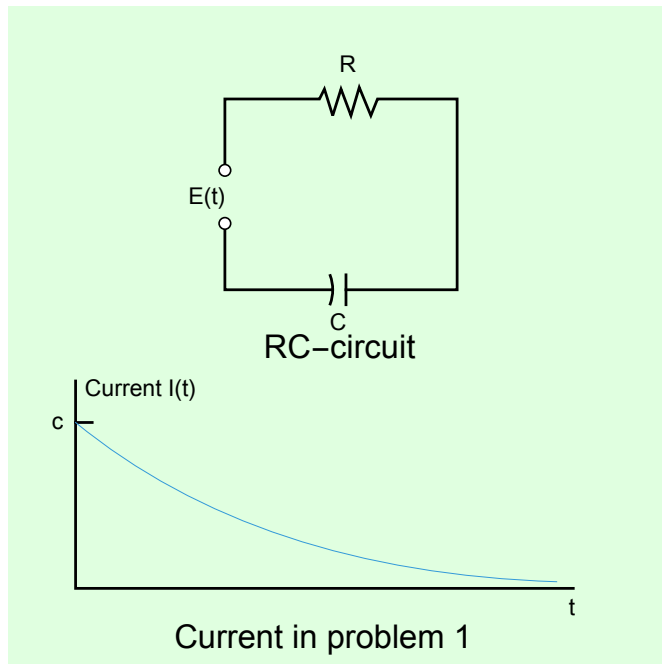


1 - 6 RLC-Circuits: special cases

1. RC-Circuit. Model the RC-Circuit in the figure below. Find the current due to a constant E.



```
ClearAll["Global`*"]
```

The problem is asking for a look at RC circuit, not RLC.

The site <https://www.intmath.com/differential-equations/6-rc-circuits.php> assumes a constant voltage source, just what the problem specifies. Below: There is no inductance here, only R and C.

```
eqnw = rR (D[eye[t], t]) + eye[t] / cC == 0
```

$$\frac{\text{eye}[t]}{cC} + rR \text{eye}'[t] == 0$$

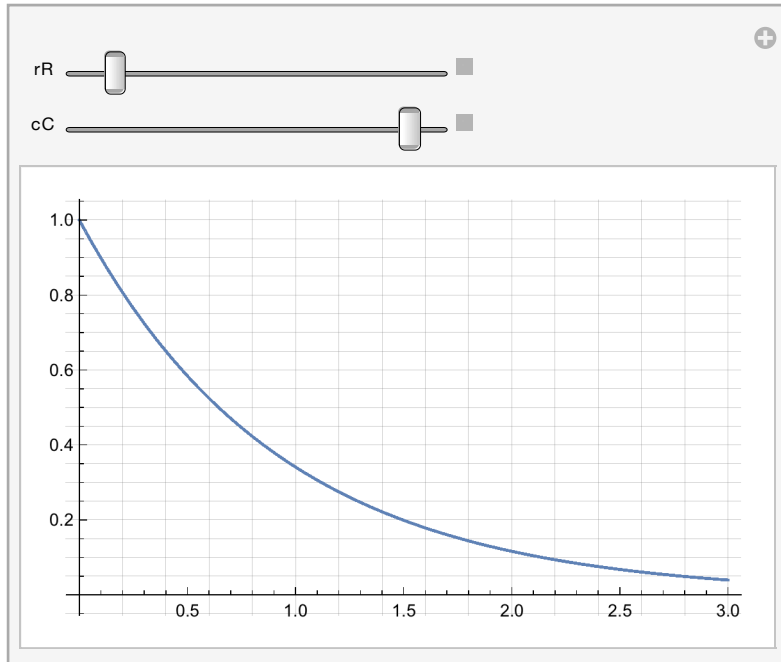
Within a certain range of capacitance and resistance, the plot resembles the one in the problem description, and can be manipulated to imitate changing parameters, with the voltage remaining constant.

```
sol2 = DSolve[eqnw, eye, t]
```

```
{ {eye -> Function[{t}, e^(-t/(cC rR)) C[1]] } }
```

It looks like the current is normalized to 1 at $t=0$, and the fraction of its max value at a given time needs to be estimated from the underlying grid.

