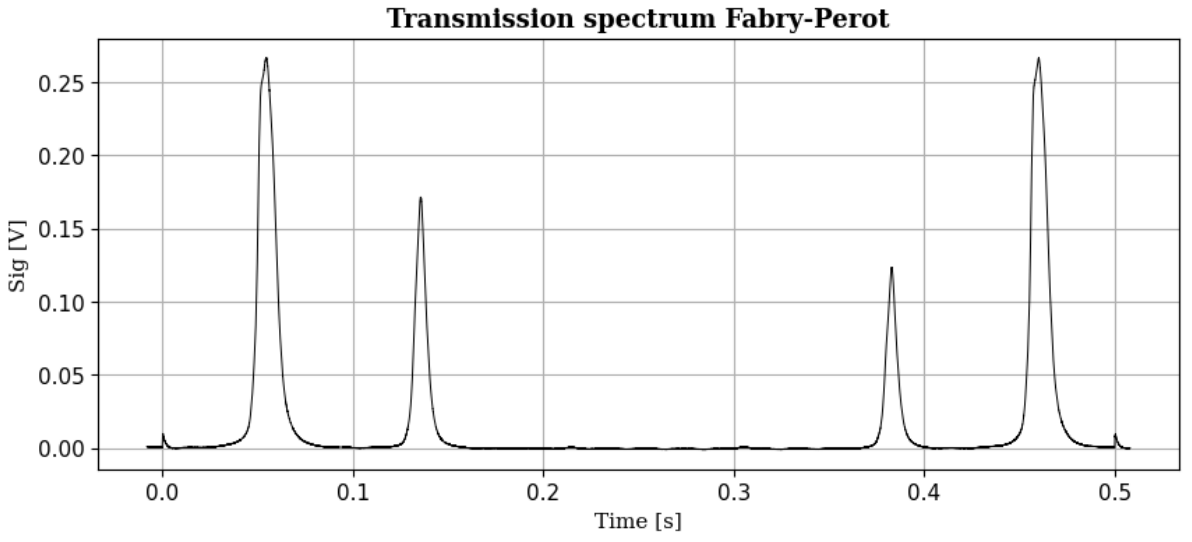
In [1]:
```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
from FabryPerot import SpectrumAnalyzer as SA

MEDIUM_SIZE = 11
BIGGER_SIZE = 13

plt.rc('font', size=MEDIUM_SIZE)          # controls default text sizes
plt.rc('axes', titlesize=BIGGER_SIZE)     # fontsize of the axes title
plt.rc('axes', labelsize=MEDIUM_SIZE)    # fontsize of the x and y labels
plt.rc('xtick', labelsize=MEDIUM_SIZE)    # fontsize of the tick labels
plt.rc('ytick', labelsize=MEDIUM_SIZE)    # fontsize of the tick labels
plt.rc('legend', fontsize=MEDIUM_SIZE)    # legend fontsize

base_font = {'family': 'serif',
        'size': MEDIUM_SIZE,
        }

title_font = {
        'family': 'serif',
        'color':  'black',
        'size': BIGGER_SIZE,
        'weight' : 'bold'
        }
```

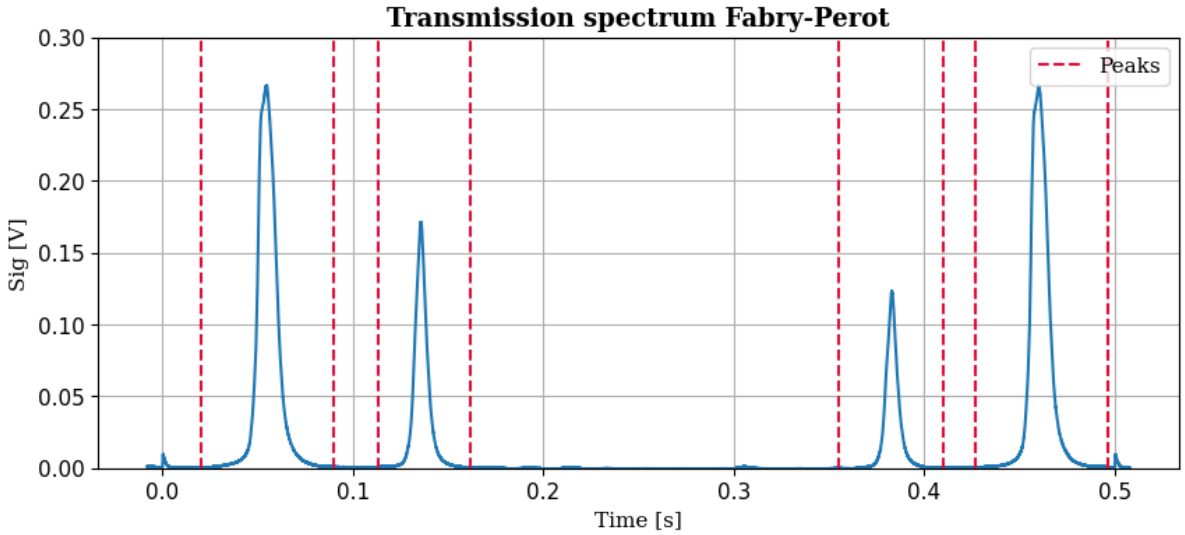
In [2]:
```
spectrum = SA('Farby_Perot_transmission_signal')
```

In [3]:
```
spectrum.plot_spectrum()
plt.show()
```



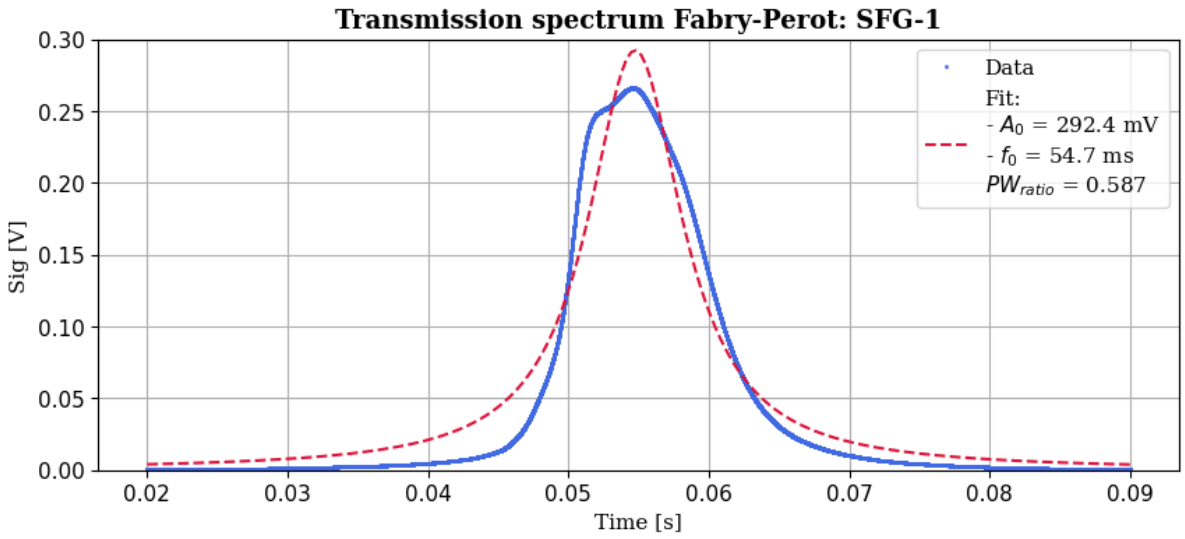
In [4]:
```
spectrum.select_peaks(
  [
    [0.02, 0.09],
    [0.113, 0.162],
    [0.355, 0.41],
    [0.427, 0.496]
```

```
  ]
)
```

