Testing and Burn-In of Actel FPGAs

Introduction

Burn-in tests, for operating devices dynamically at a high temperature and extrapolating the failure rate to typical operating conditions, are a requirement for HiRel applications on logic devices. System designers must spend a lot of time designing specific boards and developing test vectors for gate array designs that can be used to exercise their designs dynamically at elevated temperatures. Test vectors are needed for dynamic operations and to verify fault coverage after burn-in. Even after the vectors are developed, fault coverage for typical designs may be only about 70 percent. With a 70 percent fault coverage, two to five percent of devices in typical masked gate array designs are likely to be defective.

In general, field programmable logic devices have reduced the need to develop test vectors. These devices enable the semiconductor vendor to perform the tests prior to programming. However, most one-time programmable logic devices have not yet achieved the functional quality levels of other semiconductors because they don't allow the chip manufacturer to access and test all internal gates. In the past, one-time programmable devices had poor test coverage, and users often experienced functional failure rates of more than ten percent on parts that had passed programming. On-chip test circuits and testing techniques have greatly improved since then, and now one-time programmable devices have functional defect rates in the range of 0.1 to one percent.

This application note explains the difference between burn-in on a blank FPGA versus a programmed FPGA and describes the testability features of the FPGA from initial test to post programming. Specifically, this application note describes the testability of a blank Actel FPGA and explains why performing post programming burn-in on Actel FPGA's offers no benefit, and is therefore unnecessary.

Testability of Actel FPGAs

Although Actel's FPGAs use a one-time programmable technology, the unique device architecture permits a high degree of testability that is comparable to reprogrammable devices. Special test modes allow functional testing of unprogrammed devices at essentially 100 percent fault coverage, regardless of device density. This is possible because of the architecture and programming of Actel FPGAs.

Architecture

The logic module is the basic building block of all Actel FPGAs. Each logic module is programmable and capable of implementing all two-input logic functions, most three-input functions, and many other functions (up to eight inputs). With an architecture similar to a channeled gate array, logic modules are organized in rows and columns across the chip (Figure 1). Adjacent to each row of logic modules are routing channels that run vertically and horizontally across the chip. These routing channels are used to configure logic modules and connect inputs and outputs of logic modules together to implement a design. I/O buffers and test circuits surround the array of logic modules and routing channels.

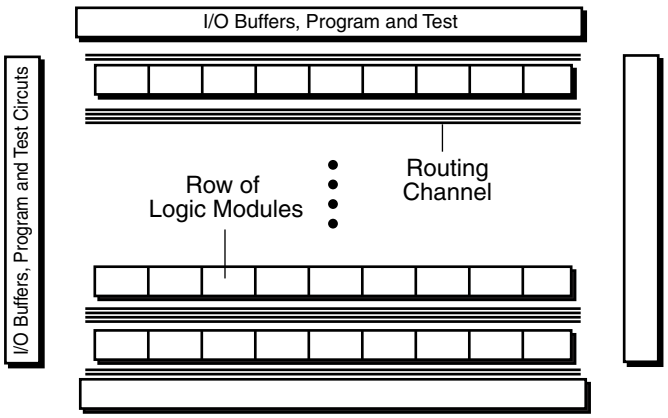

Figure 1 • Architecture

Programmable antifuse (PLICE™) elements are located within the routing channels. The antifuse is normally open and is programmed to form an electrical connection between routing elements. An antifuse that connects a horizontal routing track to a vertical track is called a cross antifuse. As shown in Figure 2, the output from Module 3 is connected to a horizontal routing track by a programmed cross antifuse. Another cross antifuse is programmed to connect an input to Module 4. In a similar manner, the output of Module 3 is connected to the input of Module 2. Some horizontal tracks are broken into segments to enable logic modules that are close to each other to be connected without using a full horizontal track that would waste routing resources and slow down circuit performance. Sometimes, however, it is necessary to connect two segments together to form a longer segment. This can be done by programming a horizontal antifuse. In Figure 2, the output of Module 3 is connected to the input of Module 1 by programming two cross antifuses and one horizontal antifuse. Vertical antifuses are used in the same way to connect two vertical segments.

