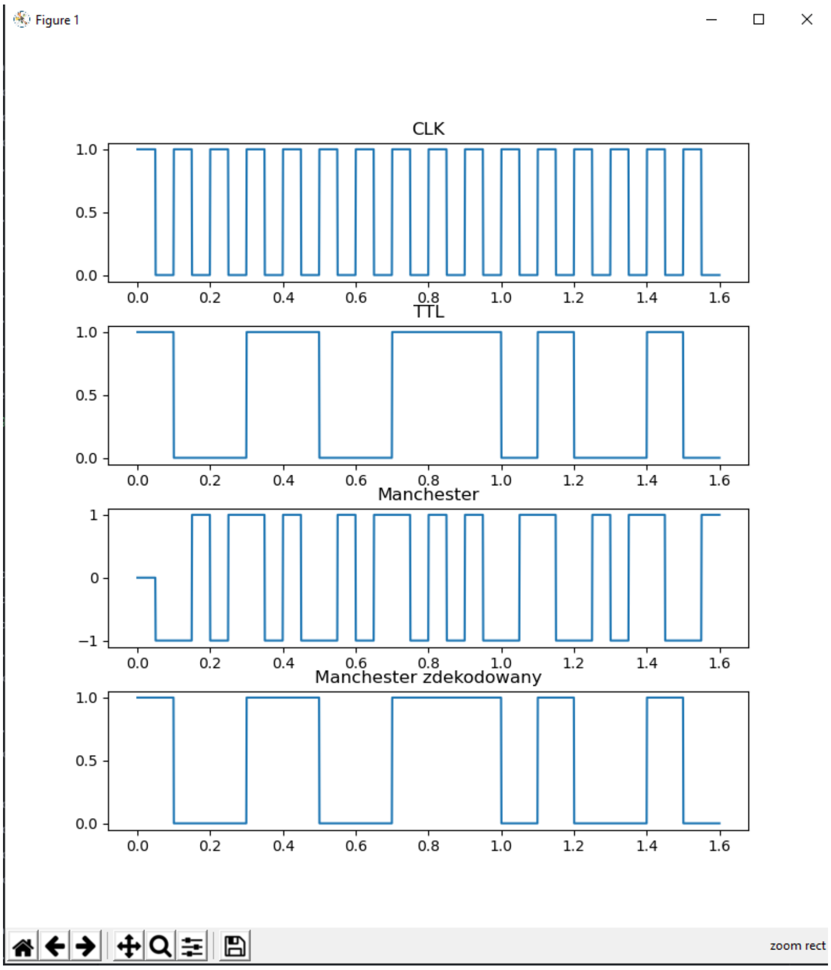
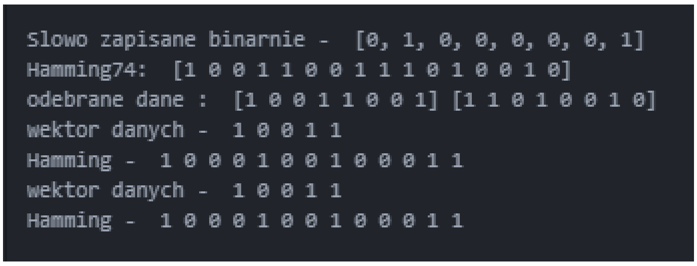
Sprawozdanie Lab09

Autor: Kamil Szóstak

```
import numpy as np
import matplotlib.pyplot as plt
import cmath
import math
import pylab
from scipy.interpolate import interp1d

from Utilities import Binary, Hamming_SECDED, Hamming_Dekoder_SECDED
from Utilities import CLK,info_Signal,Manchester,DekoderManchester,info_Signal

word = Binary('a')
#Test
word = [0,1,0,0,0,0,0,1]
[x,wynik]=Hamming_SECDED(word)



clk = CLK(100, len(wynik))
t, ttl = info_Signal(0.1, wynik, 100)
manchesterSamples = Manchester(clk, ttl)
manchesterBits = DekoderManchester(clk, ttl, 100)
tzz, manchestersTtl = info_Signal(0.1, manchesterBits, 100)
mb=np.array(manchesterBits)
[a1,a2] = np.array_split(mb, 2)
print('odebrane dane : ',a1,a2)
Hamming_Dekoder_SECDED(word,a1)
Hamming_Dekoder_SECDED(word,a2)
```

```
plt.figure()
plt.subplot(411)
plt.title('CLK')
plt.plot(t, clk)
plt.subplot(412)
plt.title('TTL')
plt.plot(t, ttl)
plt.subplot(413)
plt.title('Manchester')
plt.plot(t, manchesterSamples)
plt.subplot(414)
plt.title('Manchester zdekodowany')
plt.plot(tzz, manchestersTtl)
plt.show()
```





```
Slowo zapisane binarnie -  [0, 1, 0, 0, 0, 0, 0, 1]
Hamming74:  [1 0 0 1 1 0 0 1 1 1 0 1 0 0 1 0]
odebrane dane :  [1 0 0 1 1 0 0 1] [1 1 0 1 0 0 1 0]
wektor danych -  1 0 0 1 1
Hamming -  1 0 0 0 1 0 0 1 0 0 0 1 1
wektor danych -  1 0 0 1 1
Hamming -  1 0 0 0 1 0 0 1 0 0 0 1 1
```