In [8]:

```python
import numpy as np
import matplotlib.pyplot as plt
fs = round(1/((t_end-t_start)/len(a)))
print(f"Calculated samplerate fs = {fs} kS/s")
t = np.linspace(0, t_end-t_start, len(a))

fig, ax = plt.subplots(3, 1)
ax = ax.flatten()
ax[0].plot(t, a)
ax[0].set_title("Original signal")
ax[0].set_xlabel("Time [s]")
ax[0].set_ylabel("Voltage [V]")
plt.tight_layout()
plt.grid()
plt.minorticks_on()
#plt.show()

a_fft = np.fft.fft(a)           # Original FFT
a_fft = a_fft[1:]                 # Delete DC component
a_fft = a_fft[:round(len(t)/2)] # First half ( pos freqs )
a_fft = np.abs(a_fft)           # Absolute value of magnitudes
a_fft = a_fft/max(a_fft)        # Normalized so max = 1

freq_x_axis = np.linspace(1, fs/2, len(a_fft))
ax[1].plot(freq_x_axis, a_fft, "o-")
ax[1].set_title("Frequency magnitudes")
ax[1].set_xlabel("Frequency [Hz]")
ax[1].set_ylabel("Magnitude")
#plt.grid()
#plt.minorticks_on()
#plt.show()


f_loc = np.argmax(a_fft) # Finds the index of the max
f_val = freq_x_axis[f_loc] # The strongest frequency value

samplenums = round(2*(1/f_val)*fs)

ax[2].plot(t[0:samplenums],a[0:samplenums])
ax[2].set_title("Two periods")
ax[2].set_xlabel("Time [s]")
ax[2].set_ylabel("Voltage [V]")
#plt.grid()
#plt.minorticks_on()

plt.show()

print(f"The largest frequency component is f = {f_val}")
plt.savefig('fft.pdf', bbox_inches='tight')
```

```
Calculated samplerate fs = 4422 kS/s
```