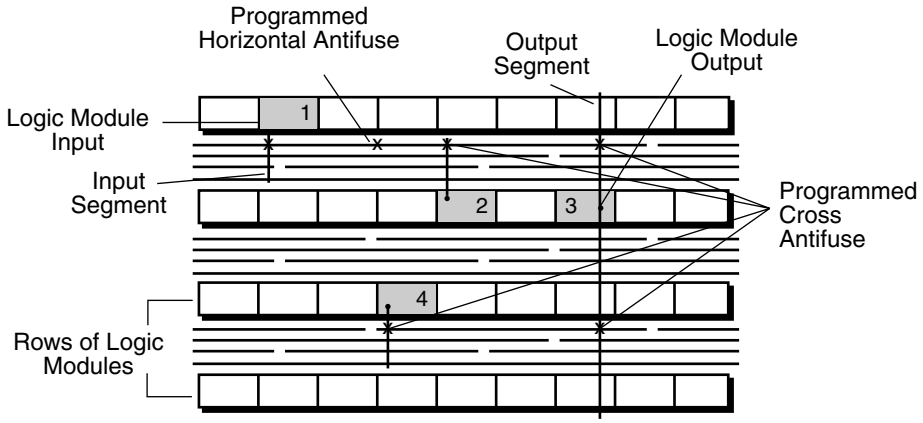

Figure 2 • *Routing*

A more detailed example of the Actel FPGA architecture is shown in Figure 3 on page 3. Six logic modules (two rows, three columns) are shown. Between the two rows are six horizontal tracks. Running down each column are five vertical tracks. Note that the products actually have 25 to 36 horizontal and 13 to 15 vertical tracks. The circles at the intersection of vertical and horizontal tracks represent cross antifuses. Circles lying only on the horizontal tracks are horizontal antifuses. Additionally, vertical and horizontal pass transistors can be activated to connect horizontal or vertical tracks even if an antifuse has not been programmed. This ability to connect tracks in unprogrammed devices is used extensively during antifuse programming and is one of the key elements responsible for the excellent testability of Actel FPGAs.

Logic configuration of modules is interesting because there are no dedicated antifuses in the module. Instead, the inputs (and outputs) of logic modules extend into the cross antifuse array. Each logic module has eight to ten inputs and one output. By programming the appropriate antifuses, an input can be connected to a dedicated horizontal ground line, a V_{CC} line, or a horizontal routing track. The logic module implements a particular logic function by tying appropriate unused inputs to ground or V_{CC}.

Test Modes of Actel FPGAs

The unique architecture described earlier allows outstanding testability of unprogrammed devices at the factory. The various test modes are as follow:

1. The shift register circling the periphery of the chip can be both downloaded and uploaded. This enables the use of various test patterns to ensure that the shift register is fully functional.
2. All vertical and horizontal tracks can be tested for continuity and shorts. There are several ways to implement these tests. One way of performing continuity testing is to precharge the array, turn on all vertical or horizontal pass transistors on a track, drive the track low from one side of the chip, and read a low on the other side. Shorts are detected by driving every other track low after precharge and reading back on the other side. These tests also confirm that the vertical and horizontal pass transistors will turn on.
3. For programming, it is important to make sure that all tracks can hold the precharge level. By charging a track, floating it, and waiting a predetermined amount of time, the track can be read back and confirmed to still be high.