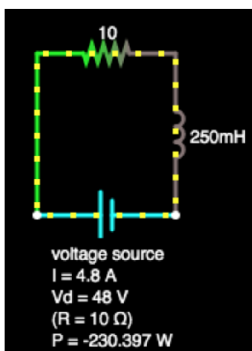
```
Manipulate[Plot[ $e^{-\frac{t}{cC rR}}$ , {t, 0, 3}, PlotRange → All, GridLines → All], {rR, 0.2, 10}, {cC, 0.01, 1}]
```



A random scrap from a different perspective, kept as interesting junk.

```
{ind, cap, res} = {l i'[t] == v_l[t], v_c'[t] == 1/c i[t], r i[t] == v_r[t]};  
kirchhoff = v_l[t] + v_c[t] + v_r[t] == v_s[t];
```

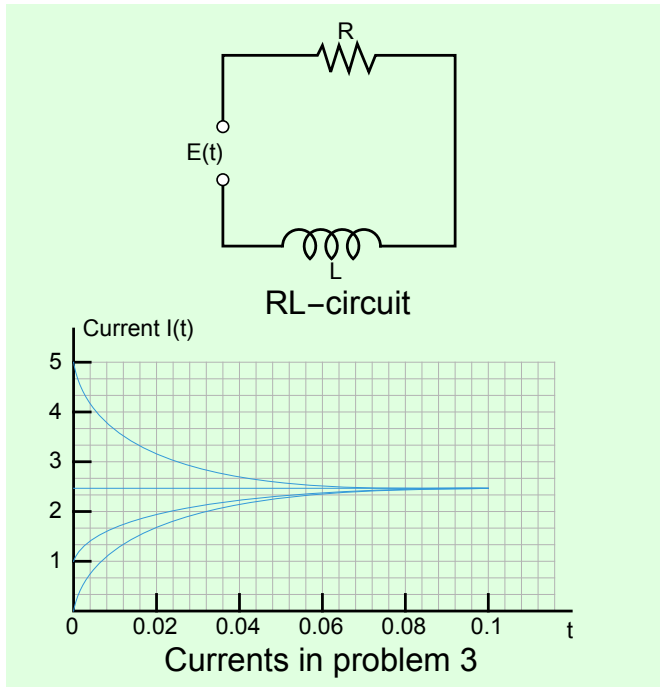
3. RL-Circuit. Model the RL-circuit in the figure below. Find a general solution when R , L , E are any constants. Graph or sketch solutions when $L = 0.25$ H, $R = 10 \Omega$, and $E = 48$ V.



The above screenshot came from the online app at <https://falstad.com/circuit/>. The current it shows agrees with the old formula for current, $I=E/R$, and was captured after the resistance had plenty of time to decay. And that's all it is, except that there is a time constant to apply. The time constant becomes ever smaller as the operation time increases. Since the

problem description talks in terms of a constant state, it seems the time constant would become vanishingly small, leaving merely $I=E/R=4.8$ amps.

In[8]:= **ClearAll["Global`*"]**



When there are a lot of variables to watch, the Manipulate command is the only way I know to get an overview. The box below is based on the material at <https://www.electronics-tutorials.ws/inductor/lr-circuits.html> and may not agree with the text in detail.

In[9]:= **eye[vee_, are_, ell_, tee_] = $\frac{\text{vee}}{\text{are}} \left(1 - e^{-\frac{\text{are tee}}{\text{ell}}} \right)$**

Out[9]=
$$\frac{\left(1 - e^{-\frac{\text{are tee}}{\text{ell}}} \right) \text{vee}}{\text{are}}$$

It takes some time for the current to reach its max value. From $t=0.4$ on in the green grid below, the circuit current is nominal.

