

# Local and Remote Laboratory User Experimentation Access using Digital Programmable Logic

J. Murphy\*, I. Grout\*, J. Walsh\*, T. O'Shea\*

\*University of Limerick/Department of Electronic and Computer Engineering, Limerick, Ireland

**Abstract**—This paper will discuss the structure and operation of a programmable logic based experimentation arrangement that is suitable for both local and remote teaching and learning scenarios targeting electronic and microelectronic circuit design and test principles. With this experimentation arrangement, the ability to provide both local and Internet based “remote” access for the student and the teacher can provide a number of advantages where physical laboratory accessibility is limited and/or the learning experience must be undertaken with one or more of the parties remotely based. The paper concentrates on the design and example use of a system developed within the University of Limerick.

**Index Terms**—Remote laboratory, microelectronics design and test education

## I. INTRODUCTION

Remote learning [1][2] has matured over the last number of years to provide a realistic and important support mechanism in which practical laboratory based experimentation work can be undertaken by remote learners, who are provided with access to facilities that they would not otherwise necessarily be able to utilise. The emphasis has been on providing an enhancement to the student learning experience by adopting the Internet as a suitable means in which to provide a communications channel between the learner, teacher and hardware/software experimentation facilities for a remote learner. This type of facility is particularly important in the engineering and science disciplines. The ability to design, develop and utilise experimentation based on remote learner requirements is an exciting and important step in the evolution of, and increasing access to, teaching and learning resources. The ability then to design, develop and utilise a core experimentation arrangement that is accessible by both local and remote users, providing flexibility in the location of both the teacher and learner, is a step in the direction to provide for cost-efficient experimentation. Whilst the technological ability to support remote learning exists in a number of forms, and has been widely demonstrated, the economics of providing such a learning support mechanism cannot be underestimated, and must be realistic in the context of course provision. Experimentation set-ups with multiple modes of use can aid a reduction in the cost of such

laboratory equipment: the same set-up can be used for a range of experiments thereby reducing the cost of laboratory equipment and the equipment set-up time for each experiment. This paper will discuss the use of the remote experimentation arrangement as applied to the area of electronic/microelectronic circuit test engineering education, and will discuss the development of the experimentation arrangement aimed to provide a common experimentation arrangement to support both local and remote users. The paper is structured as follows. *Section II* will look at the need for a better focus on test engineering education. *Section III* will show the importance of interactivity in the learning environment. *Sections IV* and *V* introduce the experimentation arrangement and Internet access equipment developed. *Section VI* will place the work into context in the support of local and remote users. *Section VII* will discuss the web page interface design and the processes that occur when a command is executed. *Sections VIII and IX* will look at the operation of the arrangement in both local and remote modes. *Section X* will conclude the paper. The discussions presented will concentrate on the hardware experiment arrangements and web based user interface design and support.

## II. THE NEED FOR MICROELECTRONIC CIRCUIT TEST ENGINEERING EDUCATION

The increasing design complexity and reduced error margins in semiconductor manufacturing are forcing design and test engineers within industry to be ever diversifying in the range of activities undertaken on a day-to-day basis. With Integrated Circuit (IC) designs becoming more complex, they require an optimal solution for design, fabrication and test. Despite the increasing design complexities, no matter how complex an IC design becomes, testing of the design relies on the understanding and effective utilisation of basic concepts, which need to be obtained from suitably defined 3<sup>rd</sup> level (University/Institute of Technology) education. There are a number of directions that can be followed in the area of microelectronic test education, depending on the skills set required to be developed by the students: consideration must be given to whether it is hardware or software test that is the primary focus. In this work, electronic hardware test is the primary concern. In hardware test,

the main areas of concern are device (Integrated Circuit (IC)) level and the system level. Today, however, it should be noted that the boundary between device and system is less well defined with the advent of the “*System on a Chip*” (SoC), where the concern is to integrate as much of an electronic system as possible onto a single silicon die. The teaching of test engineering at 3<sup>rd</sup> level is critical to providing the students with the necessary skills to graduate into a range of industry based engineering positions, primarily those who will be focused towards associated test engineering activities, but also those who may be involved in activities from design through to production development and support. Dedicated test engineering taught modules within a degree scheme of study need to address device level test, system level test and production test systems. The correct amount of local and possibly remote learning also needs to be addressed. Table I discusses requirements on the educational side of each of the test engineering areas mentioned.

