Import needed libraries:

`numpy` - for working with arrays

`matplotlib` - for plotting

In [11]:
```
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
```

Load the text files containing the total alpha and beta DOS for all atoms and convert them to numeric arrays

In [12]:
```
alp_tot = np.array(np.loadtxt("s2-pspw-energy.mulliken_dos_alpha_all"))
bet_tot = np.array(np.loadtxt("s2-pspw-energy.mulliken_dos_beta_all"))
```

This is how the arrays look like:

In [13]:
```
print(alp_tot)
```

```
[[-8.5449e-01  1.5742e-01  1.1965e-04]
 [-8.5298e-01  1.6170e-01  3.6547e-04]
 [-8.5147e-01  1.6618e-01  6.1809e-04]
 ...
 [-1.0329e-01  3.2557e-01  1.4005e+01]
 [-1.0178e-01  3.1695e-01  1.4005e+01]
 [-1.0027e-01  3.0869e-01  1.4006e+01]]
```

Now, we split the table into the corresponding columns (to be the 1D arrays):

- first - energy
- second - the DOS of the first sulphur atom, S1
- third - the DOS of the second sulphur atom, S2

Then we do some manipulations - e.g. compute the total alpha or beta DOS of all atoms:

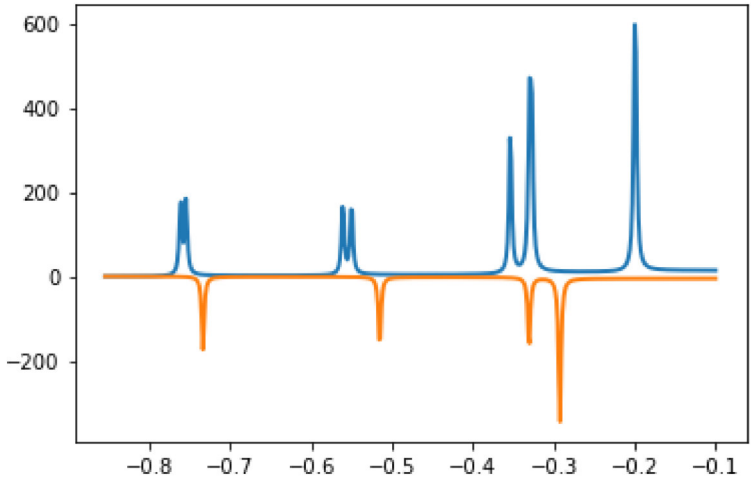
In [14]:
```
E = alp_tot[:, 0]
Sa1 = alp_tot[:, 1]
Sa2 = alp_tot[:, 2]
Sa = Sa1 + Sa2

Sb1 = bet_tot[:, 1]
Sb2 = bet_tot[:, 2]
Sb = (Sb1 + Sb2)
```

We can now plot what we want:

In [15]:
```
plt.plot(E, Sa, lw = 2, label="alpha")
plt.plot(E, Sb, lw = 2, label="beta")
```

Out[15]: [<matplotlib.lines.Line2D at 0x14576c03fd60>]



In [ ]: