Mp3 Project

In [1]:

```
fileName='wavs/8k8bitpcm.wav';
[y, fs]=wavread(fileName);
```

In [2]:

```
interval = 1:5000;
```

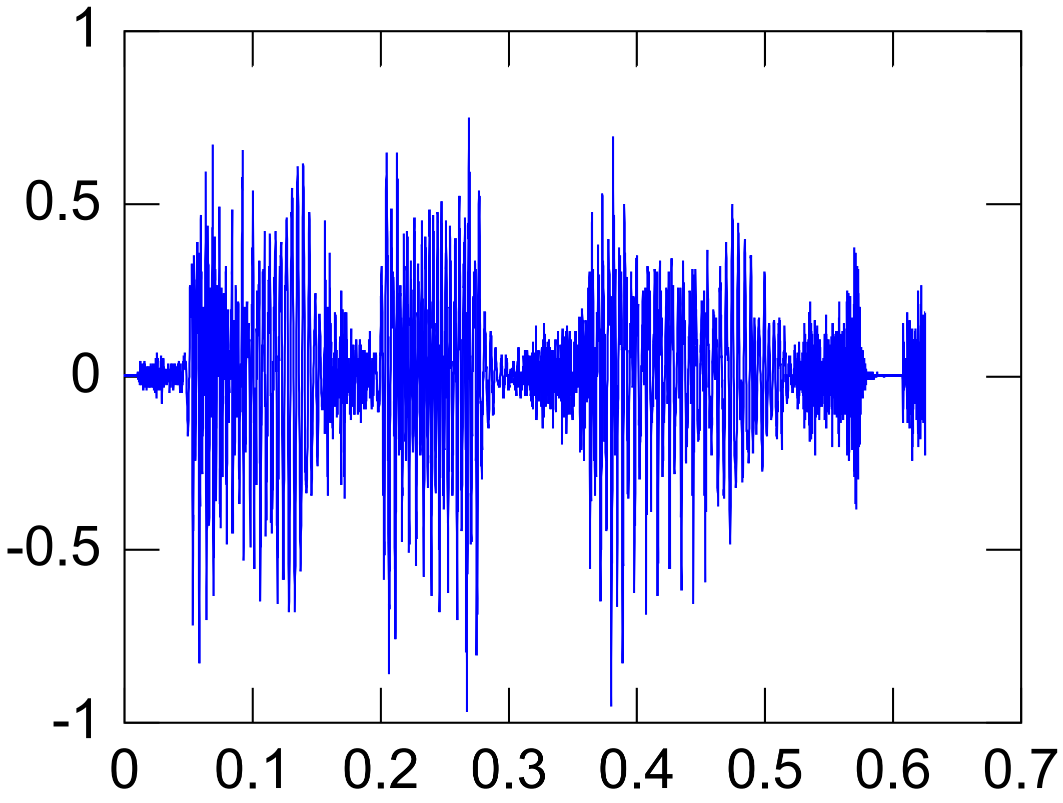
In [3]:

```
time = (1:length(y(interval))) / fs;
```