TABLE I.  
KEY AIMS OF TEST ENGINEERING MODULES

Device Level Test	System level test and production test systems
Provide a discussion into, and application of, test engineering concepts in the testing of Integrated Circuits during, and post-fabrication.	Provide a discussion into, and application of, test engineering concepts in the testing of electronic systems (PCB, SoC, SiP and MCM based) and the Automatic Test Equipment (ATE) requirements.
To introduce the design and testing of specific circuit architectures commonly encountered in analogue, digital and mixed-signal ICs.	To discuss system level test issues, the causes, effects and detection of system level faults.
To introduce and discuss <i>Design for Testability</i> (DfT).	To discuss system level test, in particular the structure and use of the IEEE 1149.1 (Digital Boundary Scan) & 1149.4 (Mixed-Signal Test Bus) standards and implementation.
To practice the <i>build &amp; test</i> of electronic circuits within structured laboratory sessions emulating analogue, digital and mixed-signal designs that would normally be encountered.	To practice simulation, fault simulation and DfT insertion of electronic circuits within structured laboratory sessions emulating analogue, digital and mixed-signal designs that would normally be encountered.

### III. THE IMPORTANCE OF INTERACTIVITY WITHIN A LEARNING ENVIRONMENT

In recent years, many universities and organisations have ventured into the area of remote learning and laboratories, where some universities and organisations are devoted to providing curriculum delivery through distance learning (e.g. The UK Open University [3], Electronic Design Education Consortium (EDEC) [4] and FÁS eCollege [5]). Remote learning environments can provide a number of advantages over a conventional classroom; for example, in a conventional classroom environment, some students can be reserved and shy by nature, and are

reluctant to ask questions. In a remote learning class with suitable structure and technical support, the reserved students may be less inhibited and so contribute to discussions, so enhancing their personal learning experience. Additionally, an important concept addressed by research on remote learning is that of interactivity. For example, it was found that students' retention of knowledge increased from 20% to 75% of the total amount of information when students interact with the teaching materials [6].

Educators now need to investigate ways in which to improve on the traditional learning procedure, where the first step would be a lecture supported by a laboratory session. A possible alternative is that after an introductory lecture, the next lectures could be done using a computer based learning (CBL) environment. CBL can be defined as “any situation in which a computer enhances and assists the overall learning experience” [7][8]. The use of a CBL environment in microelectronic circuit test education can have several advantages over traditional learning methods. For example, the students have greater flexibility, and can use different time frames for the learning of each block of new material as a function of their personal requirements. The learning environment may also be task driven, where on completion, allow the students progress to the next lecture. Learning environments can also allow for greater interaction through visual tools as suggested in figure 1. At home, the student can use the Internet to have access through a University firewall to a complete set of tools linked in for each lecture, for example a practical session using a remote engineering laboratory. The lecturer need only give a lecture, after the students acquire a certain amount of knowledge about the subject. The lectures may then have a different function, which may have more a role of discussion (tutorial) about the subject giving time for creative debates, which can be very productive.

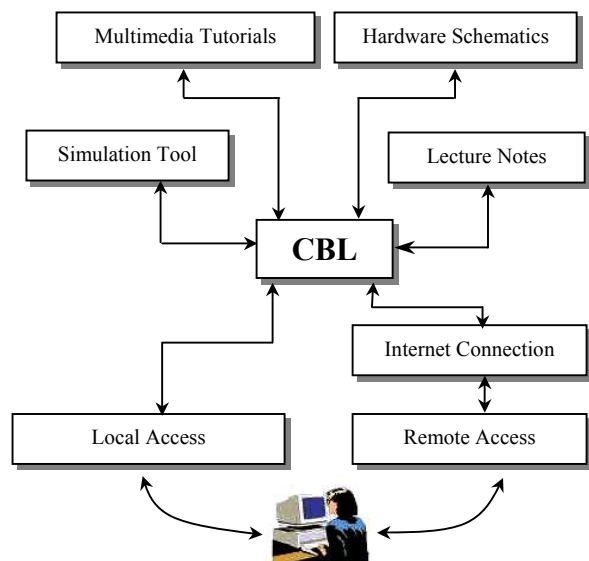


Figure 1. Example CBL tool structure

## LOCAL AND REMOTE LABORATORY USER EXPERIMENTATION ACCESS USING DIGITAL PROGRAMMABLE LOGIC

The progression of each student will give the lecturer feedback, regarding the benefits, and drawbacks of the remote learning environment.

