Fourier Transforms

ECE 320 Electronics I

Jake Glower - Lecture #16

Please visit Bison Academy for corresponding
lecture notes, homework sets, and solutions

Phasors

- Real numbers are all that's needed for DC analysis
- Complex numbers help with AC analysis

Voltages

$v(t) = a\cos(\omega t) + b\sin(\omega t)$ *time domain*

$V = a - jb$ *phasor (frequency) domain*

Impedances

$R \rightarrow R$

$L \rightarrow j\omega L$

$C \rightarrow \frac{1}{j\omega C}$

Voltage nodes, current loops, voltage division etc. work for both DC and AC

- You get complex numbers with AC however

Superposition

RLC circuits are linear

$$f(a+b) = f(a) + f(b)$$

If there are several inputs

- Analyze separately for each input .
- The total input is found by summing up each of the inputs.
- The total output is found by summing up each of the outputs.

Fourier Transforms convert an signal that is *not* a bunch of sine waves into a signal which *is* a bunch of sine waves

- Allows you to solve using phasor analysis and superposition

Fourier Transform

Assume a signal is periodic in time T:

$$x(t) = x(t+T)$$

then

$$x(t) = \sum_{n=0}^{\infty} a_n \cos(n\omega_0 t) + b_n \sin(n\omega_0 t) \qquad \omega_0 = \frac{2\pi}{T}$$

Translation:

If you add up a bunch of signals which are periodic in time T, the result is periodic in time T

If you have a periodic signal which is not a pure sine wave, it is made up of harmonics.

Computing Fourier Coefficients

All sine waves are orthogonal

$$mean(\sin(at) \cdot \cos(bt)) = 0$$

$$mean(\sin(at) \cdot \sin(bt)) = \begin{cases} 0 & a \neq b \\ \frac{1}{2} & a = b \end{cases}$$

$$mean(\cos(at) \cdot \cos(bt)) = \begin{cases} 0 & a \neq b \\ \frac{1}{2} & a = b \end{cases}$$

meaning:

$a_0 = mean(x(t))$ *a.k.a. the DC value of x(t)*

$a_n = 2 \cdot mean(x(t) \cdot \cos(n\omega_0 t))$ *cosine() terms*

$b_n = 2 \cdot mean(x(t) \cdot \sin(n\omega_0 t))$ *sine() terms*

Example 1: Known answer

$$x(t) = 1 + 3\cos(t) + 4\sin(2t)$$

In Matlab

```
t = [1:10000]' / 10000 * 2 * pi;
x = 1 + 3*cos(t) + 4*sin(2*t);
a0 = mean(x)
a0 =    1.0000

a1 = 2*mean(x .* cos(t))
a1 =    3.0000

b1 = 2*mean(x .* sin(t))
b1 =  2.9165e-015

a2 = 2*mean(x .* cos(2*t))
a2 = -3.0340e-015

b2 = 2*mean(x .* sin(2*t))
b2 =    4.0000
```

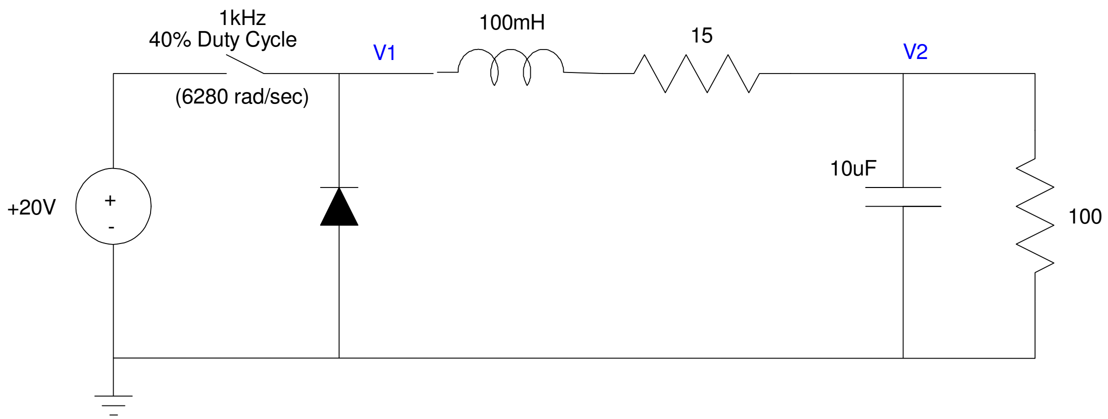
Complex Fourier Transform

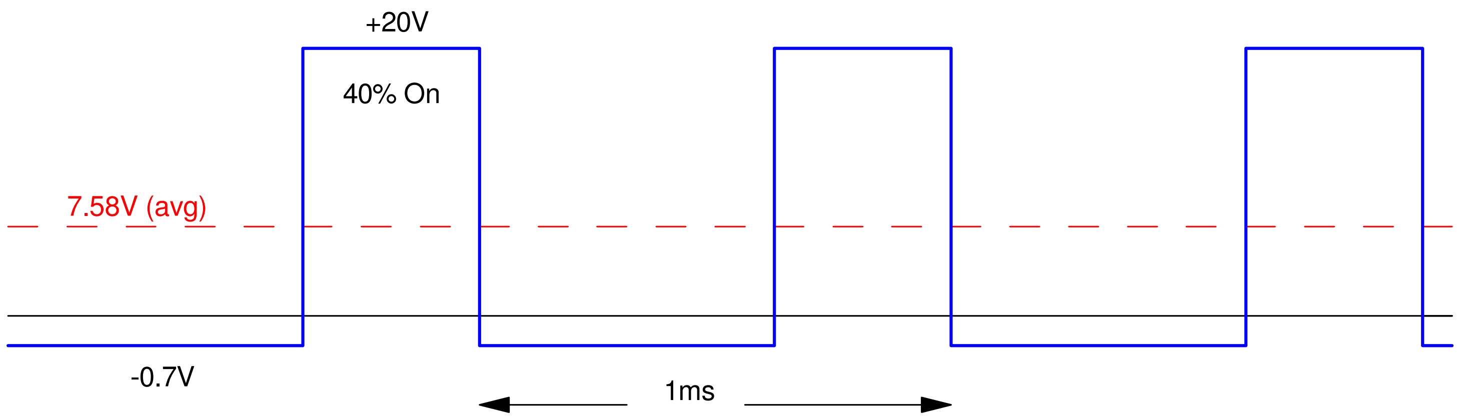
Easier if you don't mind complex numbers

$$X_n = a_n - jb_n = 2\ mean(x(t) \cdot e^{-jn\omega_0 t})$$

```
X1 = 2*mean( x .* exp(-j*t) )
X1 =   3.0000 - 0.0000i

X2 = 2*mean( x .* exp(-j*2*t) )
X2 =  -0.0000 - 4.0000i

X3 = 2*mean( x .* exp(-j*3*t) )
X3 =  5.4526e-015 +5.7784e-016i
```

Circuit Analysis with Fourier Transforms: Buck Converter





Previous Solution

Approximate V1(t) as

- A DC term (7.58V), and
- An AC term (20.7Vpp @ 1kHz)

The answers we got for the voltage at V2 were close to what CircutLab computed, but a little off.

Using Fourier Transforms, you can get more accurate results.

Step 1: Find the Fourier Series expansion for V1(t).

- Change in variable so the period is 2π

```
t = [1:10000]' / 10000;
x = 20*( t < 0.4)  - 0.7*( t >= 0.4);
t = t * 2 * pi;

X0 = mean(x)
X0 =  7.5779

X1 = 2*mean( x .* exp(-j*t) )
X1 =   3.8725 -11.9184i

X2 = 2*mean( x .* exp(-j*2*t) )
X2 =  -3.1360 - 2.2784i

X3 = 2*mean( x .* exp(-j*3*t) )
X3 =   2.0861 - 1.5157i

X4 = 2*mean( x .* exp(-j*4*t) )
X4 =  -0.9686 - 2.9811i
```

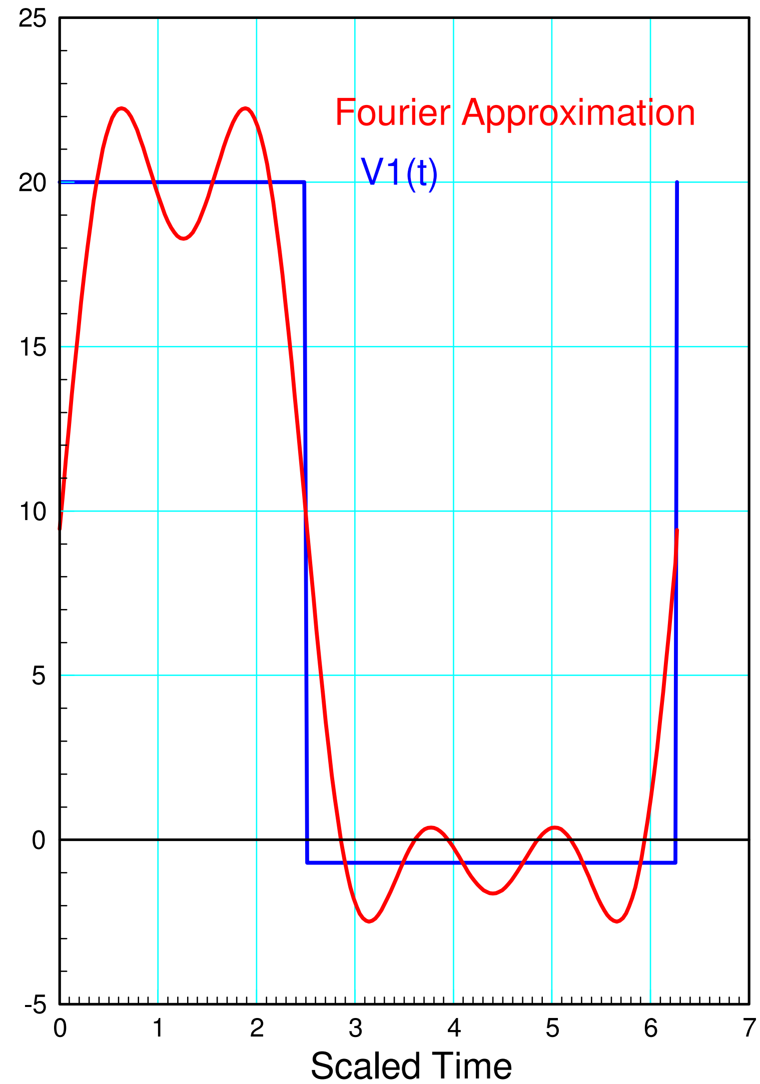
What this means is that

```
V1     = 7.5779
       + 2.8725 cos(t) + 11.9184 sin(t)
       - 3.1360 cos(2*t) + 2.2784 sin(2*t)
       + 2.0861 cos(3*t) + 1.515 sin(3*t)
       - 0.9686 cos(4*t) + 2.9811 sin(4*t)
```

- Note: Adding more terms improves the approximation.

Now you can use superposition to solve for V2(t)

- Treat this as 5 separate problems: each at a different frequency
- Solve for V2(t) at each frequency
- Add up the answers to get the total answer.



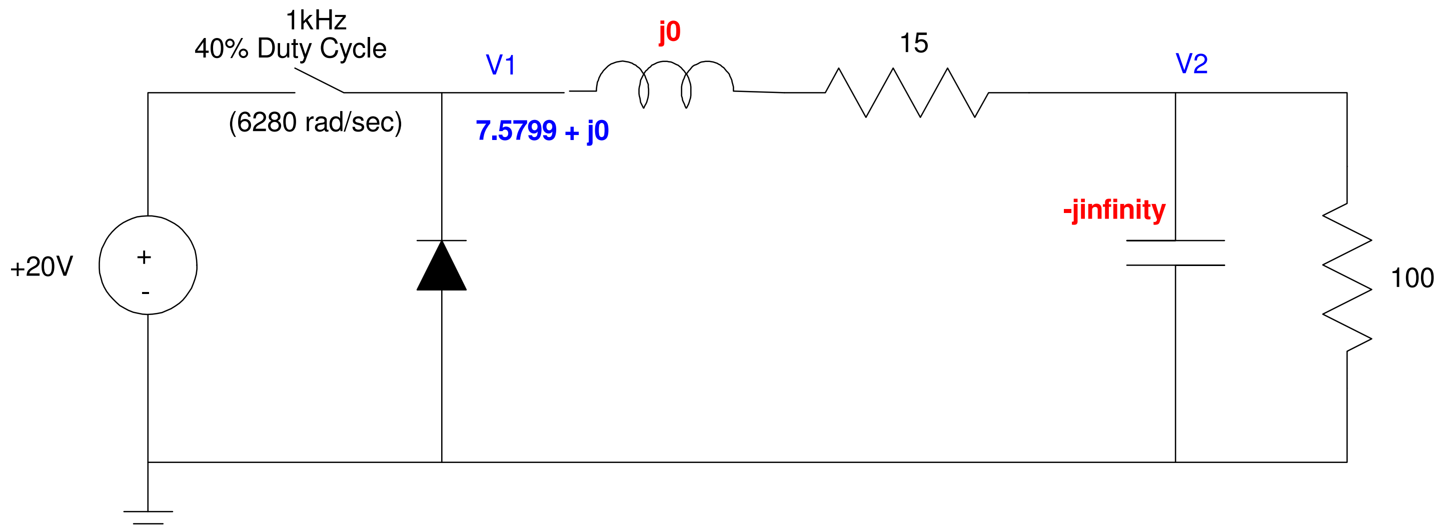
DC Analysis: $V_1(t) = 7.5779$

Redraw the circuit at w = 0 and solve for V2

$$V_1 = 7.5779 + j0$$

$$V_2 = \left(\frac{100}{100+15}\right) 7.5779$$

$$V_2 = 6.5895$$

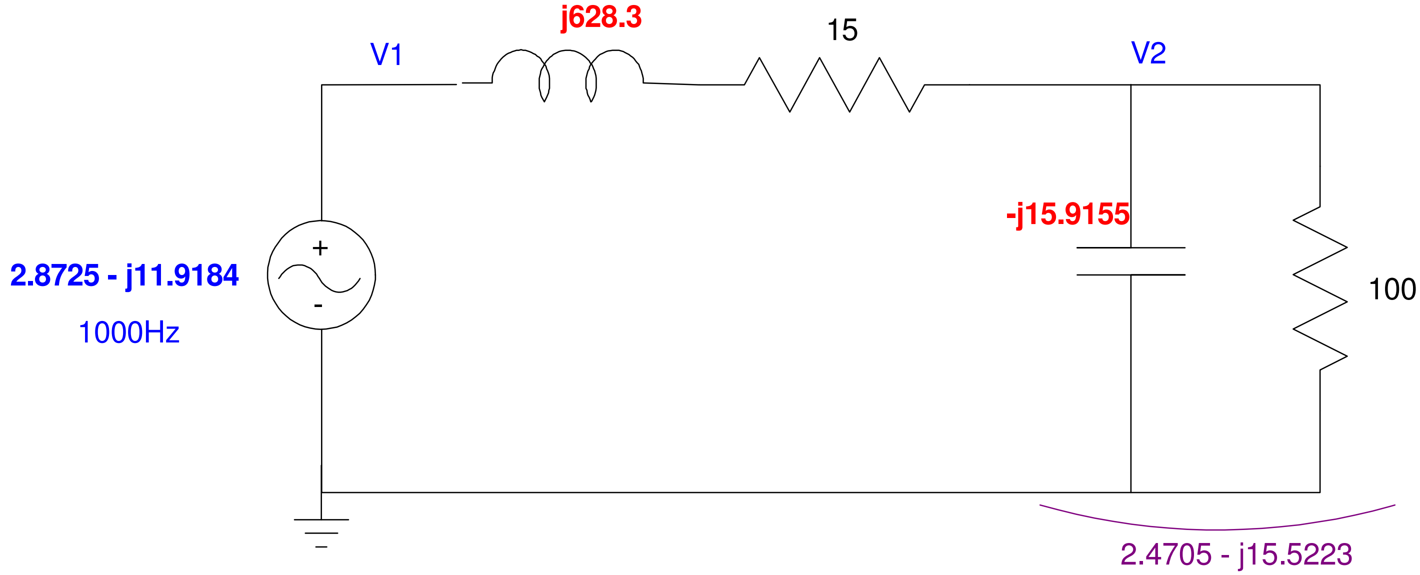


1st Harmonic: 1000Hz

$$V_1 = 2.8725\cos(\omega_0 t) + 11.9184\sin(\omega_0 t) \rightarrow 2.8725 - j11.9184$$

$$V_2 = \left(\frac{(2.4705 - j15.5223)}{(2.4705 - j15.5223) + (15 + j628)}\right)(2.8725 - j11.9184)$$

$$V_2 = -0.1290 + j0.2866$$

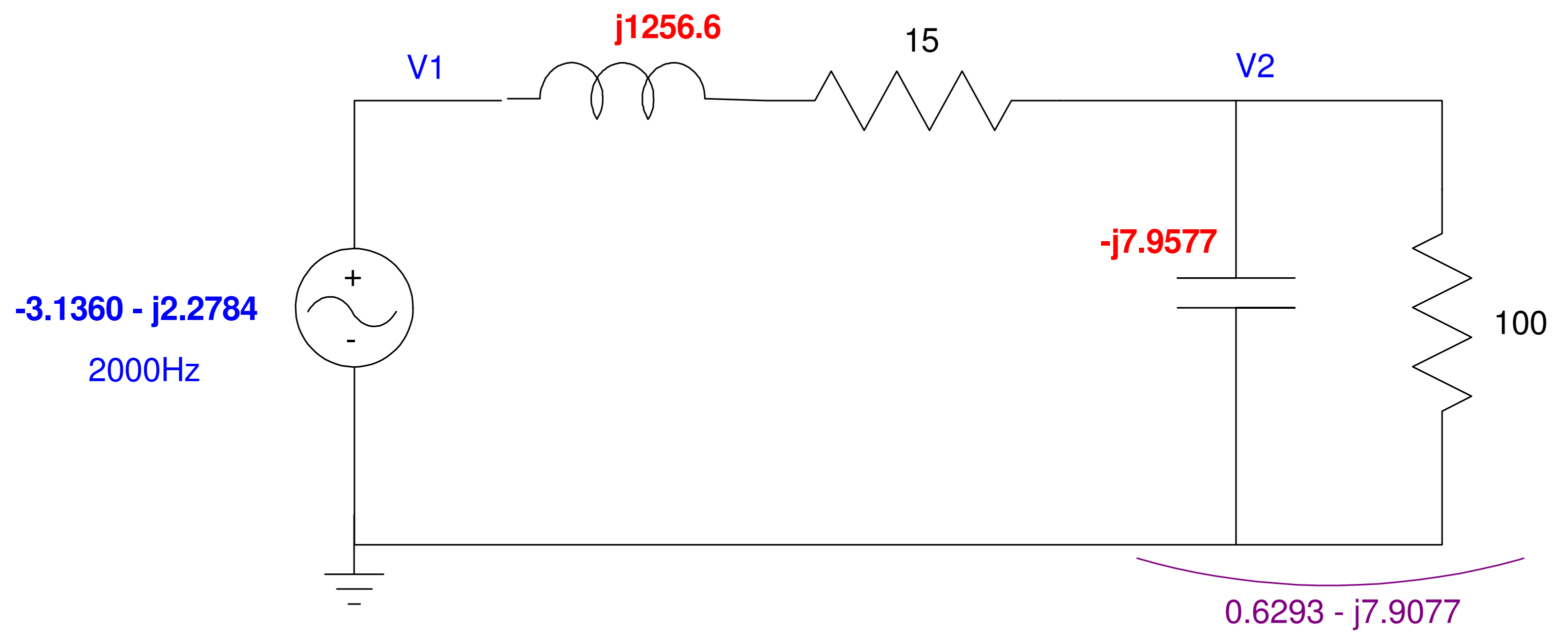


2nd Harmonic (2000Hz)

$$V_1 = -3.1360\cos(2\omega_0 t) + 2.2784\sin(2\omega_0 t) \rightarrow -3.1360 - j2.2784$$

$$V_2 = \left(\frac{(0.6293 - j7.9077)}{(0.6293 - j7.9077) + (15 + j1256.6)}\right)(-3.1360 - j2.2784)$$

$$V_2 = 0.0185 + j0.0162$$

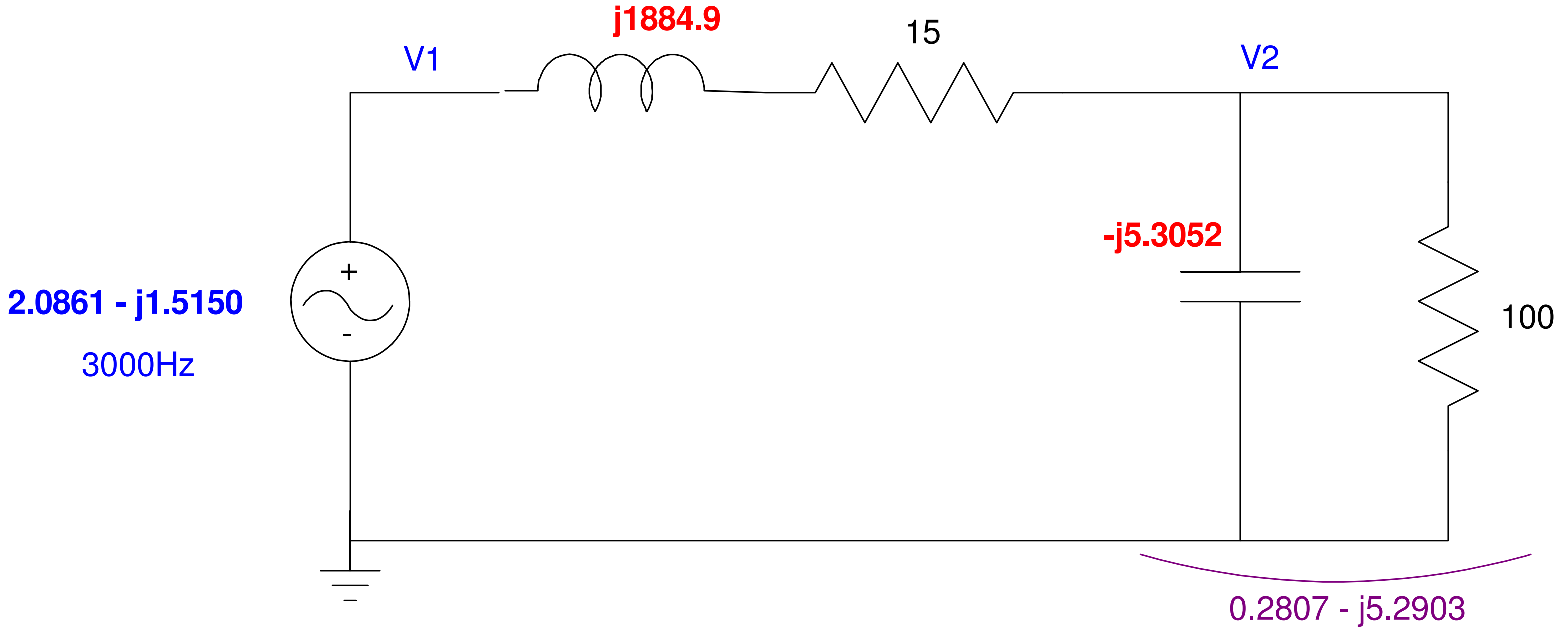


3rd Harmonic: 3000Hz

$$V_1 = 2.0861\cos(3\omega_0 t) + 1.5150\sin(3\omega_0 t) \rightarrow 2.0861 - j1.5150$$

$$V_2 = \left(\frac{(0.2807 - j5.2903)}{(0.2807 - j5.2903) + (15 + j1884.9)}\right)(2.0861 - j1.5150)$$

$$V_2 = -0.00061 + j0.00039$$

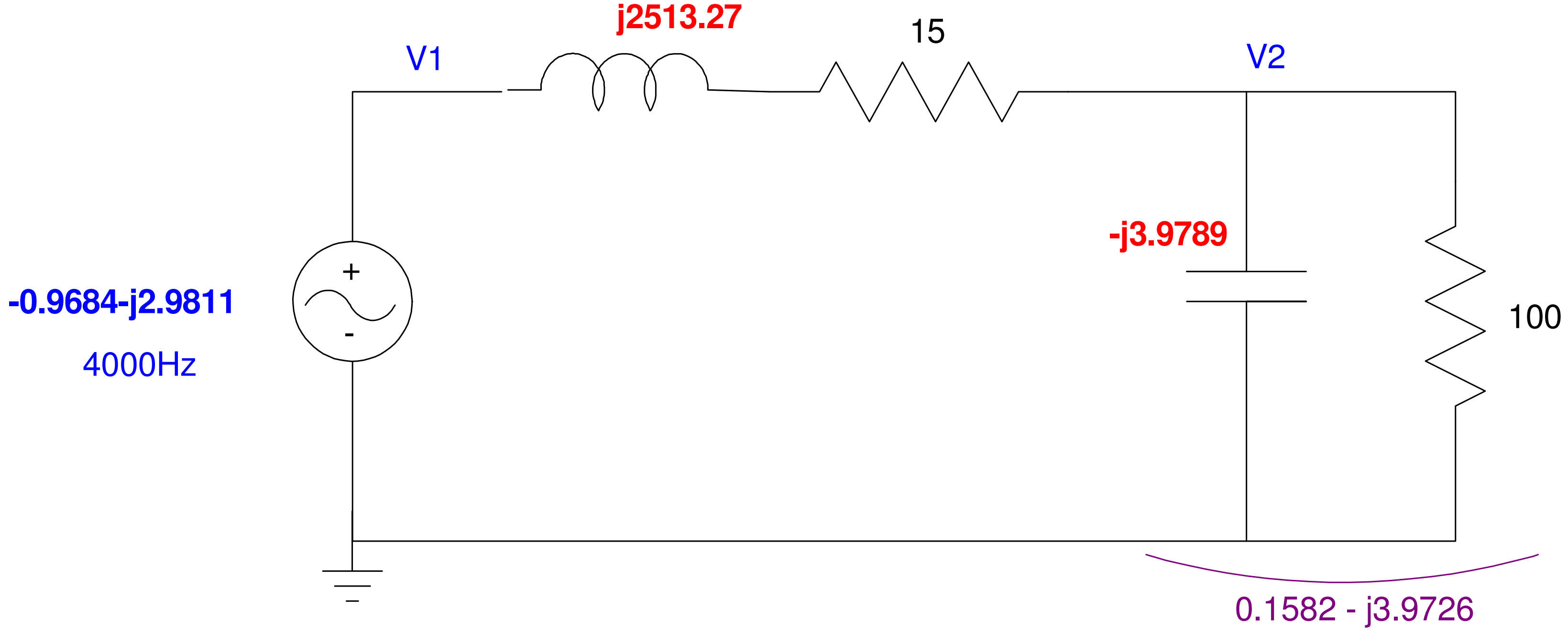


4th Harmonics: 4000 Hz

$$V_1 = -0.9684\cos(4\omega_0 t) + 2.9811\sin(4\omega_0 t) \rightarrow -0.9684 - j2.9811$$

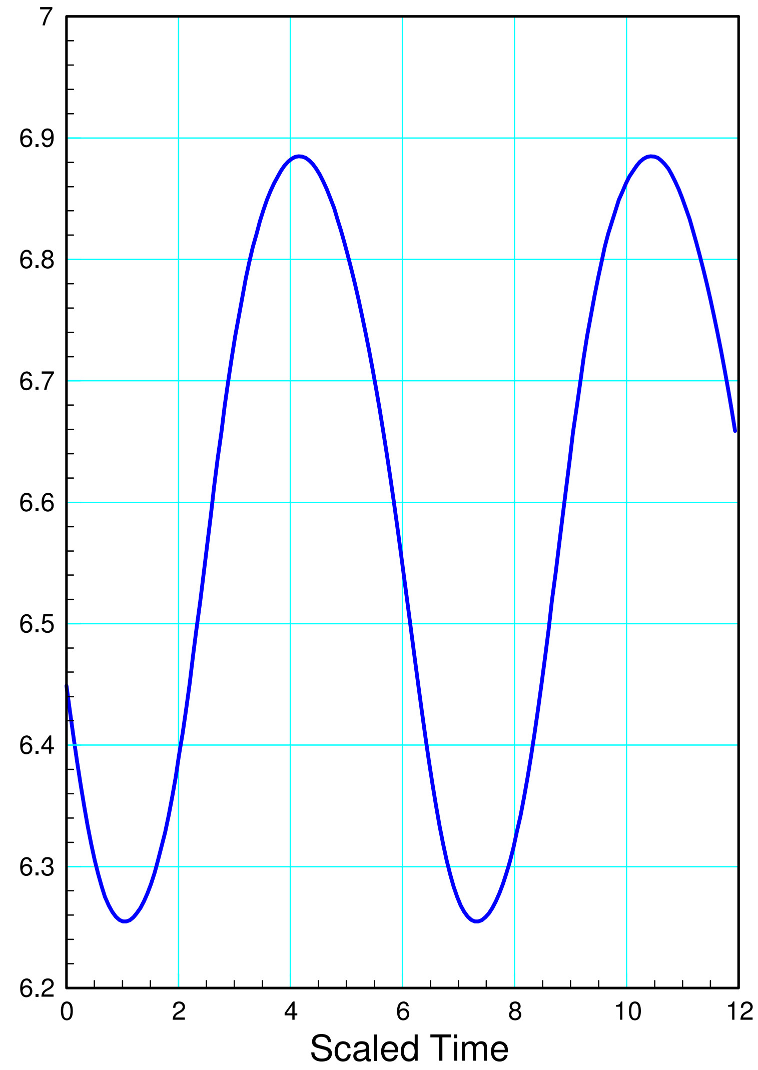
$$V_2 = \left(\frac{(0.1582 - j3.9726)}{(0.1582 - j3.9726) + (15 + j2513.27)}\right)(-0.9684 - j2.9811)$$

$$V_2 = 0.00132 + j0.00479$$



Net Result:

$$V_2 = 6.5895 +$$
$$-0.1290\cos(\omega_0 t) - 0.2866\sin(\omega_0 t)$$
$$+0.0185\cos(2\omega_0 t) - 0.0162\sin(2\omega_0 t)$$
$$-0.00061\cos(3\omega_0 t) - 0.00039\sin(3\omega_0 t)$$
$$+0.00132\cos(4\omega_0 t) - 0.00479\sin(4\omega_0 t)$$
$$+\cdots$$



Note

- In theory, you have to go out to infinity
- In practice, the harmonics go to zero very quickly
- Truncating the series after the 4th harmonic is very close

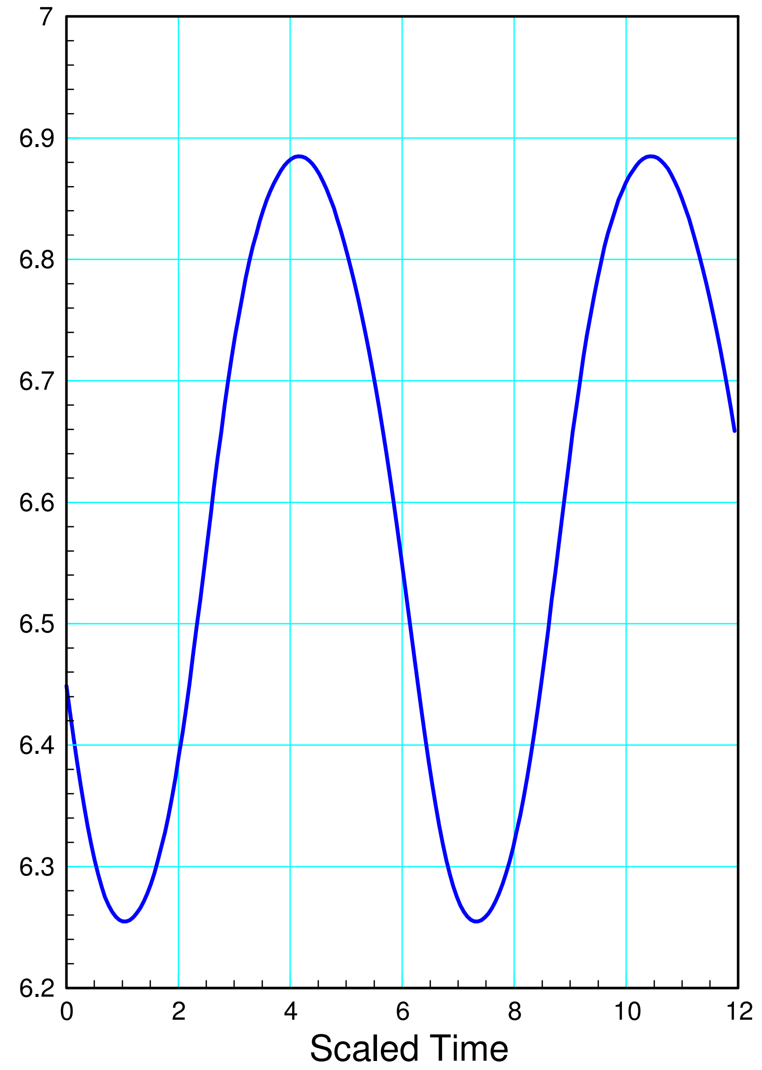
In Matlab:

- Output = Gain * Input
- note: slightly different answers from before due to rounding

```
>> Y0 = 100/(100+15)*X0
     6.5895

>> Y1 = (2.4705-j*15.52)/((2.4705-j*15.52)+(15+j*628.2))*X1
  -0.1542 + 0.2819i

>> Y2 = (0.6293-j*7.9077)/((0.6293-j*7.9077)+(15+j*1256.6))*X2
   0.0185 + 0.0162i

>> Y3 = (0.2807-j*5.2903)/((0.2807-j*5.2903)+(15+j*1884.9))*X3
  -0.0061 + 0.0039i

>> Y4 = (0.1582-j*3.9726)/((0.1582-j*3.9726)+(15+j*2513.27))*X4
   0.0013 + 0.0048i
```

Plotting V2(t):

```
V2 = Y0 + real(Y1)*cos(t) - imag(Y1)*sin(t);
V2 = V2 + real(Y2)*cos(2*t) - imag(Y2)*sin(2*t);
V2 = V2 + real(Y3)*cos(3*t) - imag(Y3)*sin(3*t);
V2 = V2 + real(Y4)*cos(4*t) - imag(Y4)*sin(4*t);
plot(t,V2)
>>
```



This matches CircuitLab results

- In theory, you need to include an infinite number of terms
- In practice, the terms quickly go to zero
- Fourier Transforms allow you to compute the explicit form for V2(t),
- Fourier Transforms are more accurate than what we did last lecture, and
- They are a ***lot*** more work.

Sidelight:

A good approximation for V2(t) is to include

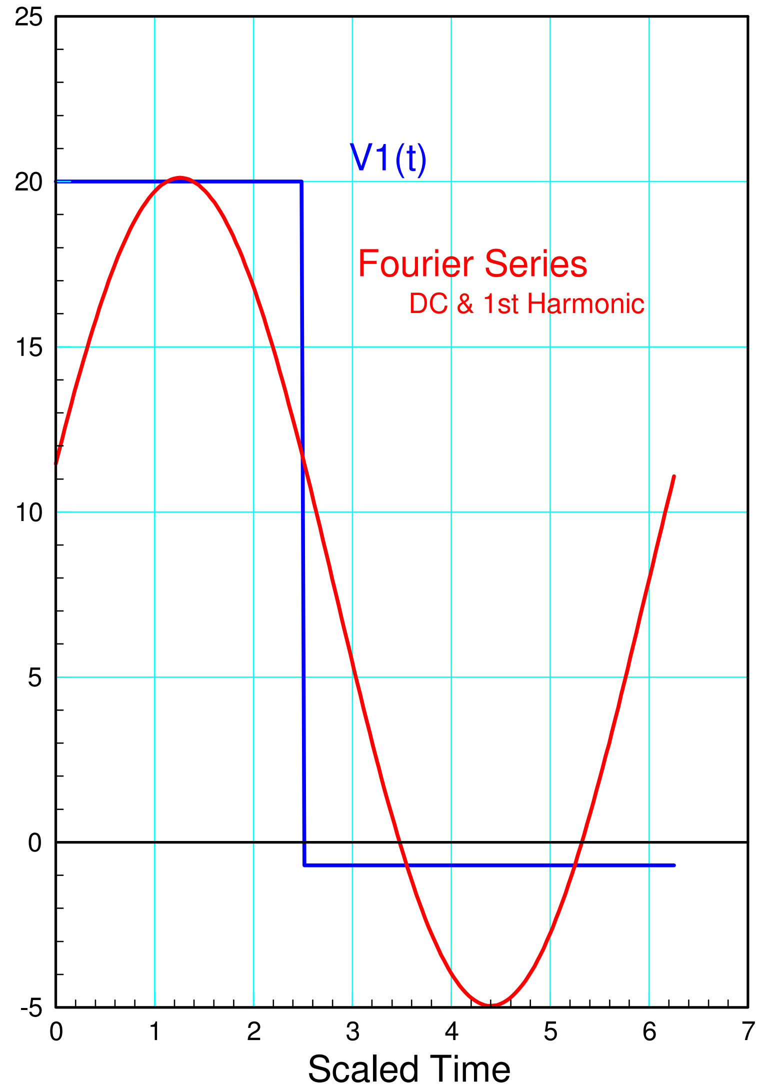
- The DC term and
- The 1st harmonic

Hence, approximate V1(t) including

- The DC term and
- The 1st Harmonic

A more accurate approximation of V1 uses

- The DC term, and
- The 1st term of the Fourier series for V1
- (rather than V1pp)



DC Analysis in Matlab:

```
V1dc = mean(x)
    7.5779

V2dc = 100/(100+15) * V1dc
    6.5895

note: mean(V2) = 6.5895
```

AC Analysis in Matlab

```
X1 = 2*mean( x .* exp(-j*t) )
    3.8725 -11.9184i

V1pp = 2*abs(X1)
    25.0635       vs. 20.8Vpp

V2pp = (2.4705-j*15.52)/((2.4705-j*15.52)+(15+j*628.2))*V1pp;
V2pp = abs(V2pp)
    0.6426

note: max(V2)-min(V2) = 0.6303
```