Quantization

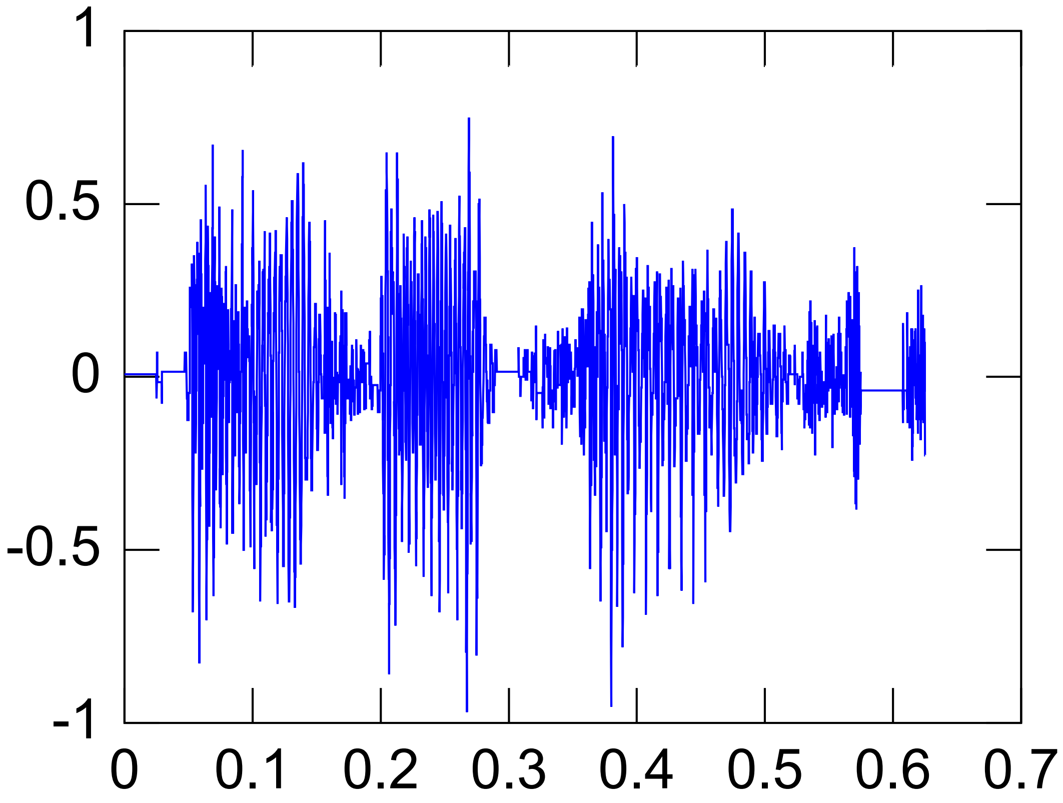
In [4]:

```
plot(time,y(interval))
```



In [5]:

```
plot(time, filterAndQuantize(y(interval)))
```



Huffman Encoding

In [6]:

```
[encodedFre, fres, huffTable] = huffEncodeFre(y(interval)');
disp('Not quantized huffman')
disp(length(encodedFre));
```

```
Not quantized huffman
 32731
```

In [7]:

```
disp('Quantized huffman')
[encodedFre, fres, huffTable] = huffEncodeFre(filterAndQuantize(y(interva
l))');
disp(length(encodedFre));
```

```
Quantized huffman
 32037
```

In [8]:

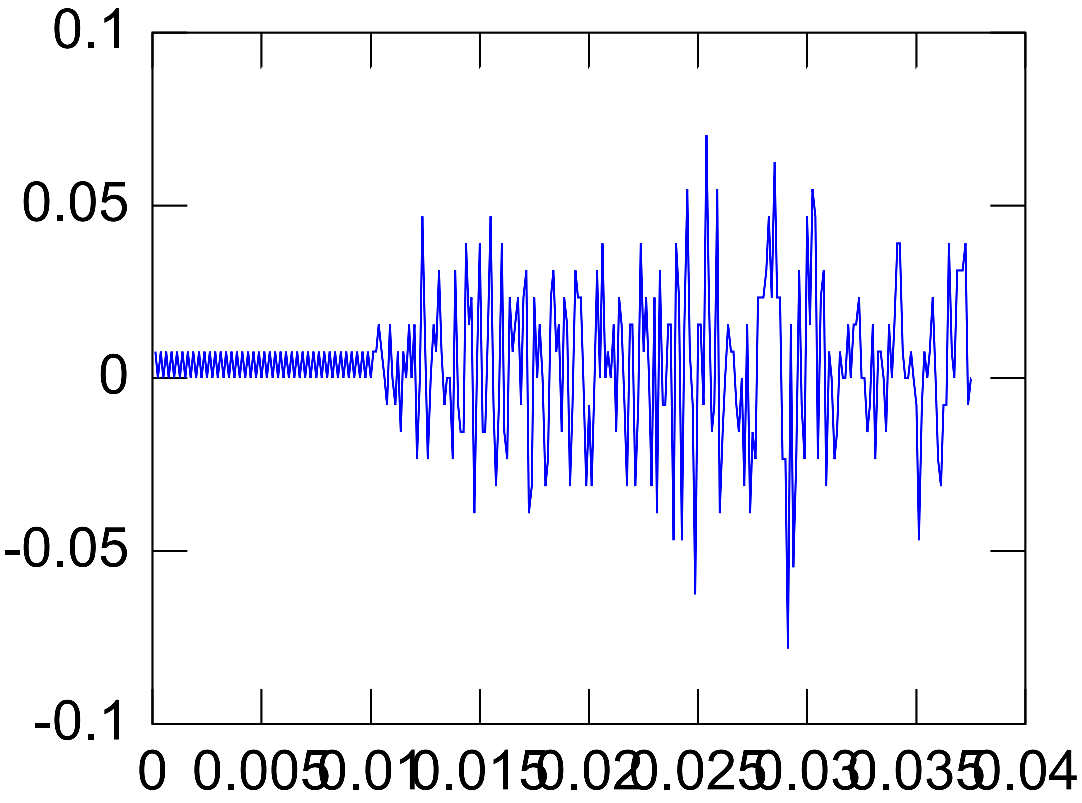
```
disp('Real File size')
disp(length(y(interval)) * 17);
```

```
Real File size
 85000
```

Decoding

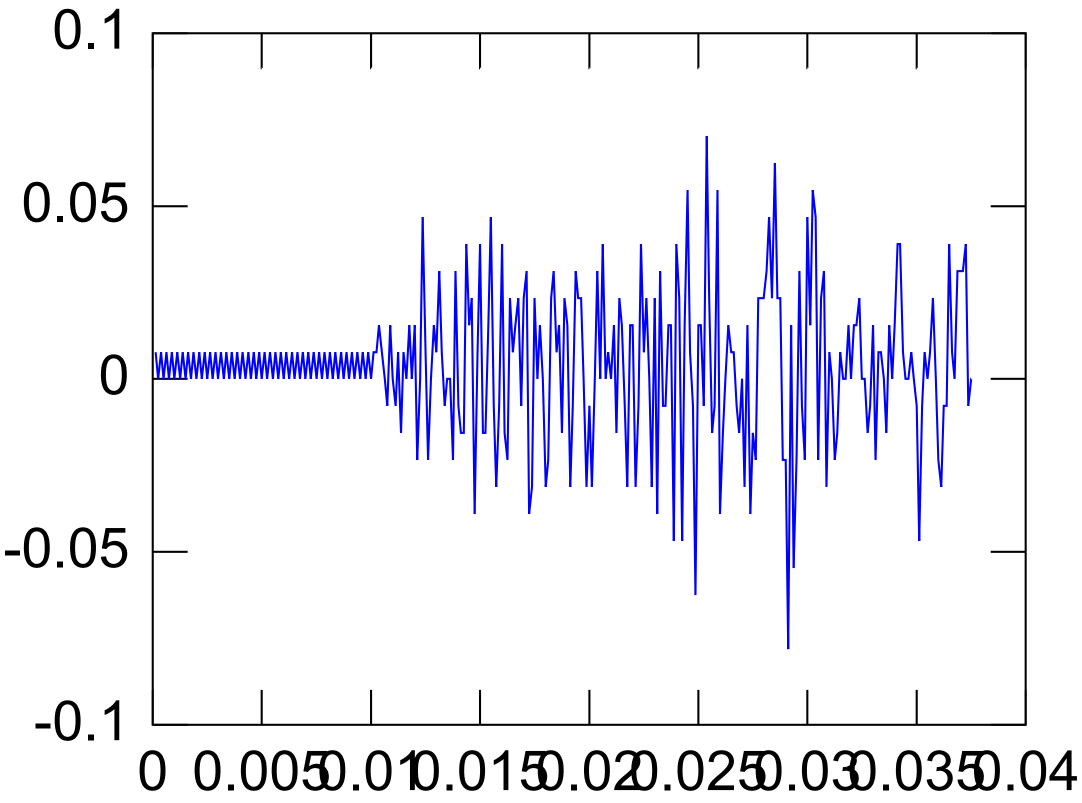
In [14]:

```
interval = 1:300;
time = (1:length(y(interval))) / fs;
plot(time, y(interval));
[encodedFre, fres, huffTable] = huffEncodeFre(y(interval)');
```