In the learning, for example, of the use of microelectronic CAD (Computer Aided Design) tools (design capture (schematic/HDL (Hardware Description Language)), analysis/simulation and physical realization), the tool should be highly integrated with the subject lectures. It is evident that some features from a CAD tool may provide more realistic demonstrations in a CBL environment, than those found solely within a textbook. An interesting graphical user interface (GUI) can help the student to enjoy learning the particular subject or CAD tool. Another advantage of learning microelectronics using a computer-based solution is the ability to perform simulation studies. This makes it possible to build and test models of microelectronic circuits, and lets the students check their physical and behavioral structures through simulation.

### IV. PROGRAMMABLE LOGIC BASED LABORATORY EXPERIMENTATION

The basic system here consists of a circuit hardware prototyping system, which once configured (via a PC), can be run, in conjunction with, or independent to, a user PC. The system was originally developed for local learning scenarios. The core of the system is based on the Lattice Semiconductor Inc. [9][10] MACH4A CPLD (Complex Programmable Logic Device). This is configured via the PC parallel port and the JTAG (Joint Test Action Group) boundary scan interface within the CPLD. The system hardware contains support circuitry including a visual display and 8-bit resolution data converters (Analogue to Digital (A/D) and Digital to Analogue (D/A))[11][12]. When the user is ready, a new or a previously designed circuit would be downloaded on to the CPLD. The user can produce a test vector set for the circuit using software program with a suitable Graphical User Interface (GUI) in order to determine the functionality of the design via suitable test vectors for the particular design. This system can provide a new fast and effective way of teaching microelectronic design and test engineering concepts.

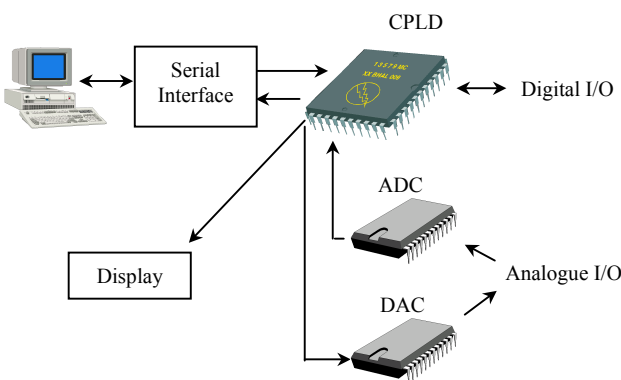


Figure 2. System arrangement

The system arrangement for this microelectronic circuit engineering laboratory environment is shown in figure 2. This is referred to as the University of Limerick Test Engineering Development Box (ULTEDB)[13] environment, see figure 3. The basic principle of operation is as follows: (1) A design is downloaded into the CPLD (typical digital designs intended to highlight specific aspects relating to both design, test and Design for Testability (DfT)). The circuit may be configured to be a normal fault-free circuit, or to include specific logical circuit faults. (2) The student is then required to analyse the design, develop and provide test vectors in order to undertake test procedures on the design. The test vectors are either applied manually (via switches/external logic) or automatically via the PC.

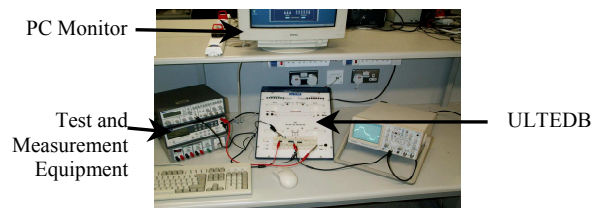


Figure 3. ULTEDB environment

### V. WEB SERVER ARRANGEMENT

The web server arrangement [14] is based on the Apache Web Server, with PHP scripting, a MySQL database and Visual Basic™ (VB) application programs. This arrangement allows for Internet access (web page I/O for the user) and experimentation hardware access (for this arrangement, via the PC RS-232 serial port). The system arrangement is shown in figure 4.

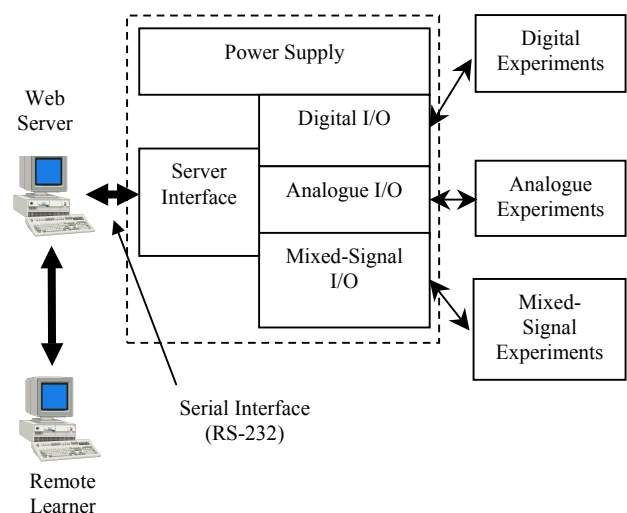


Figure 4. Web server environment [5]

With this arrangement, all the necessary hardware and software interfacing exists (i.e. in a modular, template form) such that extensions to the experimentation arrangement can be readily undertaken in order to extend the functionality of the system. This system was upgraded to facilitate connection of the ULTEDB and to

## LOCAL AND REMOTE LABORATORY USER EXPERIMENTATION ACCESS USING DIGITAL PROGRAMMABLE LOGIC

provide for both local and remote access. Figure 5 shows the arrangement now considered in this work.

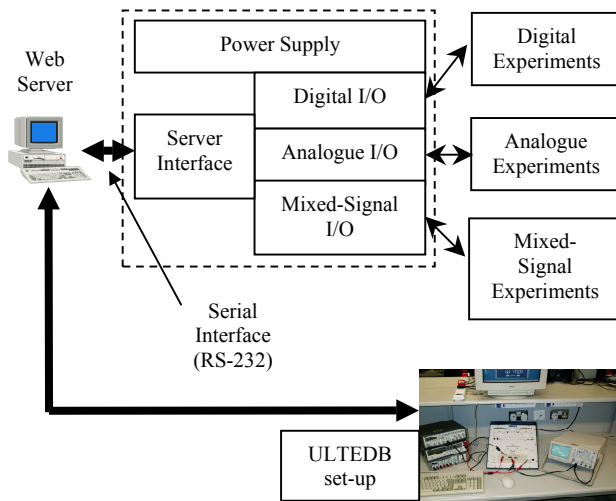


Figure 5. Extended web server environment

### VI. REMOTE AND LOCAL ACCESS MODES

A requirement for the work undertaken was to design, develop and utilise the hardware/software solution by developing as much of the hardware and software as practical. This has cost implications and the ability to re-use as much of the system as possible in order to extend the functionality was considered a benefit. In order to consider the re-use of the experimentation, it is possible to envisage the following modes of operation for both the teacher and learner, see figure 6.

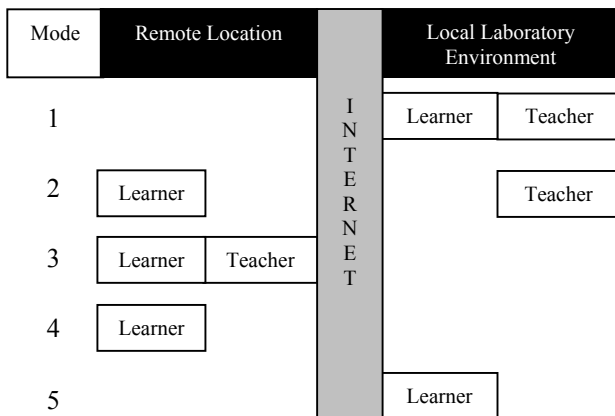


Figure 6. Modes of operation

Figure 6 shows 5 modes of operation that could be undertaken in order to access the hardware experiments:

- In Mode 1, this is seen as the “traditional” laboratory mode in that both the teacher and learner are locally based and have a “hands-on” utilisation of the experimentation.
- In Mode 2, the learner is remotely based, accessing the experimentation via the Internet

and the teacher provides the necessary support in a tutorial style of interaction.

- In Mode 3, both the teacher and learner are remotely based and access the experimentation via the Internet in different modes (a teacher and a learner mode).
- In Mode 4, the learner is remotely based, but there is no direct teacher support.
- In Mode 5, the learner is locally based, but there is no direct teacher support.