In[10]:= **Grid[Table[{tee, eye[48, 10, 0.25, tee]}, {tee, 0, 0.6, 0.1}], Frame → All]**

Out[10]=

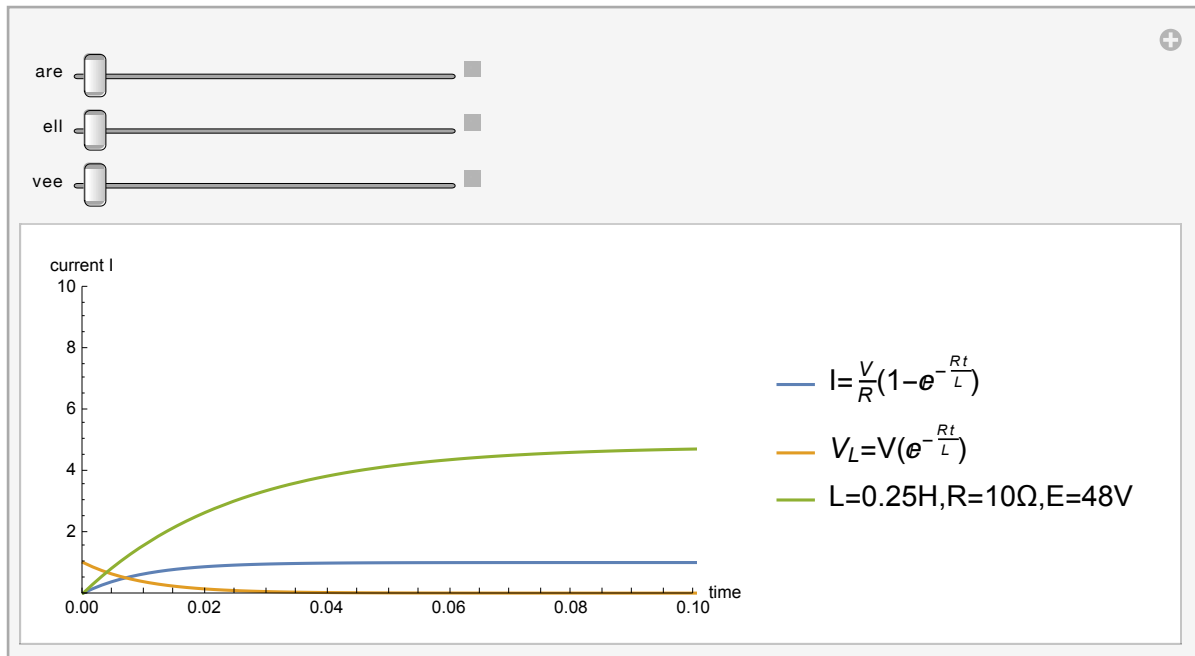
0.	0.
0.1	4.71208
0.2	4.79839
0.3	4.79997
0.4	4.8
0.5	4.8
0.6	4.8

```
In[11]:= veel[vee_, are_, ell_, tee_] = vee  $\left( e^{-\frac{\text{are tee}}{\text{ell}}} \right)$ 
```

