Cracking the Digital Design Interview

Deepak Unnikrishnan

March 9, 2015

Thanks to Vishwas Vijayendra and other members of the Reconfigurable Computing Group, UMass Amherst*De finibus bonorum et malorum*

Preface

This book is a collection of popular digital design interview questions. I was inspired to compile the collection of problems presented in this book during my final year as a graduate student. During this period, I was interviewing several for full-time positions at ASIC/FPGA companies. Soon, I found myself wondering how to best prepare to crack the digital interview. I quickly realized that cracking the interview requires not only a good academic background and work experience, but it also requires a good foundation of fundamental digital design principles. As a first step, myself and my friend, Vishwas Vijayendra started collecting interview questions from popular Internet websites. We compiled these questions in a Google document and shared it internally within our research group. As time went by, many members within and outside our research group have contributed to our collection. To my surprise, I have found many of these problems being reused across interviews in many companies. From my experience as an interviewee, I have found that while having a good resume can get you to the doors of the company, a good understanding of the fundamentals is impeccable to cracking the interview. I would encourage readers to use these problems to test your knowledge rather than using it as a comprehensive interview preparation guide. Useful resources to brush up your fundamental knowledge in the subject area have been provided towards the end of this book.

The collection of problems in this book have been compiled from a variety of sources in the Internet. We don't own the copyright for these problems. The copyright rests with the respective owners, when cited. Credits are provided to the sources of these problems. Good luck!

Contents

Chapter 1

Combinational Logic

Exercise 1.

Using only NAND gates, NOR gates and 2:1 MUX design the following 2 input gates:

a) AND

b) NOT

c) NOR

d) NAND

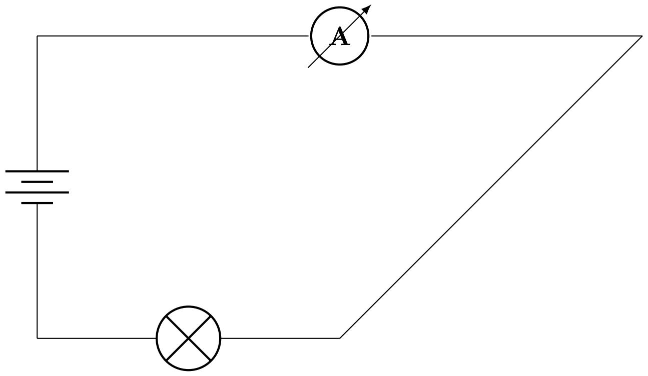
e) OR

f) XOR

g) XNOR

h) 2:1 MUX

Solution 1.



Test sol

Exercise 2.

Describe 2 ways of writing a 4:2 bit priority encoder in Verilog.

Exercise 3.

How do you multiply a 8 bit value by 7?

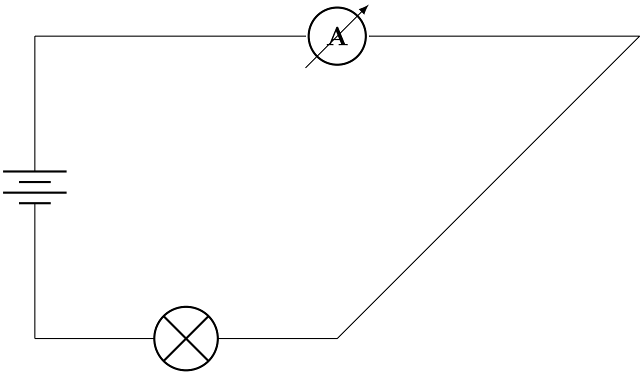
Exercise 4.

Design a 8:1 MUX using 2:1 MUX.

Exercise 5.

How do realize inverter using 2 input NAND gate? Show me 2 ways of doing it.

Solution 1.



Test sol

Solution 2.

Test solution.

Solution 3.

Test solution.

Solution 4.

Solution 5.

Test solution.