The system arrangement identified and discussed in Sections IV and V provides the necessary base hardware/software to facilitate these potential modes of operation. A number of important considerations exist, include:

- Flexible use.
- Support local and remote teaching and learning scenarios.
- Custom solution – both hardware and software in order to minimise the need for specific hardware/software licensing agreements.
- Minimal set-up and maintenance requirements.
- A library of case-study circuits is available – including circuit operation in both fault-free and specific circuit fault conditions.
- Ability to include an automatic self-test routines and system “health” reporting to the teacher/technical support staff.

### VII. WEB PAGE DESIGN AND OPERATION

The web page interface is designed using PHP as the coding language with a MySQL database providing storage of results and user profiles and as a queuing system for the execution of laboratory control commands. As stated in Section V, there are also VB application programs that perform the passing and retrieving of information, through the RS-232 interface, between the server and the hardware experimentation.

The interface page allows a remote user to submit a series of inputs as a text file. Another text file is then created in a specified manner with the submitted inputs included. There is an entry then made in a database table, queue, and this specifies the location of the newly created input file. The table queue is checked by a PHP script which is executed by scheduling software once every minute, and if there is an entry in the queue table, the appropriate VB application program is executed by an internal system command using the created input file as input parameters which then executes on the ULTEDB hardware setup. The ULTEDB then outputs the results, which are inserted into a results file by the VB application program and saved to a known location. The PHP script can then use this results file to display the results to the remote user as raw text format, and also to display the results in a graphic form.

# LOCAL AND REMOTE LABORATORY USER EXPERIMENTATION ACCESS USING DIGITAL PROGRAMMABLE LOGIC

As described in Section VI, in mode 3, as well as the remote user being able to access the experiment for learning purposes, the teacher can access the hardware remotely in a different manner. Here, the teacher can program a new circuit design into the ULTEDB. The teacher only needs to select one of the predefined circuit configurations and ULTEDB is reprogrammed to that configuration. When the teacher selects a new circuit design, the PHP script executes an internal system command which starts the software for the CPLD and changes the circuit design.

## VIII. LOCAL ACCESS MODE

A local access case study laboratory was set-up and evaluated by a group of test engineering students from the BSc in Electronic Systems course in the academic year 2003-2004. The aim of this laboratory session was to investigate the operation and testing procedure of a Linear Feedback Shift Register (LFSR) circuit using the ULTEDB development platform. The Circuit Under Test (CUT) used is shown in figure 7.

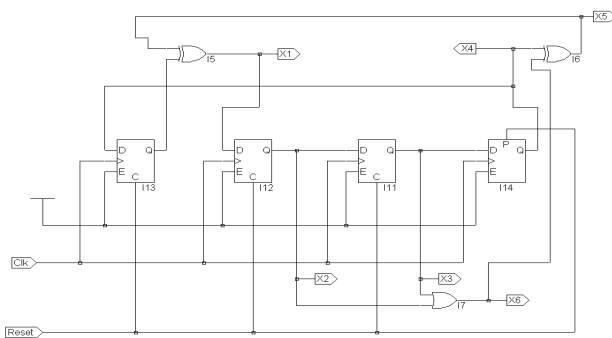


Figure 7. Linear feedback shift register (LFSR)

The CUT was to be considered as a single complex Integrated Circuit (IC) where access to control and monitor of the circuit operation was via the primary inputs and outputs only, as shown in figure 8.

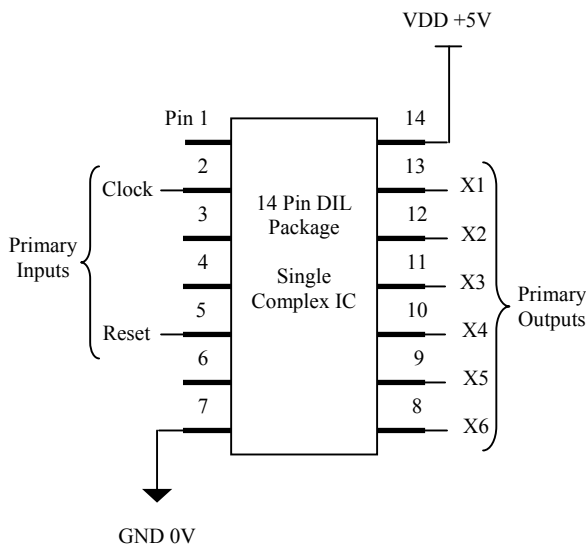


Figure 8. Single complex IC equivalent

The experiment was laid out to investigate design and test issues relating to a Linear Feedback Shift Register circuit, in particular circuit design issues which may result in certain faults being undetectable.

The ULTEDB CPLD contained the LFSR circuit. The students were instructed to follow the laboratory manual to investigate whether a single Stuck-At-Fault (SAF) existed in the circuit or not. The students used the ULTEDB development platform, and a specifically designed VB user interface to apply a series of test patterns in order to reset and clock the LFSR into each state, and record the state truth table. This was then compared to the theoretical state truth table that was also determined, and then identified the possible fault or faults in the circuit. See table II for a list of additional laboratory exercises.

TABLE II.  
LIST OF LABORATORY ADDITIONAL EXERCISE EXAMPLES

I. Using the circuit schematic in figure 7, determine the state truth table for fault-free operation of the circuit. Sketch the resulting truth table in the space below. Show all rough work, use the additional rough work sheets at the back of the laboratory notes if required.
II. In the circuit implementation in figure 7, how many clock pulses does it require before returning to the reset state?
III. Consider the Stuck-At Fault (SAF). Each output node in the circuit may be considered to be either: <ul style="list-style-type: none"> <li>• Fault-free</li> <li>• Stuck-At Logic 0 (SA0)</li> <li>• Stuck-At Logic 1 (SA1)</li> </ul> <p>After the reset pin is sent low, if there was a (SA0) fault in output node X3 many clock pulses would it require before you could detect that fault. Repeat this for X6 (SA1).</p>

## IX. REMOTE ACCESS MODE

As discussed, the experimentation set-up has the potential to be utilised in a number of modes and for a range of electronic hardware design and test related scenarios. In the local access scenario, the ULTEDB was set up as a LFSR and accessed locally by the students. Where the students would be required to access the ULTEDB remotely, a different design example, a priority encoder, has been considered. Table III identifies the truth table for an 8-input encoder, with an example VHDL description in figure 9.

TABLE III.  
PRIORITY ENCODER TRUTH TABLE

INPUTS								OUTPUTS		
A7	A6	A5	A4	A3	A2	A1	A0	B2	B1	B0
0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	1	0	0	0	1	0
0	0	0	0	1	0	0	0	0	1	1
0	0	0	1	0	0	0	0	1	0	0
0	0	1	0	0	0	0	0	1	0	1
0	1	0	0	0	0	0	0	1	1	0
1	0	0	0	0	0	0	0	1	1	1



```

-----
-- VHDL Description of an 8-bit priority
-- encoder
-----
-- The design has 8-inputs (A<7:0>) and
-- two outputs. The main output (B<2:0>)
-- is a binary count output from 000 to
-- 111. The second output (Valid) is a
-- logic '1' only when a valid input code
-- is received.
-----
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

-----
-- VHDL Entity 'encoder'
-----
entity encoder is
  Port (A : in std_logic_vector(7 downto 0);
        B : out std_logic_vector(2 downto 0);
        Valid : out std_logic);
end encoder;

-----
-- VHDL Architecture 'simple'
-----
architecture simple of encoder is

begin

  B <= "000" when A = "00000001" else
      "001" when A = "00000010" else
      "010" when A = "00000100" else
      "011" when A = "00001000" else
      "100" when A = "00010000" else
      "101" when A = "00100000" else
      "110" when A = "01000000" else
      "111" when A = "10000000" else
      "000";

  Valid <='1' when A="00000001" or
                  A="00000010" or
                  A="00000100" or
                  A="00001000" or
                  A="00010000" or
                  A="00100000" or
                  A="01000000" or
                  A="10000000"

                  else '0';

end simple;

-----
-- End of File
-----

```

Figure 9. VHDL description of the encoder design

Such a circuit is a useful combinational logic design in the learning of the following:

- Digital combinational logic design (truth-tables, Boolean logic).
- Hardware Description Languages (HDLs) – Verilog-HDL [15] or VHDL [16] - types of HDLs available, design development, code syntax, coding styles and HDL based design flows.

- The use of programmable logic for implementing digital designs (concepts of programmable logic, schematic entry and HDL synthesis, simulation).
- Digital logic test (test stimulus generation, fault modelling, functional and structural test, test program creation).