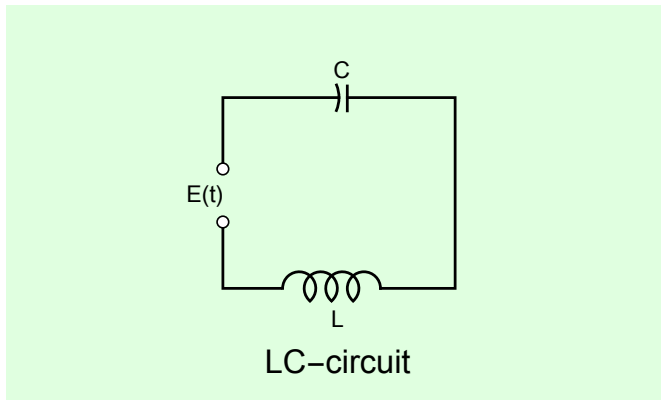
```
Out[11]=  $e^{-\frac{\text{are tee}}{\text{ell}}} \text{vee}$ 
```

```
In[12]:= Manipulate[
  Plot[{Abs[eye[vee, are, ell, tee]], Abs[veel[vee, are, ell, tee]],
    Abs[eye[48, 10, 0.25, tee]]}, {tee, 0, 5},
  PlotLegends → {" $I = \frac{V}{R}(1 - e^{-\frac{Rt}{L}})$ ", " $V_L = V(e^{-\frac{Rt}{L}})$ ", "L=0.25H,R=10Ω,E=48V"},
  PlotRange → {{0, 0.1}, {0, 10}}, AxesLabel → {"time", "current I"},
  AspectRatio → 0.5], {are, 1, 200}, {ell, 0.01, 10}, {vee, 1, 50}]
```

```
Out[12]=
```



5. LC-Circuit. This is an RLC-circuit with negligibly small R (analog of an undamped mass-spring system). Find the current when $L=0.5$ H, $C = 0.005$ F, and $E = \sin[t]$ V, assuming zero initial current and charge.



I ran across a couple of snippets, including one from the *Mathematica* documentation, suggesting that state space modeling would be a good way to look at circuits in Mathematica. I use it here.

```
ClearAll["Global`*"]
```

```
eqns = {eL q''[t] + aR q'[t] + 1/cC q[t] == Vee[t]};
```

```
m1 = StateSpaceModel[eqns,
  {{q[t], 0}, {q'[t], 0}}, {{Vee[t], 0}}, {q'[t]}, t]
```

$$\left(\begin{array}{cc|c} 0 & 1 & 0 \\ -\frac{1}{cC eL} & -\frac{aR}{eL} & \frac{1}{eL} \\ \hline 0 & 1 & 0 \end{array} \right) S$$

Here I put in the given parameters, taking the opportunity to equate the resistance with zero.

```
ms = m1 /. {cC -> 0.005, eL -> 0.5, aR -> 0}
```

$$\left(\begin{array}{cc|c} 0 & 1 & 0 \\ -400. & 0. & 2. \\ \hline 0 & 1 & 0 \end{array} \right) S$$

The way to get output from a state space model is to use the command **OutputResponse**. Since the voltage depends on a periodic function, I drop the V for the input field, the voltage, because it is just a label.

```

outz = OutputResponse[{ms}, Sin[t], t]
{ (1.46082 × 10-17 + 0.0526316 i)
  ( (0. + 0.0952381 i) Cos[20. t] - (0. + 1. i) Cos[19. t] Cos[20. t] +
    (0. + 0.904762 i) Cos[20. t] Cos[21. t] -
    (1.66533 × 10-16 - 7.21645 × 10-17 i) Cos[20. t] Sin[19. t] -
    (2.24688 × 10-17 + 6.60847 × 10-19 i) Sin[20. t] +
    (2.35922 × 10-16 + 6.93889 × 10-18 i) Cos[19. t] Sin[20. t] -
    (2.13454 × 10-16 + 6.27805 × 10-18 i) Cos[21. t] Sin[20. t] +
    (5.96745 × 10-17 - 1. i) Sin[19. t] Sin[20. t] +
    (1.50673 × 10-16 - 6.52917 × 10-17 i) Cos[20. t] Sin[21. t] -
    (5.39912 × 10-17 - 0.904762 i) Sin[20. t] Sin[21. t]) }

```

It is necessary to clean up the result with a small **Chop**.

```

outt = Chop[ComplexExpand[Re[outz]], 10-16] // FullSimplify
{0.00501253 Cos[1. t] - 0.00501253 Cos[20. t] + 3.46945 × 10-18 Cos[39. t]}

```

Recognizing the periodic value of cosine, I can get the expression ready for a second chop by doing

```

outtff = outt /. Cos[39. t] → 1
{3.46945 × 10-18 + 0.00501253 Cos[1. t] - 0.00501253 Cos[20. t]}

```

And then the **Chop**.

```

outtfff = Chop[%, 10-17]
{0.00501253 Cos[1. t] - 0.00501253 Cos[20. t]}

```

Testing the identity of those coefficients

```

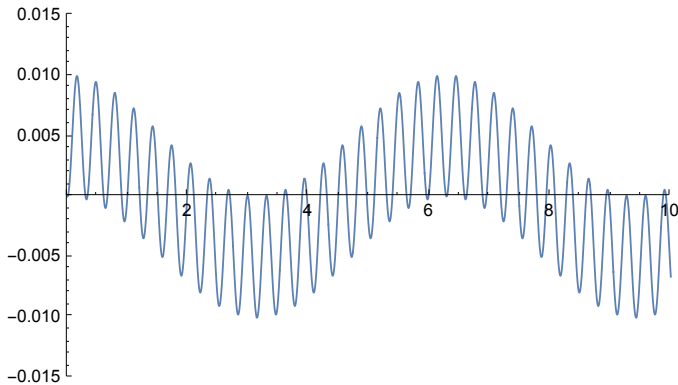
1/0.005012531328320802`
199.5

```

I find that the answer matches the text answer, justifying the green coloration above.

The plot is interesting.

```
Plot[outtff, {t, 0, 10}, ImageSize → 350, AspectRatio → 0.6,
  PlotRange → {{-0.01, 10}, {-0.015, 0.015}}, PlotStyle → Thickness[0.003]]
```



7 - 18 General RLC-circuits

7. Tuning. In tuning a stereo system to a radio station, we adjust the tuning control (turn a knob) that changes C (or perhaps L) in an RLC-circuit so that the amplitude of the steady-state current, numbered line (5), p. 95 becomes maximum. For what C will this happen?

It is where the particular solution of the homogeneous equation is maximized. Numbered line (5) looks like

$$I_p(t) = I_0 \sin[\omega t - \theta]$$

The quantity θ is known as the phase lag, and, I suppose, the signal is best, I_p maximized, when θ equals zero.

8 - 14 Find the steady-state current in the RLC-circuit in the figure below for the given data.

9. $R = 4 \Omega$, $L = 0.1 \text{ H}$, $C = 0.05 \text{ F}$, $E = 110 \text{ V}$

$$L D[q[t], \{t, 2\}] + R D[q[t], t] - \frac{1}{C} q[t] = v[t]$$

$$\text{eqn} = 0.1 q''[t] + 4 q'[t] - \frac{1}{0.05} q[t] == 110$$

$$-20. q[t] + 4 q'[t] + 0.1 q''[t] == 110$$

```
sol = DSolve[eqn, q, t]
```

$$\left\{ \left\{ q \rightarrow \text{Function}\left[\{t\}, -5.5 + e^{-44.4949 t} C[1] + e^{4.4949 t} C[2] \right] \right\} \right\}$$

If $C[1]=C[2]=0$, then the green cell above matches the text answer.

$$11. R = 12 \, \Omega, L = 0.4 \, \text{H}, C = \frac{1}{80} \text{F}, E = 220 \sin[10 t] \, \text{V}$$

The state space method has been working where former methods I tried did not, so it makes sense to stick with it.

```
ClearAll["Global`*"]
```

$$\text{eqns} = \left\{ eL q''[t] + aR q'[t] + \frac{1}{cC} q[t] == \text{Vee}[t] \right\};$$

```
m1 = StateSpaceModel[eqns,
  {{q[t], 0}, {q'[t], 0}}, {{Vee[t], 0}}, {q'[t]}, t]
```

$$\left(\begin{array}{cc|c} 0 & 1 & 0 \\ -\frac{1}{cC} & -\frac{aR}{eL} & \frac{1}{eL} \\ 0 & 1 & 0 \end{array} \right) S$$

Here I put in the given parameters.

$$\text{ms} = \text{m1} /. \left\{ cC \rightarrow \frac{1}{80}, eL \rightarrow 0.4, aR \rightarrow 12 \right\}$$

$$\left(\begin{array}{cc|c} 0 & 1 & 0 \\ -200. & -30. & 2.5 \\ 0 & 1 & 0 \end{array} \right) S$$

The way to get output from a state space model is to use the command **OutputResponse**.

```
outz = OutputResponse[{ms}, 220 Sin[10 t], t]
```

$$\left\{ 0. + e^{-30. t} \left(22. e^{10. t} - 27.5 e^{20. t} - 7.10543 \times 10^{-15} e^{20. t} \cos[10. t] + 5.5 e^{30. t} \cos[10. t] + 7.10543 \times 10^{-15} e^{20. t} \sin[10. t] + 16.5 e^{30. t} \sin[10. t] + 7.10543 \times 10^{-15} e^{40. t} \sin[10. t] \right) \right\}$$

It is necessary to clean up the result with a **Chop**.

```
outt = Chop[outz, 10-14] // FullSimplify
```

$$\left\{ 22. e^{-20. t} - 27.5 e^{-10. t} + 5.5 \cos[10. t] + 16.5 \sin[10. t] \right\}$$

I guess the e factors can be dropped if they are small enough, say, at 3 seconds.

$$\text{N}[-27.500000000000007 e^{-10. t}] /. t \rightarrow 3$$

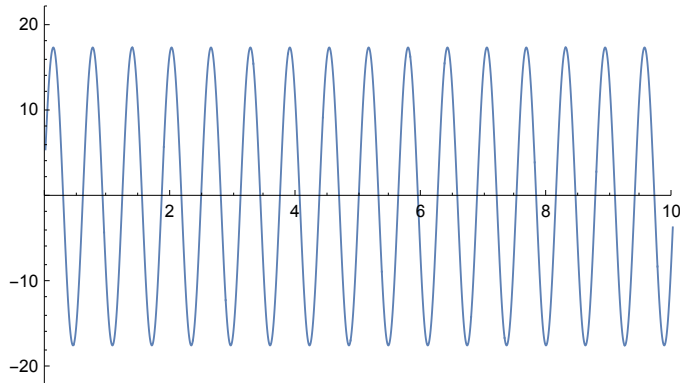
$$-2.57335 \times 10^{-12}$$

Evidently the text considers that size to be negligible, leaving

$$5.5 \cos[10. t] + 16.5 \sin[10. t]$$

as the answer. The plot looks routine.


```
Plot[5.5 Cos[10. t] + 16.5 Sin[10. t],
 {t, 0, 10}, ImageSize → 350, AspectRatio → 0.6,
 PlotRange → {{-0.01, 10}, {-22, 22}}, PlotStyle → Thickness[0.003]]
```



13. $R = 12$, $L = 1.2$ H, $C = \frac{20}{3} \times 10^{-3}$ F, $E = 12,000 \sin[25 t]$ V

$$C = \frac{20}{3} * \frac{1}{1000} = \frac{20}{3000} = \frac{2}{300}$$

```
ClearAll["Global`*"]
```

```
eqns = {eL q''[t] + aR q'[t] + 1/cC q[t] == Vee[t]};
```

```
m1 = StateSpaceModel[eqns,
 {{q[t], 0}, {q'[t], 0}}, {{Vee[t], 0}}, {q'[t]}, t]
```

$$\left(\begin{array}{cc|c} 0 & 1 & 0 \\ -\frac{1}{cC eL} & -\frac{aR}{eL} & \frac{1}{eL} \\ 0 & 1 & 0 \end{array} \right) S$$

Here I put in the given parameters.

```
ms = m1 /. {cC → 20/3 * 10^-3, eL → 1.2, aR → 12}
```

$$\left(\begin{array}{cc|c} 0 & 1 & 0 \\ -125. & -10. & 0.833333 \\ 0 & 1 & 0 \end{array} \right) S$$

The way to get output from a state space model is to use the command **OutputResponse**.

```

outz = OutputResponse[{ms}, 12 000 Sin[25 t], t]
{ (0. + 0. i) - (400. + 1.56319 × 10-14 i) e-5. t
  ((-1. + 0. i) Cos[10. t] + (1. + 0. i) e5. t Cos[10. t]2 Cos[25. t] +
   (0.75 - 4.80505 × 10-16 i) Sin[10. t] -
   (3.19744 × 10-16 - 3.21521 × 10-16 i) e5. t Cos[10. t] Cos[25. t]
   Sin[10. t] + (1. + 2.45581 × 10-16 i) e5. t Cos[25. t] Sin[10. t]2 -
   (0.5 - 7.49623 × 10-17 i) e5. t Cos[10. t]2 Sin[25. t] +
   (1.42109 × 10-16 - 9.97247 × 10-17 i) e5. t Cos[10. t] Sin[10. t]
   Sin[25. t] - (0.5 - 1.39035 × 10-16 i) e5. t Sin[10. t]2 Sin[25. t]) }

```

It is necessary to clean up the result with a **Chop**.

```

outt = Chop[ComplexExpand[Re[outz]], 10-15] // Simplify
{-300. e-5. t Sin[10. t] +
  Cos[10. t] (400. e-5. t + 1.27898 × 10-13 Cos[25. t] Sin[10. t]) -
  2.84217 × 10-14 Sin[20. t] Sin[25. t] +
  Cos[10. t]2 (-400. Cos[25. t] + 200. Sin[25. t]) +
  Sin[10. t]2 (-400. Cos[25. t] + 200. Sin[25. t]) }

```

There is a $\sin^2 + \cos^2$ trig identity in the above, but I'm going to have to pull it out by hand.

```

outhnd = -300. e-5. t Sin[10. t] +
  Cos[10. t] (400. e-5. t) + (-400. Cos[25. t] + 200. Sin[25. t])
400. e-5. t Cos[10. t] - 400. Cos[25. t] - 300. e-5. t Sin[10. t] + 200. Sin[25. t]

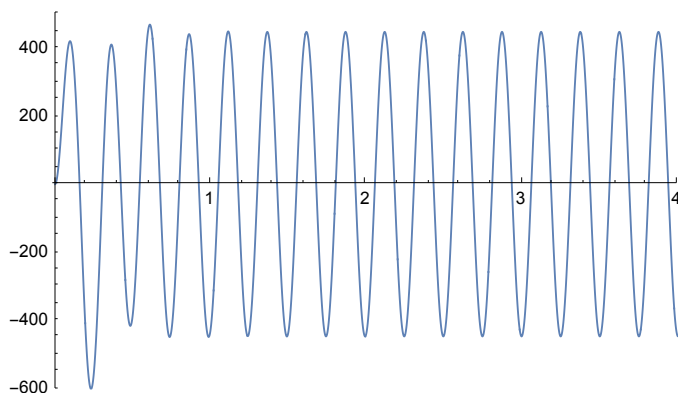
outhnd2 = Collect[outhnd, e-5. t]
Clear["Global`*"]

```

```
-400. Cos[25. t] + e-5. t (400. Cos[10. t] - 300. Sin[10. t]) + 200. Sin[25. t]
```

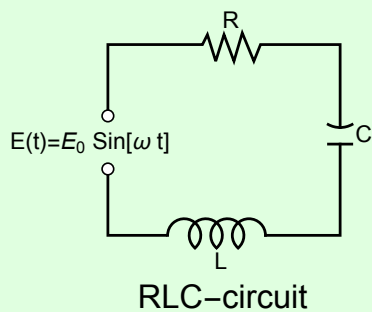
While I was pulling things out by hand, I pulled out a choppable term. The text constant B is equal to -300. The text constant A is equal to 1 in one position and 400 in another position. That makes my answer wrong, technically. I guess I should make it yellow, though I don't feel it is a just action to do so. I feel like it is correct.

```
Plot[-400. Cos[25. t] + e-5. t (400. Cos[10. t] - 300. Sin[10. t]) +  
200. Sin[25. t], {t, 0, 4}, ImageSize -> 350, AspectRatio -> 0.6,  
PlotRange -> {{-0.01, 4}, {-600, 500}}, PlotStyle -> Thickness[0.003]]
```



