

Flash*Freeze Control Using an Internal Oscillator

Introduction

The Actel IGLOO[®] and ProASIC[®]3L families of FPGA devices are based on Actel nonvolatile flash technology and single-chip ProASIC3L FPGA architecture. These devices are part of a 1.2 V to 1.5 V operating voltage, offering the industry's lowest power consumption, smallest footprint, competitive prices, and many advanced features. The Flash*Freeze technology used in IGLOO and ProASIC3L devices enables entering and exiting Low Power mode, which consumes as little as 2 μ W of power while retaining all SRAM and register data. Flash*Freeze technology simplifies power management through input/output (I/O) and clock management, with rapid recovery to full operational mode. Refer to the [IGLOO FPGA Fabric User's Guide](#) and the [ProASIC3L FPGA Fabric User's Guide](#) for additional information on the device features and use of the Flash*Freeze pin.

In applications where the device is in Flash*Freeze mode for most of the operation or in cases such as remote deployment with the Flash*Freeze pin asserted, there is a need to control the Flash*Freeze entry and exit from within the device to save power, simplify software, and avoid continuous toggling of the Flash*Freeze pin.

In addition, IGLOO and ProASIC3L devices cannot be programmed while in Flash*Freeze mode and the logic that enables authentication for encrypted programming is turned off by default to save power. Therefore, a solution is needed to bring the device out of Flash*Freeze mode and to internally turn on the authentication circuit in applications where physical access to the Flash*Freeze pin is not available. This logic, which is programmed into the device along with the user application, acts as a Master for the Flash*Freeze mode. Two distinct use models have been developed to address this type of application. You can choose either one of these solutions, depending on the individual application requirements:

- Flash*Freeze control with JTAG: This solution provides on-demand wake-up of the device for programming and also for authentication.
- Flash*Freeze control using an internal oscillator

This application note describes the internal oscillator for Flash*Freeze control for all IGLOO and ProASIC3L devices and addresses the following requirements:

- Internal clock source for logic (counter) to control entry to and exit from Flash*Freeze mode, a ring oscillator
- Modification required for the Flash*Freeze management IP core

Flash*Freeze control with JTAG is described in [Flash*Freeze Control Using JTAG](#).

Internal Oscillator for Flash*Freeze Control

The functionality is accomplished using a macro which consists of the following elements:

- Ring oscillator that acts as clock source
- Counter used as a Master for Flash*Freeze mode
- User low static ICC macro, ULSICC
- Flash*Freeze Management (FFM) IP core with a minor modification, requiring the following:
 - 2-input AND gate
 - 2-input OR gate with one input inverted

Figure 1 shows the block diagram for Flash*Freeze Master mode with the internal ring oscillator.