The students can log onto the ULTEDB web page and obtain a specific lab manual for each experiment. Using the lab manual instructions, the students can develop a fault matrix and from that can determine a minimal set of test vectors for the encoder.

The students then create a text file containing the set of test vectors determined from the fault matrix, and apply them to the ULTEDB hardware remotely using the web page. The ULTEDB outputs are then recorded in a results file and emailed to the student. Using the results that are received, the students can then determine whether a stuck-at-fault exists in the circuit or not.

## X. CONCLUSIONS

An extension to a remote learning experimentation arrangement has been discussed in which a CPLD based programmable logic experiment has been added in order to provide both local and remote access modes of operation. This extends the functionality of the developed experiments and provides a demonstrator for a flexible experimentation arrangement that was developed to support the teaching and learning of electronic and microelectronic circuit design and test engineering concepts. The modes of operation, local and remote, show how test engineering students, can access the ULTEDB arrangement and successfully carry out experiments.

## ACKNOWLEDGEMENTS

The authors would like to thank the University of Limerick and the University of Limerick Foundation for the funding to support the initial support of the Remote Test Laboratory access project, along the TRIP initiative managed by the Centre for Teaching and Learning at the University of Limerick.

## REFERENCES

- [1] Palloff R. and Pratt K., *The Virtual Student - A Profile and Guide to Working with Online Learners*, Jossey-Bass, 2003, ISBN 0-7879-6474-3
- [2] Coleman J. et al., *Effectiveness of Computer-Aided Learning as a Direct Replacement for Lecturing in Degree-Level Electronics*, IEEE Transactions on Education, Vol. 41, No. 3, August 1998, pp177-184
- [3] The UK Open University, <http://www.open.ac.uk/>
- [4] Electronic Design Education Consortium, <http://edec.brookes.ac.uk/>
- [5] FÁS eCollege, [www.fas.ie](http://www.fas.ie)
- [6] Ricardo Augusto da Luz Reis, Leandro Soares Indrusiak, *Microelectronics Education Using WWW*, IEEE International

## LOCAL AND REMOTE LABORATORY USER EXPERIMENTATION ACCESS USING DIGITAL PROGRAMMABLE LOGIC

- Conference on Microelectronic Systems Education, Arlington, Virginia, July 19 - 21, 1999
- [7] Sixth International Conference on Computer Based Learning, University of Cyprus 5-10 July 2003, <http://www.ucy.ac.cy/cblis2003/>
- [8] Giorgio Da Bormida, Domenico Ponta, and Giuliano Donzellini, Methodologies and Tools for Learning Digital Electronics, IEEE Transaction in Education, November 1997, Vol. 40, NO. 4, IEEDAB (ISSN 0018-9359)
- [9] *MACH4A CPLD*, Lattice Semiconductor Corporation, <http://www.latticesemi.com/>
- [10] Venkateswaran R. and Mazumder P., *A survey of DA techniques for PLD and FPGA based systems*, Integration, the VLSI journal, No. 17, 1994, pp191-240
- [11] I A Grout, J Walsh, T O'Shea, M Canavan, "A Laboratory on a Chip for Test Engineering Education", Proceedings of the 12th Sensors & their Applications (S & A XII) Conference, University of Limerick, Limerick, Ireland, 2 - 4 September 2003
- [12] Stevenson, Tony, - Programming an online education: Part 1, ZDNet Australia, 4th April 2002.
- [13] Joseph Walsh and Ian Grout, *An Investigation Into The Requirements Of A PC-Based Learning Environment For The Effective Education Microelectronic Test Engineering*, Proceedings of the 4th Annual Irish Educational Technology Users Conference (EdTech 2003), Waterford, Ireland, 22nd-23rd May 2003
- [14] I Grout and J Walsh, *Overview and Development of a Remote Electronic Circuit Test Engineering Experimentation Laboratory*, Proceedings of the 2003 Interactive Computer Aided Learning (ICL2003) Workshop, Carinthia Tech Institute, Villach, Austria, 24 - 26 September 2003.
- [15] IEEE 1364-1995, IEEE Standard Verilog Hardware Description Language, IEEE, USA
- [16] IEEE Std 1076-2002, IEEE Standard VHDL Language Reference Manual, IEEE, USA