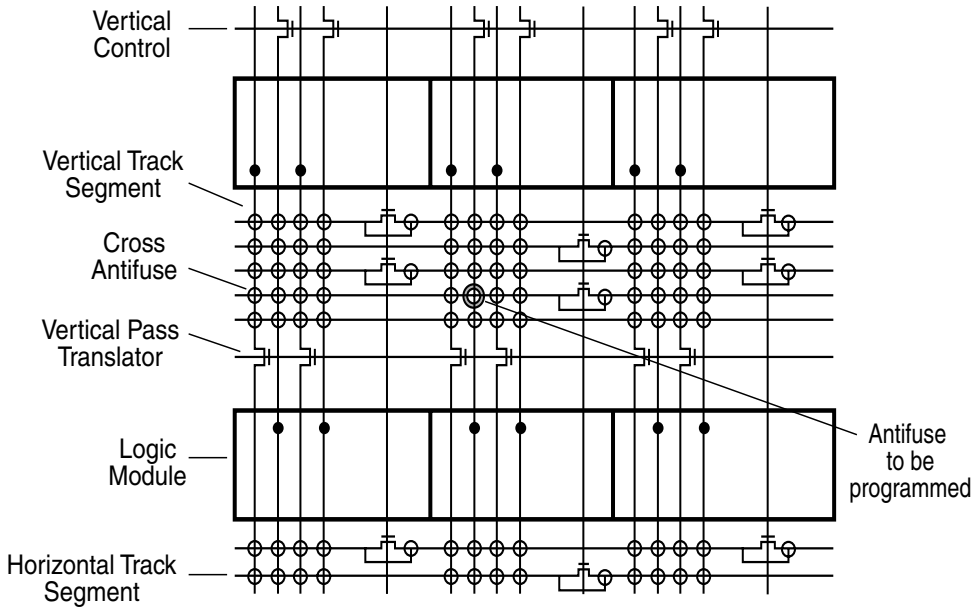

Figure 3 • *Programmable Interconnect*

4. Test for leakage of vertical and horizontal pass transistors by driving one side of a track to a voltage via the V_{PP} pin and grounding the other side. All pass transistors except the one being tested are turned on. If excess current is detected on the V_{PP} pin, the pass transistor is considered defective.
5. There are one or two dedicated clock buffers that travel across all horizontal channels. Test these buffers by are driving them with the clock pin and then reading them for the proper levels at the sides of the array.
6. Two pins, Probe A and Probe B, enable observation of internal signals. By entering a test mode, the shift register can be made to address the internal output of any logic module. This output is then directed to one of two dedicated vertical tracks, which in turn can be observed externally on the Probe A or B pins. This ability to observe internal signals (even on unprogrammed parts) allows Actel to perform a large number of functional tests. The first such test is the Input Buffer Test. Input buffers on all I/O pins can be tested for functionality by driving at the input pad and reading the internal I/O output node through the probe pins.
7. Test modes exist to drive all output buffers low, high, or to tri-state. This allows testing of V_{OL}, V_{OH}, I_{OL}, I_{OH}, and leakage on all I/Os.
8. One of the key tests is the ability to functionally test all internal logic modules. By activating various vertical pass transistors and driving from the top or bottom of the chip, any of the eight to ten module inputs can be forced to a high or low. The logic module test allows 100 percent fault coverage of each module. Additionally, test time is reduced because the architecture allows modules to be tested in parallel.
9. Actel FPGAs have one or two dedicated columns on the chip that are transparent to the user and used by the factory for speed selection. These columns are referred to as the binning circuit. Modules in the columns are connected to each other by programming antifuses. The speed of the completed test circuit can then be tested. The binning circuit allows the separation of units into different speed categories. It also allows the speed distribution within each category to be minimized. Additionally, functionality of the programming circuitry can be verified by programming the binning circuit.
10. To confirm that the programming circuitry is working, a basic junction stress/leakage test is performed. The program mode is enabled and V_{PP} voltage plus a guardband is applied to the V_{PP} pin. All vertical and horizontal tracks are driven to V_{PP}, so no voltage is applied across the antifuses. The I_{PP} current is then measured. If it exceeds its normal value, the device is rejected.
11. The Antifuse Shorts Test (or Blank Test) ensures that none of the antifuses are programmed. In this test, the array is precharged and then the vertical tracks are driven to ground. The horizontal tracks are then read to confirm that they are still high (a programmed or leaky antifuse would drive a horizontal track low). The test is repeated by driving horizontal tracks low and reading vertical tracks.
12. Actel FPGAs also have a Silicon Signature™. In the ACT 1 family, the Silicon Signature consists of four words

of data. The first word is hard wired (no antifuses) and contains a manufacturer ID number as well as a device ID number. A programmer can read these numbers and will automatically select the proper programming algorithm. The other code-words contain antifuses and are programmable by the designer. Actel is currently using bits in these words to store information such as the chip's run number and wafer number. Thus, each Actel FPGA has traceability down to the wafer level. The functionality of the programming circuitry is tested by programming this information. Actel software also allows the user to program a design ID and checksum into the Silicon Signature. By reading this back later, the user can verify that the chip is correctly programmed to a given design.

13. The stress test is one of the most important antifuse tests. When this test is enabled, a voltage applied to the V_{PP} pin can be applied across all antifuses on the chip (the other side of the antifuse is grounded). The voltage applied is the precharge voltage plus a significant guardband. After the voltage is applied, the Antifuse Shorts Test is used again to ensure no antifuses have been programmed. The antifuse stress test is effective in catching antifuse defects. Because the reliability of the antifuse is much more voltage than temperature dependent, this test is also an effective antifuse infant mortality screen.

Burn-In of Actel FPGAs

Since antifuse infant mortality failures are effectively screened out during electrical testing, it is unnecessary to do any kind of burn-in to screen out such failures in standard commercial production units. However, burn-in is still an effective screen for standard CMOS infant mortality failure mechanisms, and it is required for all military 883D products. MIL-883D Method 1005 allows several types of burn-in screens, which can be divided into two categories: steady-state (static) and dynamic. Static burn-in applies DC voltage levels to the pins of the device under test. The device may or may not be powered up. Dynamic burn-in applies AC signals to device inputs. These signals are selected so that the device receives internal and external stresses similar to those it may experience in a typical application.

Static burn-in is easier to implement. By choosing appropriate biasing conditions and load resistors, it is possible to design a single burn-in circuit for both unprogrammed and programmed devices. The pattern programmed into the device does not matter. Static burn-in can be an effective screen for some types of failure modes, particularly those that may happen at device inputs or outputs (such as screening for mobile ionic contamination). However, it is not very effective at stressing internal device circuits. Many internal nodes may be biased at ground without receiving any voltage or current stress. Signal lines will not toggle, and it may not be possible to screen failure modes such as metal electromigration.

A properly designed dynamic burn-in can effectively stress inputs, outputs, and internal circuits. However, dynamic burn-in of ASICs can be very expensive because customer-specific burn-in circuits and burn-in boards must be designed and built to properly stress each design implemented in the ASIC. This results in large NRE costs and long lead times. From the standpoint of burn-in, a programmed FPGA is essentially the same as a mask-programmed ASIC, and it requires similar custom burn-in circuits to do a dynamic burn-in. However, Actel uses the testability features of its FPGA products to allow effective dynamic burn-in of unprogrammed devices. This dynamic burn-in enables the circuits to be stressed in a way that static burn-in is unable to duplicate.

During dynamic burn-in of unprogrammed units, test commands are serially shifted into each device using the SDI pin and clocked using the DCLK pin. There are three test modes shifted into each device. The first test stresses each cross antifuse with a voltage of V_{PP} – 2V (V_{PP} is normally set at 7.5V so that each antifuse gets 5.5V across it). This voltage is applied to all vertical tracks while the horizontal tracks are grounded. Once enabled, the stress mode is held for 10 ms.

The second test mode is identical to the first except that the horizontal tracks are driven to V_{PP} – 2V while the vertical tracks are grounded. Note that both of these modes are similar to the antifuse stress test described earlier, but the stress voltage is lower during burn-in.

During the first and second test mode, not only do the tests stress the antifuses, but they also toggle all routing tracks in the chip to V_{PP} – 2V and ground. All input and output tracks to the logic modules are also toggled, stressing the internal devices of every logic module. Unlike a typical programmed design where a percentage of logic modules and antifuses are stressed, in the blank dynamic burn-in 100 percent of all logic modules and cross antifuses are stressed and tested.

The fourth test drives several I/O pins on the chip to a low state. Prior to this, they are at a high impedance state and held at V_{CC} through pull-up resistors. This test confirms that the burn-in is being properly implemented by looking at these I/O pins to see if they display the proper waveform. It also passes current through each I/O as it toggles low.

Although the chip is unprogrammed, these tests allow us to apply stresses to the inputs, outputs, and internal nodes that are similar to what a programmed device may experience in normal operation. Once burn-in is completed, post burn-in testing as specified by MIL-883D, including Percent Defect Allowed (PDA), is performed to ensure that fully compliant devices are shipped to the customer.

Programming

The following discussions about programming and testing modes are specific to the ACT 1 family of FPGAs. However, basic concepts also apply to the other Actel device families.

An antifuse is programmed by applying a sufficiently high voltage across it. This voltage is referred to as V_{PP}. To access an antifuse deep inside the chip, it is necessary to create electrical paths from V_{PP} and ground to the antifuse. This is done by turning on the appropriate horizontal and vertical pass transistors (in normal chip operation, these transistors are always off). The transistors are turned on by applying V_{PP} to their gates. Figure 4 shows an example of how a typical cross antifuse can be programmed. V_{PP} is applied to a vertical track at the top of the chip and ground is applied to a horizontal track on the right side. The design of the A1010B/A1020B allows V_{PP} or ground to be applied from the top, bottom, left, or right as is appropriate to best access a particular antifuse. Notice that V_{PP} is also applied to the gates of the horizontal and vertical pass transistors on the tracks accessing the cross antifuse. The circled cross antifuse now has V_{PP} applied to it on one side and ground on the other. This voltage breaks down the antifuse's dielectric and creates an electrical connection between the horizontal and vertical routing tracks.

There is another important consideration when programming an antifuse. Notice that the cross antifuses in the same vertical track as the antifuse to be programmed also have V_{PP} applied to them on one side. This is true until the track is broken by a vertical pass transistor below it that is turned off. However, the potential on the other side of the antifuses are not being driven. Should this potential be at ground, the other cross antifuses on the vertical segment could accidentally be programmed.

The same logic applies to other antifuses on the same horizontal track. Here, one side of the antifuse is being driven to ground, but if the other side is at V_{PP}, extra antifuses could be programmed. This problem is solved by first applying a "precharge cycle". During the precharge cycle, all horizontal and vertical tracks are charged to $V_{PP}/2$. As a result, there is no voltage across the antifuses. The appropriate vertical track is then driven to V_{PP} and a horizontal track is driven to ground (Figure 5 on page 5). At this point, other antifuses on the vertical track have a potential of $V_{PP}/2$ across them (V_{PP} on one side and $V_{PP}/2$ on the other). This $V_{PP}/2$ voltage is not sufficient to program the antifuses. Other antifuses on the same horizontal track also have $V_{PP}/2$ across them ($V_{PP}/2$ on one side and ground on the other). Most other antifuses in the chip still have $V_{PP}/2$ on both sides and will not be programmed accidentally.