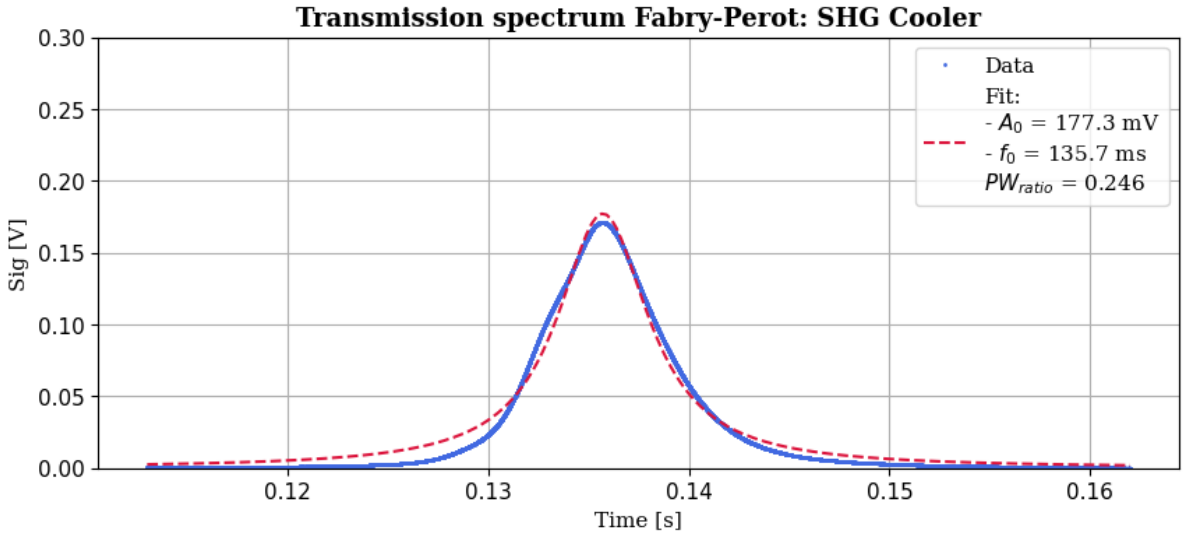


In [5]:
```
names = ['SFG-1', 'SHG Cooler', 'SHG Rep', 'SFG-2']
tot_pw = 0
for i, peak in enumerate(spectrum.peaks[:-1]):
  peak.integrate()
  tot_pw = tot_pw + peak.Integral

spectrum.Tot_PW = tot_pw

for i, peak in enumerate(spectrum.peaks):
  peak.integrate()
  print('--------------------------------------------------------------------------------
  peak.fit(interval=[peak.Time[0], peak.Time[-1]], plot=True, tot_pw=spectrum.Tot_P
```

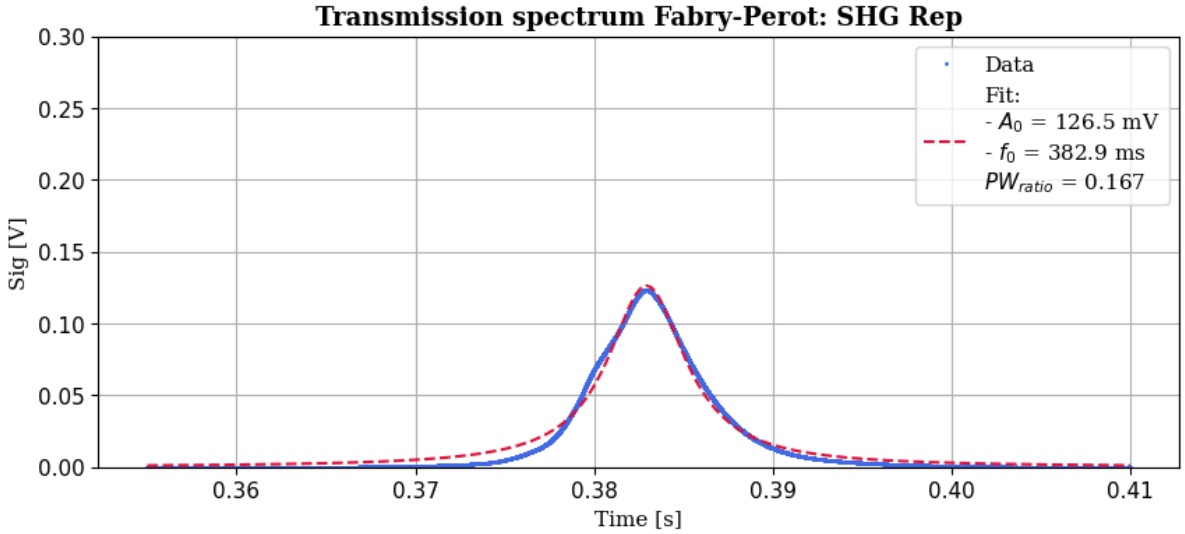
```
--------------------------------------------------------------------------------
-------

Peak height = 292 mV, Peak Pos = 55 ms, PW Ratio = 0.587
```