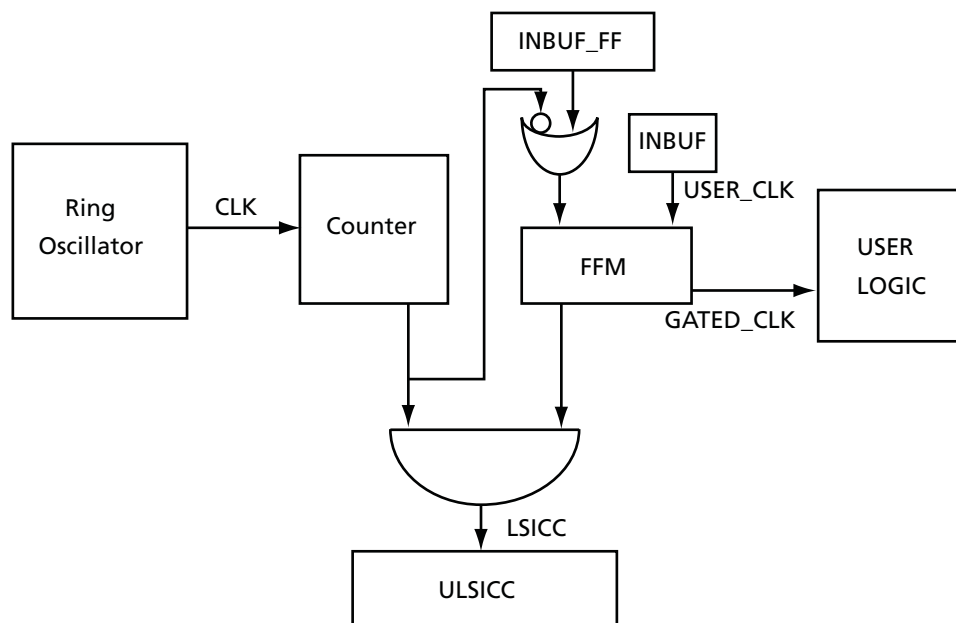


Figure 1 • Internal Ring Oscillator Block Diagram

Ring Oscillator

When an IGLOO or ProASIC3L device is in Flash*Freeze mode, none of the input signals (external clock) are seen by the internal logic. The ring oscillator is the internal clock source to the design, which must provide a reasonably reliable clock with minimum power consumption.

The ring oscillator consists of an inverter loop with an odd number of inverters used to create a clock. One of the inverters is changed to NAND2, with one input tied to RESET, to prevent harmonics on the clock, which could result from uninitialized power-up values. The generated clock frequency is divided down using toggle flip-flops (T-FF) in series to form a clock divider chain.

Use the flow below to arrive at the proper trade-off between power consumption, predictable delay, and optimal utilization:

1. The three-input cell A014 is configured as an inverter by tying off inputs A and B to GND, as shown in the Verilog example below. This cell has the longest pin-to-pin delay among all cells in the library. You can use a chain of inverters with one NAND2 gate, as long as the total number of cells in the chain, including NAND2, are odd.

```

module basic_macro( input a, output y);
A014 INV_DLY(.A(GND), .B(GND), .C(a), .Y(y)) /* synthesis syn_noprune = 1*/ ;
endmodule

```

2. T-FF is constructed with DFI1C0 to form the clock divider chain:

```

module tffr( inout t, input clk, input clr);
DFI1C0 T_FF(.D(t), .CLK(clk), .CLR(clr), .QN(t)) /* synthesis syn_noprune = 1*/ ;
endmodule

```

3. To prevent Designer from optimizing the inverter chain by converting A014 to a regular inverter, and to preserve the number of inverters in the chain, use this compile directive:
COB_D0_COB = 0.
4. The placement forces the use of local routing resources between inverters in the chain and T-FFs.

Figure 2 shows the ring oscillator inverter loop placement.



Figure 2 • Ring Oscillator Inverter Loop Placement

Figure 3 shows the ring oscillator T-FF chain placement.

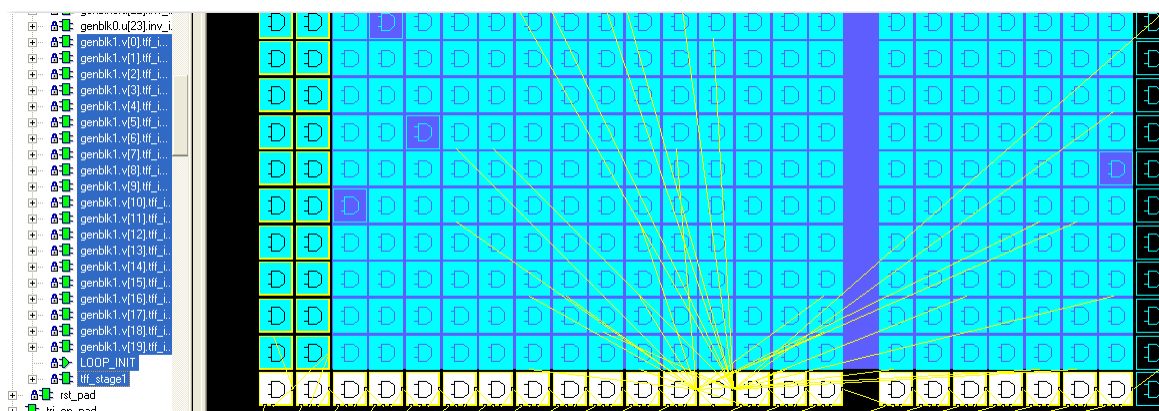


Figure 3 • Ring Oscillator T-FF Chain Placement

The number of inverters and T-FFs can be changed to achieve the desired ring-oscillator frequency. The goal is to have a clock frequency in the KHz range that is useful yet optimal in power consumption. The majority of the logic must be toggling in the KHz range to achieve this.

