PROGRAMMABLE LOGIC DEVICES IN TEACHING

David Milford*

The use of programmable logic devices in teaching digital electronics allows many worthwhile educational objectives to be achieved. The diversity and low cost of these devices makes them an ideal target for the implementation of small and medium scale digital systems.

1. Introduction

The market for Field-Programmable Logic Devices (FPLD) has shown impressive growth in the past few years, resulting in an increased diversity of low-cost devices. This has significant implications for the education of engineering students, not only because they should acquire familiarity with state-of-the-art technology but also because the use of these devices provides an opportunity to practise modern design methodology. Most importantly, they allow students to progress from a non-trivial (and hopefully interesting) design specification through to a working circuit in a realistic timescale.

2. Field Programmable Logic Devices

FPLDs, as opposed to mask-programmable devices, are application specific ICs (ASIC) which can be designed *and programmed* by the user. They include, roughly in order of increasing structural complexity:

- programmable ROMs (PROM)
- programmable array logic (PAL)
- programmable logic arrays and sequencers (PLA/PLS)
- generic array logic (GAL)
- programmable gate arrays (PGA)

Several of these, including some types of FPGA, are reprogrammable and therefore very economical to use in an educational context (although there may be other reasons for avoiding their use).

3. Design

Computer aided design (CAD) tools play an essential role in the use of FPLDs and they

* Department of Electrical and Electronic Engineering, University of Bristol, University Walk, Bristol BS8 1TR, UK

encourage a design methodology which applies equally well to other styles of ASIC implementation. Depending on the target device the toolset is likely to include:

- schematic capture
- use of Boolean equations for describing combination logic
- state machine description for sequential logic
- hardware description language (e.g. VHDL)
- simulation facilities