15. Cases of damping. What are the conditions for an RLC-circuit to be (I) overdamped, (II) critically damped, (III) underdamped? What is the critical resistance R_{crit} (the analog of the critical damping constant $2\sqrt{mk}$)

16 - 18 Solve the initial value problem for the RLC-circuit shown below, with the given data, assuming zero initial current and charge. Graph or sketch the solution.



17. $R = 6 \, \Omega$, $L = 1 \, \text{H}$, $C = 0.04 \, \text{F}$, $E = 600(\cos[t] + 4 \sin[t])\text{V}$

```
ClearAll["Global`*"]
```

```
eqns = {eL q''[t] + aR q'[t] + 1/cC q[t] == Vee[t]};
```

```
m1 = StateSpaceModel[eqns,
  {{q[t], 0}, {q'[t], 0}}, {{Vee[t], 0}}, {q'[t]}, t]

$$\left( \begin{array}{cc|c} 0 & 1 & 0 \\ -\frac{1}{cC eL} & -\frac{aR}{eL} & \frac{1}{eL} \\ 0 & 1 & 0 \end{array} \right) S$$

```

Here I put in the given parameters.

```
ms = m1 /. {cC → 0.04, eL → 1, aR → 6}
```

```

$$\left( \begin{array}{cc|c} 0 & 1 & 0 \\ -25. & -6 & 1 \\ 0 & 1 & 0 \end{array} \right) S$$

```

The way to get output from a state space model is to use the command **OutputResponse**.

```
outz = OutputResponse[{ms}, 600 (Cos[t] + 4 Sin[t]), t]
{ (0. + 0. i) +
  e-3. t ( (-100. - 1.11022 × 10-14 i) Cos[4. t] + (100. + 1.11022 × 10-14 i)
    e3. t Cos[t] Cos[4. t]2 - (1.87214 × 10-14 - 1.65445 × 10-14 i)
    e3. t Cos[4. t]2 Sin[t] + (75. + 1.52656 × 10-14 i) Sin[4. t] -
    (8.65974 × 10-15 + 1.80411 × 10-14 i) e3. t Cos[t] Cos[4. t] Sin[4. t] +
    (2.27374 × 10-13 + 2.91161 × 10-14 i) e3. t Cos[4. t] Sin[t] Sin[4. t] +
    (100. - 1.14492 × 10-14 i) e3. t Cos[t] Sin[4. t]2 -
    (0. + 7.91555 × 10-14 i) e3. t Sin[t] Sin[4. t]2) }
```

```
outt = Chop[ComplexExpand[Re[outz]]] // Simplify
```

```
{ -100. e-3. t Cos[4. t] + 100. Cos[t] Cos[4. t]2 +
  Sin[4. t] (75. e-3. t + 100. Cos[t] Sin[4. t]) }
```

```
outtf = Collect[outt, e-3. t]
```

```
{ 100. Cos[t] Cos[4. t]2 + 100. Cos[t] Sin[4. t]2 +
  e-3. t (-100. Cos[4. t] + 75. Sin[4. t]) }
```

I can see the $\sin^2 + \cos^2$ identity in the above, but will have to take it out by hand.

```
100. Cos[t] + e-3. t (-100. Cos[4. t] + 75. Sin[4. t])
```

And with that, the above cell matches the text answer.

```
Plot[100. Cos[t] + e-3. t (-100. Cos[4. t] + 75. Sin[4. t]),  
  {t, 0, 10}, ImageSize → 350, AspectRatio → 0.6,  
  PlotRange → {{-0.01, 10}, {-125, 125}}, PlotStyle → Thickness[0.003]]
```

