Digital Design Review

-an ebook
on the Fundamentals
of Digital Design

Introduction

There are a lot of very fine pieces of hardware and software available that are sorely lacking because they fail to come with adequate support literature. This is not only a problem for the buyer, but also for the vendor who is plagued with questions which shouldn't have to be asked if only the literature were sufficient. Hopefully, the Lab9500 board and accompanying workbook of experimentation will not suffer that problem. So the Lab9500 package is much more than some hardware. It is a complete package that should permit the user to successfully program a modern CPLD device, no simple accomplishment.

The Lab9500 is intended to serve as laboratory equipment for an introductory digital design course. Anything you could conceive of talking about in an introductory course is doable with the 72macro-cell XC9572XL CPLD found on the Lab9500 board. The (very sophisticated) software that the Xilinx 9500 CPLD family needs is available free off the Internet from Xilinx. Anyone taking a digital course will already have a textbook that should provide the design techniques needed to make use of the Lab9500. On the other hand, this board should prove very attractive to engineers, hobbyists and do-it-yourselfers who are not and will not be taking a digital course. It would be unfortunate if they had to purchase a $120 textbook to use the board effectively. Accordingly, although it was far from my intent, I am including a basic "mini" digital design text which I am calling Digital Design Review, as an accessory to the Lab9500 Users Manual. This is not intended to be a rigorous, full-blown course in digital design. It is intended, however, to provide the user with the basic principles of digital design.

Unlike the lab manual, Digital Design Review is not provided as hard copy. It is an e-book included on the CD-ROM. The user may read it on screen, but will find it more convenient to print it out a chapter at a time, as he needs it. The book is in Microsoft WORD format and is easily printed double-sided on most printers. Depending on the printer, you may want to print the even pages or odd pages first before flipping them and printing the other sides. At worst, you may have to reorder the pages. Punch them and put them in a binder.

A few words about digital design books are in order. Books for digital design have not changed as much as you might expect over the years. The most recent books, however, now spend some time talking about PLDs (programmable logic devices). But working on PLD problems without being able to try out your solution is not very satisfying. It's akin to taking a course in a programming language and never being able to compile and try out your programs.

Some books still spend a lot of time giving traditional seat-of-the-pants techniques for reducing Boolean equations that will never be used since Karnaugh maps do it easier and faster. Some books talk about master-slave flip-flops, how they work and how they are made. No one ever has to design one. It's just historical (and superfluous) information. Once you weed out the historical stuff while keeping what is truly valuable, the book gets a lot thinner.

Again, my intent is not to replace a good digital design book, but rather to provide enough information to enable someone who hasn't taken, (and may never take), a digital course, to use the Lab9500 board effectively.

Chapter One starts out talking about basic logic functions such as AND, OR and Invert, which everybody probably already knows, and also three-state gates. Signal conditioning and level shifting are also considered. Various logic families and their characteristics are discussed, but especially CMOS.

Chapter Two is my very abbreviated treatment of Boolean Algebra. While textbooks feel obligated to discuss the POS or product of sums representation of Boolean expressions, and usually give it equal time, in the real world there is no architecture that supports this mathematically equivalent (but functionally not very useful) way of representing logic. I simply mention it and show what it is.

Chapter Three is a discussion of combinational (or combinatorial) logic. This is logic without memory whose outputs are the instantaneous function of the inputs. Many common logic structures such as adders, decoders, encoders, multiplexors, etc. are discussed. Note, that all of these functions are easily implemented with the Lab9500 and with additional features possible not found in standard off-the-shelf digital parts. For example, the ubiquitous 74X138 decoder has low-true outputs, but one could just as easily make a 74X138-style decoder with high-true outputs with the Lab9500.

Chapter Four is a discussion of flip-flops. A flip-flop is the unit memory device, and is the stuff registered or clocked logic is made of. I would like to think that my discussion of flip-flops is superior to that found in most digital design texts.

Chapter Five is a discussion of registered logic, or logic with memory. This is the meaty stuff, the fun stuff. Often this is not particularly well covered in many texts. I talk about the *state machine* and show how to design an arbitrary counter, which is a state machine that goes unconditionally from one state to the next. It is a simple extension to talk about more general state machines whose next states depend not only on the present state, but other conditions.

Chapter Six is a brief introduction to the HDL (hardware description language), ABEL. ABEL has been around a long time and is no longer supported by its originator. It remains, however, the simplest way relating to traditional design techniques to program programmable logic. ABEL also provides an introduction to high-level software techniques that supplant traditional design methods. For example, traditionally a logic function is described with a truth table. The truth table is used to fill a Karnaugh map, and the K-map is used to reduce the Boolean equations to a reasonable number of literals. ABEL permits entering equations as truth tables, eliminating the need for the K-map. ABEL reduces the equations itself. (This takes a little of the fun out of digital design). On the other hand, one can see that spending any time on pre K-map techniques for reducing Boolean equations is a waste of time when there is so much to learn.

ABEL is a generic language intended to be used with ANY PLD family of devices. The writer of a particular ABEL characterization, however, must provide family specific information. Unfortunately, the voluminous Xilinx software and literature is somewhat lacking in spelling out the 9500 family-specific ABEL constructs. This chapter attempts to fill that gap.

As PLDs have evolved so has the need for sophisticated software to program them. The government, foreseeing a need, has invested a lot of money in developing such tools. The two leading software packages that share nearly equally the market for such software are VHDL and VERILOG. It was my intent to support only ABEL for the Lab9500 experimentation. However, it soon became clear that an introduction to one of the more sophisticated HDL's was necessary. Consequently, Chapter Seven is a brief introduction to VHDL. The idea behind VHDL is to modularize PLD development. A particular function is assigned to an 'entity'. An entity is a block whose inputs and outputs are fully specified for use by other entities. An entity has an architecture that describes the functionality of the entity and how it is to be realized. This approach enables several people or groups to work on a large design project where each is responsible for a particular module and needs only to know how it interfaces with other modules (entities).

Finally, materials that may be useful or interesting are contained in the Appendices.

While Digital Design Review will initially be quite small, it is hoped that it will grow as users contribute to it. Accordingly, teachers or even students, who feel that some topic needs to be amplified, or perhaps a different approach taken, should feel free to submit copy for consideration in being incorporated in future editions.