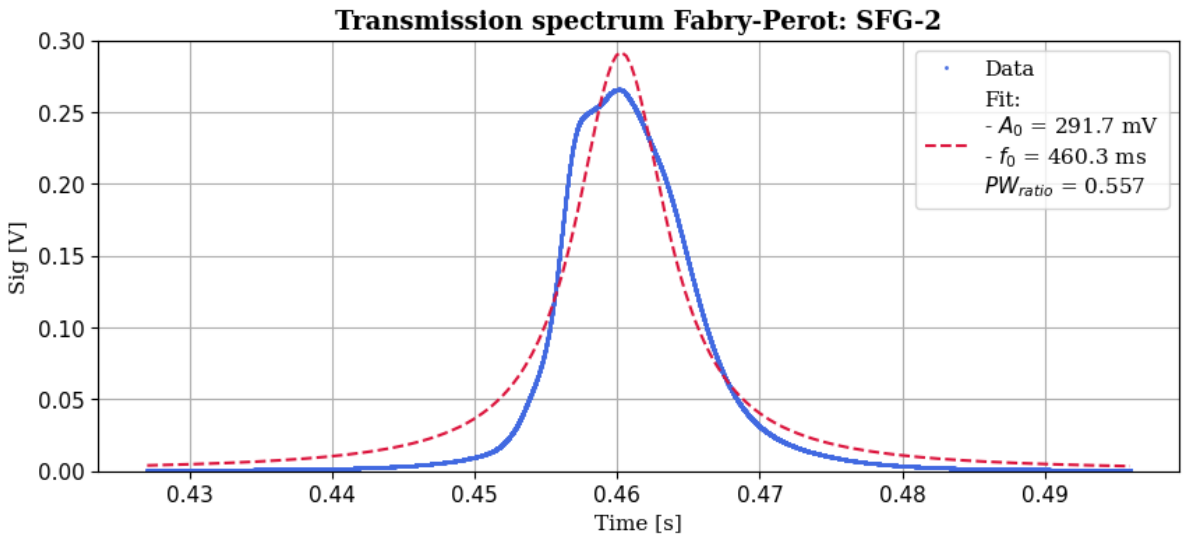
Peak height = 177 mV, Peak Pos = 136 ms, PW Ratio = 0.246



Peak height = 126 mV, Peak Pos = 383 ms, PW Ratio = 0.167



Peak height = 292 mV, Peak Pos = 460 ms, PW Ratio = 0.557



In [6]:
```
TimeToFreq = 1.5 / (spectrum.peaks[3].peak_pos - spectrum.peaks[0].peak_pos)
spectrum.Set_TimeToFreq_conv(TimeToFreq)
ax = spectrum.plot_spectrum()
ax.set_ylim(0, 0.35)
color_palette = plt.cm.jet(np.linspace(0.0, 0.35, 3))
for i, peak in enumerate(spectrum.peaks[:-1]):
  ax.text(x=(peak.peak_pos -0.02) * TimeToFreq, y=peak.height, s=names[i], fontdict
  ax.fill_between(peak.Time * TimeToFreq, peak.Sig, color=color_palette[i % 3], alp

ax.set_title('Transmission spectrum Fabry-Perot (FSR = 1.5 GHz)', fontdict=title_fo
ax.legend(loc='upper center')
plt.show()
```