In [15]:

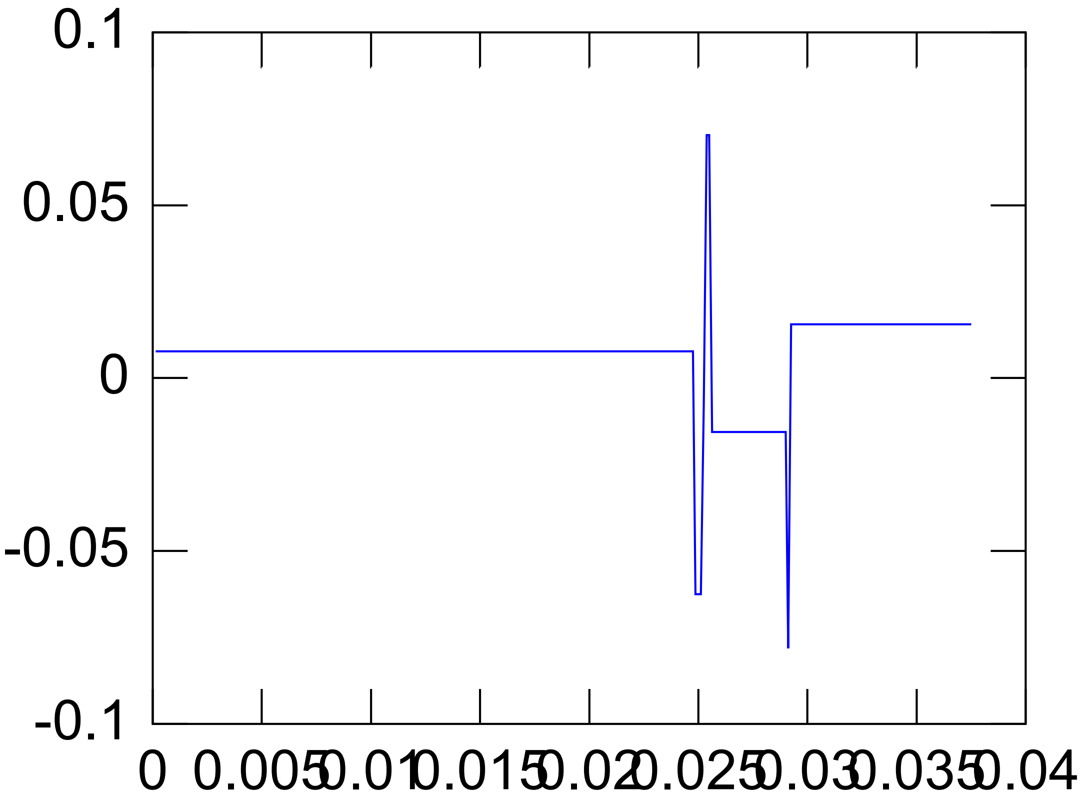
```
decodeY = huffDecodeFre(encodedFre, fres, huffTable);
plot(time, decodeY);
```



Lossy Decoding

```
In [16]:
```

```
[encodedFre, fres, huffTable] = huffEncodeFre(filterAndQuantize(y(interva
l))');
decodeY = huffDecodeFre(encodedFre, fres, huffTable);
plot(time, decodeY);
```


0.1
0.05
0
-0.05
-0.1
0
0.005
0.01
0.015
0.02
0.025
0.03
0.035
0.04