In [58]:
```
%pylab notebook

import postpic as pp
import numpy as np
import numexpr as ne
```

```
Populating the interactive namespace from numpy and matplotlib
```

In [59]:
```
# create example function with 2 oscillations with coefficients
1 and -2j

x = np.linspace(0, 2*np.pi, 8, endpoint=False)
y = ne.evaluate('exp(1j * x) - 2.j * exp(1j * 2 * x)')

# put this into a field object
y = pp.Field(y, axes=[pp.Axis(grid=x)])
```

In [60]:
```
# perform fft and change normalization

yf = y.fft()/np.sqrt(2*np.pi)
```

In [56]:
```
# print result

for xi, yi in zip(yf.grid, yf.matrix):
    print('{:f}\t{:f}'.format(xi, yi))
```

```
-4.000000       0.000000-0.000000j
-3.000000       0.000000-0.000000j
-2.000000       0.000000+0.000000j
-1.000000       0.000000+0.000000j
0.000000        0.000000+0.000000j
1.000000        1.000000+0.000000j
2.000000        -0.000000-2.000000j
3.000000        0.000000-0.000000j
```

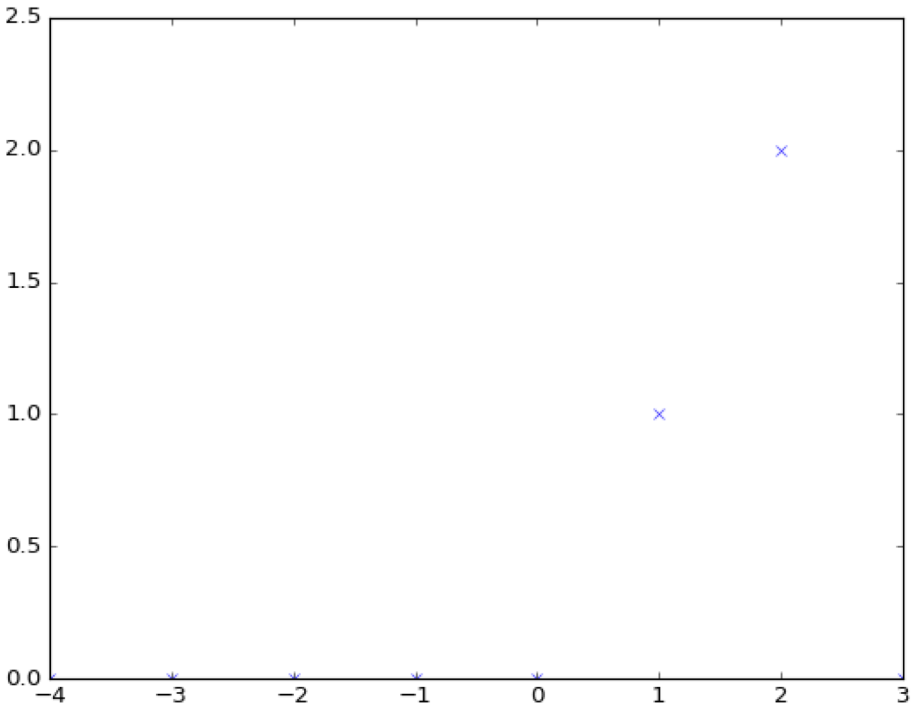
In [62]:
```
# test if result is according to expectation

np.allclose(1, yf[1.].matrix), np.allclose(-2j, yf[2.].matrix)
```

Out[62]:
```
(True, True)
```

In [63]:
```
# also plot spectrum

fig, ax = plt.subplots()
ax.plot(yf.grid, abs(yf), 'x')
```




Out[63]: [<matplotlib.lines.Line2D at 0x7febc6e623c8>]

In [ ]: