Project Check List

Each section should provide a fairly complete solution which could stand by itself (except in Part IV where we borrow the impulse response from Part III). However, it is a good idea to compare results in one section to those obtained in another to verify they are correct (well, at least they are consistent—if they aren’t you know something is wrong).

Part I: Classic Differential Equation

1. Derivation of differential equation.
2. Initial conditions (derivation shown).
3. Particular solution.
4. Characteristic equation and corresponding homogeneous solution.
5. Complete solution (homogeneous plus particular).
6. Evaluation of constants (as dictated by initial conditions).
7. Plot of final result.

Notes:

- Node voltage easiest approach to obtaining differential equation, but mesh current will work too.
- See the handouts on differential equations of August 25 and 27 for information about the steps necessary to solve the differential equation.
- See the Bode plot handout of October 25 for information about plotting a function using \textit{matlab}.

Part II: State Space

1. Derivation state and output equations.
2. Step and impulse response.
3. Transfer function.
4. Poles and zeros.

5. Bode plot.

Notes:

- Can use either *lsim()* or *step()* to obtain the step response. Use *impulse()* for the impulse response. For more information see the handout of October 4 on obtaining the transfer function.
- Use the command *tf()* to obtain the transfer function.
- The poles and zeros and be obtained either by factor the numerator and denominator of the transfer function (using, for example, *roots()* or by using *pzmap* as described on the project clarification handout of October 27.
- The Bode plot should be obtained using the command *bode()*.

**Part III: Laplace Transform**

1. Transfer function.
2. Impulse response.

Notes:

- Derivation of transfer function should start from scratch and show all the appropriate steps.
- Documents the steps you took in obtaining the impulse and step responses. Show the partial fraction expansion you used. You can use a matlab command such as *roots()* to factor polynomials, but you should not use a program (or your calculator) to do the partial fraction expansion for you (but of course you can use a program to check that your work is correct—I also don’t mind if you use a computer to help with the complex math).
- For the Bode plot, first write the transfer function where the denominator is not fully factored. Factor out the single pole and leave the quadratic part as it is. Then follow the rules in Appendix E for real and complex poles (this part needn’t be hard, so don’t make it hard!).
Part IV: Convolution

1. Plot of step response obtained using \texttt{conv}().
2. Comparison with step responses obtained using other techniques.

Notes:

- See the \texttt{conv()} handout of November 8th.
- The comparison can be qualitative or quantitative.

You should close with some concluding remarks about your assessment of the utility of the three methods that were considered here: classic differential equation approach, state-space method, and Laplace transform technique.