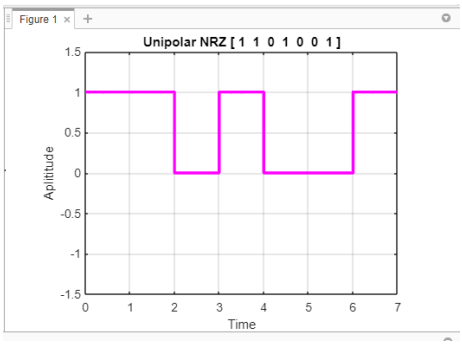
4. Unipolar Not return to zero(NRZ)

code:

```
%UnipolarNRZ matab code
clear all;
close all;
clc;
fs=100;
t=0:1/fs:1;
```

```
bit=[1 1 0 1 0 0 1];

time = [];

Digital_Signal = [];

for  i = 1: 1 : length(bit)

  Digital_Signal = [Digital_Signal (bit(i)==0)*zeros(1,length(t))+(bit(i)==1)*ones(1,length(t))];

  time = [time t];

  t = t+1;


end

subplot(1,1,1);

plot(time,Digital_Signal,'magenta','linewidth',2.5);

title(['Unipolar NRZ [ ' num2str(bit) ' ]']);

xlabel('Time');

ylabel('Aplititude');

axis([0 time(end) -1.5 1.5]);

grid on;
```

output:



5.Manchester line coding .

code:

```
clear all;
close all;
clc;
fs=100;
t=0: 1/fs:1;
bit=[1 1 0 1 0 0 1];
nbit=[];
Time=[];
Dig_Sig=[];
k=1;
for i=1: 1 : length(bit)
   if(bit(i)==1)
      nbit(k)=1;
      nbit(k+1)=-1;
   end
   if(bit(i)==0)
      nbit(k)=-1;
      nbit(k+1)=1;
   end
   k=k+2;
end
```

```
for i=1: 1 : length(nbit)
   Dig_Sig=[Dig_Sig (nbit(i)==1)*...
      ones(1,length(t)) + (nbit(i)==-1)*-ones(1,length(t))];
   Time=[Time t]; t=t+1;
end
subplot(1,1,1);
plot(Time,Dig_Sig, 'black','LineWidth',2.5);
axis([ 0 Time(end) -1.5 1.5]);
grid on;
xlabel('Time');
ylabel('Amplitude');
title(['Manchester [ ' num2str(bit) ' ]']);
```

code: