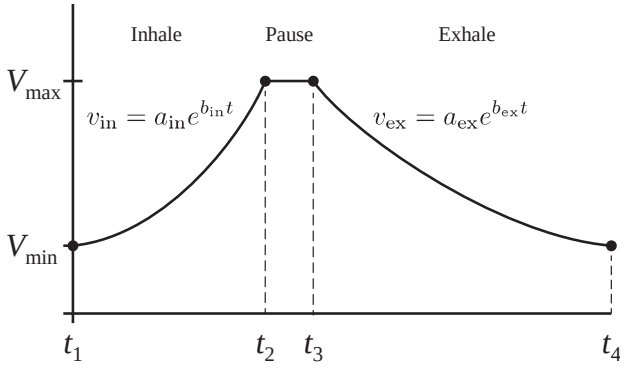
ME 120 Homework 4 due 12 Feb 2018

Use the Direct Solution Format for all problems on this assignment. Use a word processor to include the source code for all problems involving programming. The first sheet of the source code listing should have your name, the assignment number and the date. The first page does not need to be a separate cover sheet. The problem number and name of the Arduino sketch should be clearly labeled for each problem. In other words, do not just print the source code on otherwise unlabeled sheets of paper. Print the word-processor file and include it with any other sheets of paper you submit as part of your assignment. Do not email the document to your instructor.

---

1. (5 points) Choose values of V_{min}, V_{max}, t_2, t_3, and t_4 to define a model of the breathing LED as depicted in the plot to the right (we will set t_1=0).



   a. What are your values of V_{min}, V_{max}, t_2, t_3, and t_4?
   b. For the value of V_{min}, V_{max}, t_2, t_3, and t_4 obtained in the preceding step, what are the coefficients of the two exponential functions for the inhale phase (a_{in} and b_{in}) and exhale phase (a_{ex} and b_{ex}) of the breathing cycle?
   c. Use Excel to make a plot of the breathing cycle. Your plot should look like the one in the problem statement above.

2. (5 points) Build the breadboard circuit, and write an Arduino sketch to implement the breathing LED described in the class notes. Use the parameters and equations obtained in your solution to the preceding problem. Declare the parameters as variables in the header of your program. Have your code running and ready to show your instructor at the start of class on the due date.

   On the solution you turn in for grading, include your program listing in a word-processing document.

Rubric:

- +1 point if all components are there and connected (to be checked by instructor)
- +1 point if it looks like it is breathing (to be checked by instructor)
- +3 points if code is correct (to be checked by TA when grading):
  - Variables are set in header
  - If/else statement is used correctly to account for the 3 different phases
  - Equations look correct in the if/else statement

3. (5 points) Write an Arduino sketch that prints the integers in increasing order from 1 to n and then in decreasing order from n-1 to 1. Print all of the values on a single row (not a column), and leave at least two blank spaces between each value on the row. In other words, write a program to create the following output in the Serial monitor when $n = 7$.

```
1  2  3  4  5  6  7  6  5  4  3  2  1
```

In your solution for *this* problem, use $n = 7$, but declare n as variable. See problem 4 for the motivation.

Have the printing occur *only once* by putting the code in the setup() function, and leaving the loop() function empty. The following shell of an Arduino sketch at the top of the following page shows the basic structure of the solution.

```
void setup() {

  int n = 9;

  // -- your code for problem 3 goes here

}

void loop() {}    //  Leave loop() empty
```

On the solution you turn in for grading, include your program listing in a word-processing document.

Hint: Since the program executes once, you may not always catch the output in the serial monitor. In that case, just press the reset button on the board, which will cause the program to start again.

4. (5 points) Extend the preceding program by repeating the print out to values of n=4, 6, 8, 10 in the same program. The output of your program in the serial monitor should look like the following:

```
1 2 3 4 3 2 1
1 2 3 4 5 6 5 4 3 2 1
1 2 3 4 5 6 7 8 7 6 5 4 3 2 1
1 2 3 4 5 6 7 8 9 10 9 8 7 6 5 4 3 2 1
```

On the solution you turn in for grading, include your program listing in a word-processing document.