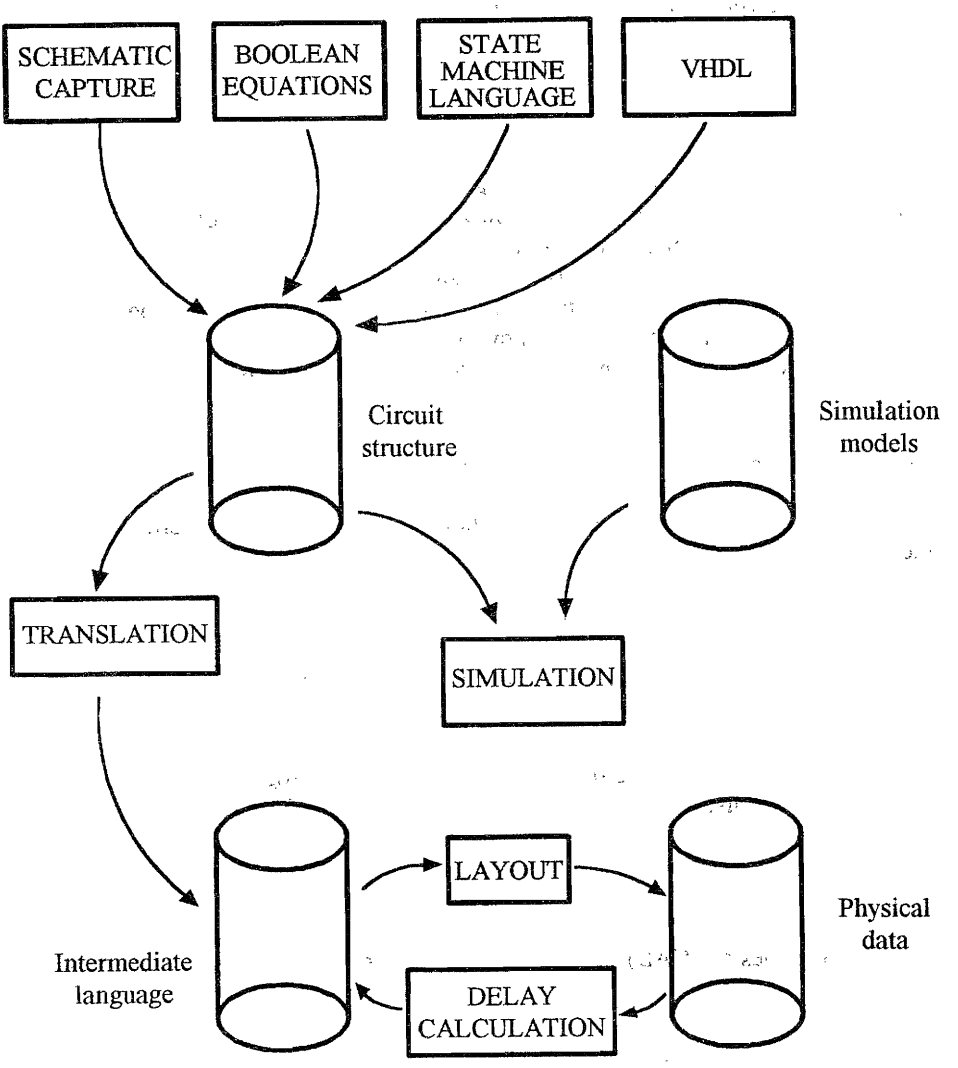

Fig. 1. Design procedure.

Many CAD tools for semicustom design can now be run on IBM compatible PCs and, for the complexity of design which is likely to arise at undergraduate level, a well-specified PC (386/486) is quite adequate.

A number of tools (e.g. CUPL, ABEL) exist to support designs targeted at a wide range of the less complex PLDs. Other vendor-specific packages (e.g. AMD/PALASM), though slightly limiting, are very good value. In the case of the more complex devices such as FPGAs it is necessary to obtain vendor-specific software to support the physical (placement and layout) design. In order to maintain flexibility, several "front-end" CAD packages (e.g. ViewLogic, OrCAD) are available for performing schematic entry and simulation. They provide interfaces to the more specific tools mentioned previously.

The overall design procedure is outlined in Figure 1.

4. Implementation and Testing

Circuit implementation has traditionally presented a problem in an educational context. Construction using prototyping techniques with discrete logic (standard SSI/MSI parts) is time-consuming and, except for very small designs, of little educational value. The level of integration provided by FPLDs makes it feasible to build and test reasonably complex designs. The emphasis on testing can be varied: for short laboratory exercises, typically using a PAL or GAL, the programmed device can be tested in a purpose-built circuit; in larger projects testing becomes an integral part of the design problem and must be taken into account from an early stage.

It is worthwhile, at this point, to return to the issue of reprogrammable components in relation to simulation and test procedures. It has been suggested that reprogrammability encourages an informal approach in which a design is iteratively debugged at the implementation level rather than during simulation. The point is that verification prior to implementation is essential for mask-programmable ASICs where each mask iteration adds a large cost. On the other hand, an increasingly small proportion of designers is likely to be involved in this technology if the capability of reconfigurable logic devices continues to develop at the current pace. Design styles will continue to evolve to match the available technology and reprogrammability is bound to shift emphasis in the design cycle. The need remains for a structured design methodology with simulation a key component; the diagnostic information required to debug an integrated circuit design is much harder to obtain at the implementation level.

5. Experience at Bristol

In our undergraduate courses for electronic and computer systems engineers we use FPLDs in a number of design projects .

In one laboratory exercise of 6 hours duration the students are required to design the control logic for a simple serial data receiver. Using algorithmic state machine techniques with computer assistance provided by CUPL, the logic is implemented on a GAL16V8 device which plugs into a ready-made board for testing.

A longer project, undertaken over a period of 5 weeks in the second year of the course, allows enough time for the design, implementation and test of a complete digital subsystem with a complexity of 500 - 1000 gates. The circuit is implemented on a Xilinx FPGA using VIEWLOGIC for schematic capture and simulation. In this exercise the students work in small teams and partition the design themselves, defining the interfaces between constituent blocks. The specification for the project has often been provided by visiting engineers from industry and has included in recent years:

- a PS/2 microchannel bus interface to a transputer link adaptor
- a CRC generator/checker
- various types of digital image filter

6. Educational Objectives

Through the use of FPLDs in design projects we feel that we satisfy a variety of worthwhile objectives. Most derive from the principle of active student involvement in the complete design process from specification through to implementation and test, namely:

- increased student motivation
- need for planning; top-down design, design for test
- exposure to semicustom design methodology
- an awareness of design constraints imposed by target technology
- appreciation of capabilities and limitations of CAD
- development of team skills

Bibliography

Bolton M. (1990): *Digital System Design with Programmable Logic.*- Addison-Wesley.

Logical Devices Inc. (1991): *CUPL, PLD/FPGA Language Compiler.*

Moore W. and Luk W. (Eds.) (1991): *FPGAs.*- Abingdon EE & CS Books.

National Semiconductor Corporation (1990): *Programmable Logic Devices Databook and Design Guide.*

Xilinx Inc. (1991): *The Programmable Gate Array Data Book.*