Note: A configuration using 25 inverters (one NAND2 and 24 inverters) and 21 T-FFs was tested on an Actel IGLOO AGL600V2 device using a 1.2 V power supply. The resulting current consumption was 225 μ A.

ULSICC Macro

This user low static ICC macro has one LSICC input that controls entry to and exit from Flash*Freeze mode. This works in conjunction with the Flash*Freeze input pin.

Counter

This is a simple binary up-counter that is clocked by the ring counter. The counter width is dependent on the application requirements—how often the device needs to transition into and out of Flash*Freeze mode. This counter acts as the Master for Flash*Freeze mode.

Flash*Freeze Management IP Core

The Flash*Freeze Management IP core is designed to manage entry to and exit from Flash*Freeze mode, as directed by the external Flash*Freeze input pin only. The Flash*Freeze Management IP core has the INBUF_FF and ULSICC Actel library cells embedded in the core. The core needs to be modified to respond to the internal Flash*Freeze control, so INBUF_FF and ULSICC are brought out and their inputs modified to include the counter bit, which acts as the internal Flash*Freeze control.

Use one of the MSB bits of the counter as an input to the 2-input AND gate that generates the LSICC input to the ULSICC macro. The second input to the AND gate is the Flash_Freeze_Enabled output from the Flash*Freeze Management IP core. This controls entry into Flash*Freeze mode.

The same MSB bit is connected to the bubbled input of the 2-input OR gate. The second input to the OR gate is the Flash*Freeze input coming from the dedicated Flash*Freeze pad through INBUF_FF I/O buffer. The output of this OR gate becomes the effective Flash*Freeze input into the IP. This controls the exit from Flash*Freeze mode.

Refer to the *Flash*Freeze Technology and Low-Power Modes* section of the IGLOO or ProASIC3L handbooks for additional information on the device features and use of the Flash*Freeze pin.

Theory of Operation

The ring oscillator provides a free running clock used as the clock for the counter. Depending on the application requirements, any of the ring oscillator clock divider outputs can be used, along with an appropriate counter bit, to arrive at the required time delay for entry to and exit from Flash*Freeze mode.

The device enters Flash*Freeze mode with the external Flash*Freeze pin asserted and when the selected count is reached. The device remains in Flash*Freeze mode as long as this bit remains High. Once that bit goes Low, the device exits Flash*Freeze mode. During this period, user logic is up and running, performs any scheduled tasks, and then returns to Flash*Freeze mode.

Conclusion

This application demonstrates the usage of the internal oscillator for Flash*Freeze control without an external clock or physically toggling the Flash*Freeze pin. While the application note shows a counter used as a Master running off the internal clock, more involved logic such as a finite state machine (FSM) that takes into account multiple events in a design can easily be the Master running off the internal clock.

Related Documents

IGLOO FPGA Fabric User's Guide

www.actel.com/documents/IGLOO_UG.pdf.

ProASIC3L FPGA Fabric User's Guide

www.actel.com/documents/PA3L_UG.pdf

*Flash*Freeze control with JTAG*

www.actel.com/documents/FlashFreeze_Ctrl_using_JTAG_AN.pdf.

Actel and the Actel logo are registered trademarks of Actel Corporation.
All other trademarks are the property of their owners.



Actel is the leader in low power and mixed signal FPGAs and offers the most comprehensive portfolio of system and power management solutions. Power Matters. Learn more at www.actel.com.

Actel Corporation

2061 Stierlin Court
Mountain View, CA
94043-4655
USA

Phone 650.318.4200
Fax 650.318.4600

Actel Europe Ltd.

River Court, Meadows Business Park
Station Approach, Blackwater
Camberley Surrey GU17 9AB
United Kingdom

Phone +44 (0) 1276 609 300
Fax +44 (0) 1276 607 540

Actel Japan

EXOS Ebisu Building 4F
1-24-14 Ebisu Shibuya-ku
Tokyo 150 Japan

Phone +81.03.3445.7671
Fax +81.03.3445.7668
<http://jp.actel.com>

Actel Hong Kong

Room 2107, China Resources Building
26 Harbour Road
Wanchai, Hong Kong

Phone +852 2185 6460
Fax +852 2185 6488
www.actel.com.cn