Massachusetts Institute of Technology
Department of Electrical Engineering and Computer Science
Department of Mechanical Engineering

6.050J/2.110J **Information and Entropy** **Spring 2004**

Issued: February 4, 2004 **Due: February 6, 2004**

Problem Set 1

Laboratory Assignment

In 6.050J/2.110J, MATLAB will be used frequently in problem sets and demonstrations. MATLAB is a mathematics software package that is efficient at matrix operations as well as a good tool for data visualization. Access MATLAB on any Athena workstation by typing `add matlab` at your `athena%` prompt. Then type `matlab &` to enter the program. A separate window will appear, in which you can enter MATLAB commands.

Whenever a problem requires MATLAB, write a text file that holds the commands or functions that you would normally type into the MATLAB command line. This text file, also known as an M-file, should have a `.m` extension (e.g., `filename.m`). This file can be executed in the MATLAB window by typing the filename without the `.m` extension. This takes away the drudgery of retyping commands over and over again. Programs like vi, textedit, emacs, and pico can be used to create and edit the M-file on Athena. Moreover, we also request that you type `diary` (in MATLAB) which keeps track of all your commands and outputs from MATLAB. The results are placed in a file called diary. Please create a separate M-file for each problem and edit the diary for readability.

Go through the MATLAB Tutorial (http://www-mtl.mit.edu/Courses/6.050/notes/matlab.pdf) given out in class to familiarize yourself with the syntax and environment. Remember that at any time, you can type `help` at the MATLAB prompt (before typing this, type `more on` to turn on page-by-page viewing). It is our intention that MATLAB be used as a helpful tool in this course, and that it not be a barrier to anyone's success in the course. We also hope that you'll develop skill with MATLAB which may be useful to you in other classes. To that end, we're here to help you if you run into problems, so feel free to ask us for assistance. If you have any questions, please email 6.050-staff@mit.edu.

Problem 1: Ill-Logical

The doctor uses a simple technique for determining if a patient who has been bitten by a dog is at risk of contracting rabies. The patient can be reassured that they will not get sick if (1) the wound is not a puncture wound, or (2) it is a puncture wound but the dog has recently been vaccinated. Thus the doctor's formula is:

$$\overline{P} + (P \cdot V) \tag{1–1}$$

where the overbar denotes logical negation (the *NOT* function). The nurse, however, does it differently. He determines that the patient will be fine if (1) the dog has been vaccinated or (2) the wound is not a puncture wound. This formula for this procedure is:

$$V + \overline{P} \tag{1–2}$$

Are these two procedures equivalent? Using MATLAB, test whether they are equivalent for all possible values of P and V. Hint: MATLAB has functions *OR*, *AND*, and *NOT*. These can be applied using `|`, `&`, and $\sim$, respectively.

Problem 2: Digital Monotonicity

You are a digital design engineer working for Monotonic Increasing Telemetry, Inc. You are asked to investigate the properties of monotonic-increasing gates for use in the digital circuits you design. A monotonic-increasing gate is defined as an n-input to one-output digital logic gate which, upon a change of any single input value from a zero to a one, does not cause the output to change from a one to a zero. That is, when raising an input, the output either stays the same or increases; it never decreases.

a. Which of the sixteen two-input logical functions are monotonic-increasing? Prove, for each function, if it is or is not monotonic-increasing. You may do this either with MATLAB or without.

   To prove that a gate is non-monotonic-increasing, it is sufficient to give a single counter-example. For example, XOR is not monotonic-increasing since we find that if the inputs are $\{A = 0, B = 1\}$ giving an output of 1, and we change A to 1, the output decreases from 1 to 0. Furthermore, it is easy to show that OR is monotonic-increasing: if either or both of the inputs is one, then the output is one. Thus, if either of the inputs increases from a zero to a one the output will either (a) increase from a zero to a one, in the case that both inputs were initially zero, or (b) stay at a one, if the non-changing input was originally a one.

b. Your boss is intrigued by the circuit of OR-gates shown in Figure 1–1, and wants to know the truth table for this topology and whether the circuit is itself monotonic-increasing. Using MATLAB, exhaustively enumerate the truth-table of this topology. Your answer should include all 32 possible input values. Is the circuit monotonic-increasing? (**hint:** look at the `bitor()` and related functions in MATLAB.)

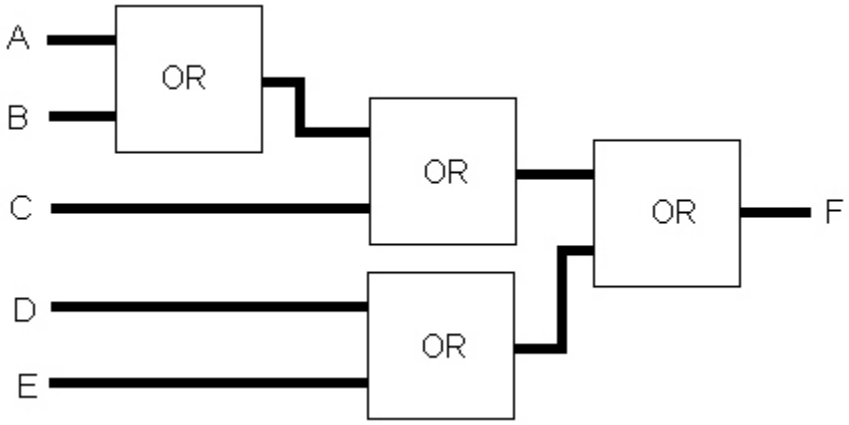

Figure 1–1: Circuit Topology

Turning in Your Solutions

You may have up to three files (two M-files and one diary). Name the M-files `ps1p1.m` and `ps1p2.m` and name the diary `ps1diary`. You can rename your files at your `athena%` prompt using the following command.

```
mv oldfilename newfilename
```

Turn in this problem set by e-mailing your M-files and diary along with your answers to any problem(s) not done using MATLAB, to 6.050-submit@mit.edu. You may do this either by attaching them to the e-mail as text files, or by pasting their content directly into the body of the e-mail (if you do this, please somehow indicate where each file begins and ends). If you decide to turn in your set electronically, be sure to follow the 6.050 Electronic Submission Guidelines. Alternatively, you may turn in your solutions on paper in Room 38-344. The deadline for submission is the same no matter which option you choose.

Your solutions are due 5:00 PM on Friday, February 6, 2004. Later that day solutions will be posted on the Web.