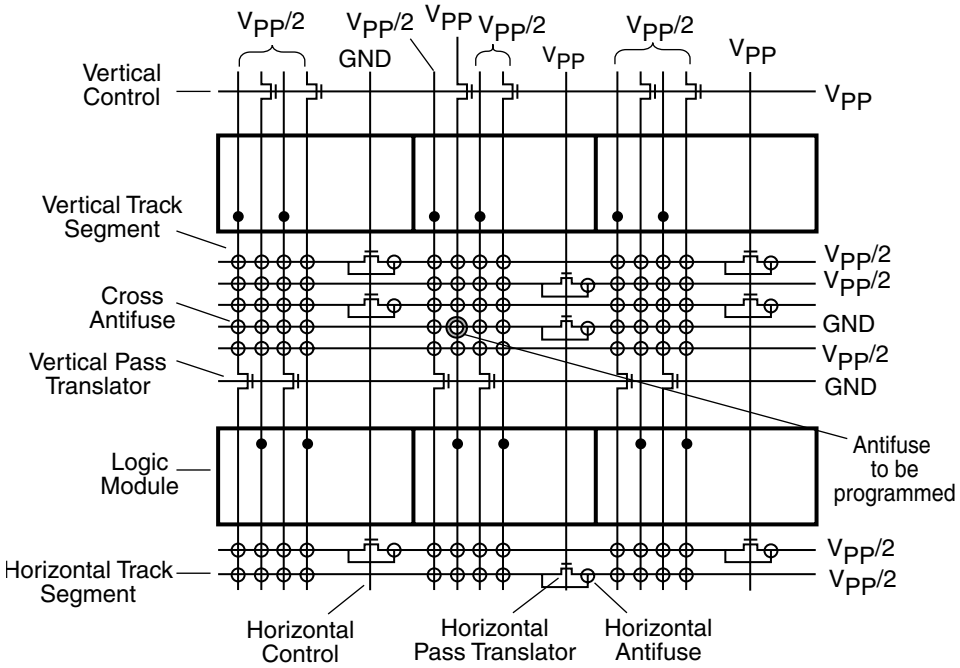

Figure 5 • *Programming of a Cross Antifuse (Precharge Cycle)*

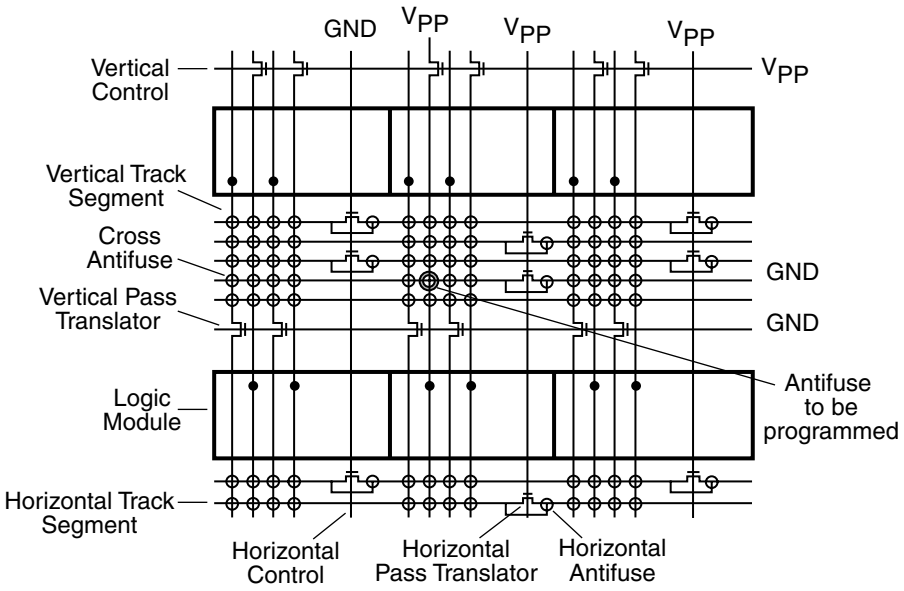

Figure 4 • *Programming of a Cross Antifuse*

Programming Algorithm

In concept, the Actel FPGAs are programmed in a manner very similar to many other programmable logic devices like memories such as EPROMs. The programming algorithm consists of the following steps:

1. An addressing sequence to select the antifuse to be programmed.
2. A programming sequence where V_{PP} is applied in pulses until the antifuse programs.
3. A soak or "overprogram" step to ensure uniform, low antifuse resistance.
4. A step to verify that the antifuse has been properly programmed.Unlike a memory where an antifuse is addressed by applying a parallel address, the FPGAs are addressed in a serial manner by using the special DCLK (Data Clock) and SDI (serial data in) pins. There is a large shift register that travels around the periphery of the chip. Bits in this shift register can be used to drive tracks to ground, V_{CC}, V_{PP}, or float. It is also possible to sense the level on the track (high or low) and load this information into the shift register. By shifting in the correct address, any antifuse can be selected for programming. The shift register also plays a key role in testing the chip, as mentioned in Step 1 of the "Test Modes of Actel FPGAs" section on page 2.

The programming sequence starts with the precharge pulse where $V_{PP}/2$ is applied to the V_{PP} pin. This is followed by a programming pulse where V_{PP} is applied to the pin. Following the program pulse, the voltage on the V_{PP} pin is returned to a nominal value (about 6V). See Figure 6 for a typical V_{PP} waveform. The precharge/program pulse sequence is repeated until either the selected antifuse is programmed or a maximum number of pulses is exceeded (in which case the antifuse is considered nonprogrammable and the device is rejected).

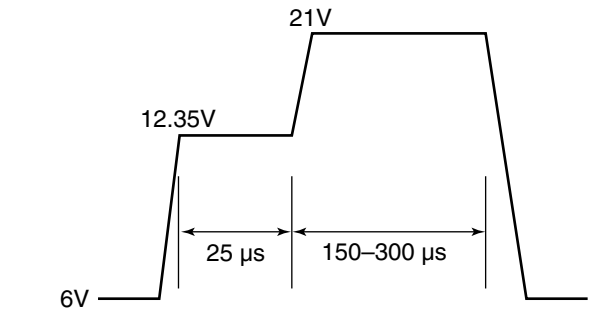

Figure 6 • *V_{PP} Waveform*

Confirmation that an antifuse has been programmed is determined by monitoring the current on the V_{PP} pin. This current is very low (typically < 10 µa) until the antifuse is programmed. After the antifuse is programmed, an electrical connection is made between V_{PP} and ground and currents ranging from 3 to 15 mA may be observed on V_{PP}. Once this current is observed, the antifuse is considered programmed and enters the soak or "overprogram" cycle. Here, extra pulses are applied to the antifuse to achieve minimum antifuse resistance. All devices in the FPGA exposed to >V_{CC} during programming are high voltage devices designed for >18V. All high voltage pass devices and

programming devices are turned off during normal operation. Any failures due to high voltage devices in the programming path will either be detected by the programmer as an increase in current or screened out at test prior to programming.

There are other tests implemented in the programmer which are transparent to the user that are used to verify programming and ensure functionality. These tests are designed to verify that no antifuses were accidentally programmed and that there are no input to output shorts. In addition, an I_{CC} delta measurement is performed prior to and following programming to ensure no damage occurred during programming. The only change in the device after programming is the rupture of the antifuse. All low voltage devices used during normal operation within the logic modules are isolated during programming

Conclusion

The architecture and numerous test modes of Actel's FPGAs enable a high degree of testability of these devices. All internal logic gates can be tested without programming antifuses other than the few for the Binning Circuit and Silicon Signature. Because Actel FPGAs are one-time programmable, the only item that is not fully tested at the factory is the programmability of all the individual antifuses. However, this is done on the programmer while the units are being programmed. Actel achieves functional yields equal to reprogrammable parts and greater than other one-time programmable devices because of its ability to test all the internal gates in its devices. In addition, due to the testability and architecture of the antifuse in the blank state, Actel is able to perform static and dynamic burn-in on blank devices, eliminating the need to perform burn-in on programmed devices.

http://www.actel.com

Actel Europe Ltd.
Daneshill House, Lutyens Close
Basingstoke, Hampshire RG24 8AG
United Kingdom
Tel: +44-(0)125-630-5600
Fax: +44-(0)125-635-5420

Actel Corporation
955 East Arques Avenue
Sunnyvale, California 94086
USA
Tel: (408) 739-1010
Fax: (408) 739-1540

Actel Asia-Pacific
EXOS Ebisu Bldg. 4F
1-24-14 Ebisu Shibuya-ku
Tokyo 150 Japan
Tel: +81-(0)3-3445-7671
Fax: +81-(0)3-3445-7668

5192662-0/4.00