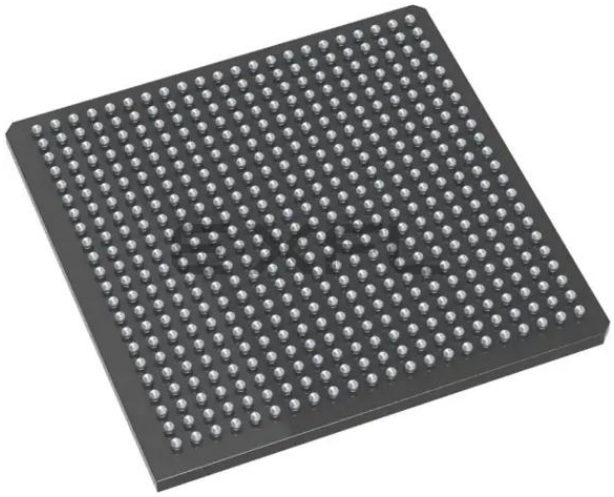

Welcome to E-XFL.COM

Understanding Embedded - FPGAs (Field Programmable Gate Array)

Embedded - FPGAs, or Field Programmable Gate Arrays, are advanced integrated circuits that offer unparalleled flexibility and performance for digital systems. Unlike traditional fixed-function logic devices, FPGAs can be programmed and reprogrammed to execute a wide array of logical operations, enabling customized functionality tailored to specific applications. This reprogrammability allows developers to iterate designs quickly and implement complex functions without the need for custom hardware.

Applications of Embedded - FPGAs

The versatility of Embedded - FPGAs makes them indispensable in numerous fields. In telecommunications.

Details	
Product Status	Active
Number of LABs/CLBs	-
Number of Logic Elements/Cells	-
Total RAM Bits	110592
Number of I/O	235
Number of Gates	600000
Voltage - Supply	1.14V ~ 1.575V
Mounting Type	Surface Mount
Operating Temperature	-40°C ~ 100°C (TJ)
Package / Case	484-BGA
Supplier Device Package	484-FPBGA (23x23)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/a3p600l-1fg484i

Clock Gating Block

Once DONE_HOUSEKEEPING is detected, the FSM will initiate the clock gating circuit by asserting ASSERT_GATE (active Low). ASSERT_GATE is named control_user_clock_net in the IP block. Upon assertion of the ASSERT_GATE signal, the clock will be gated in less than two cycles. The clock gating circuit is comprised of a flip-flop, latch, AND gate, and CLKINT, as shown in Figure 2-12. The clock gating block can support gating of up to 17 clocks.

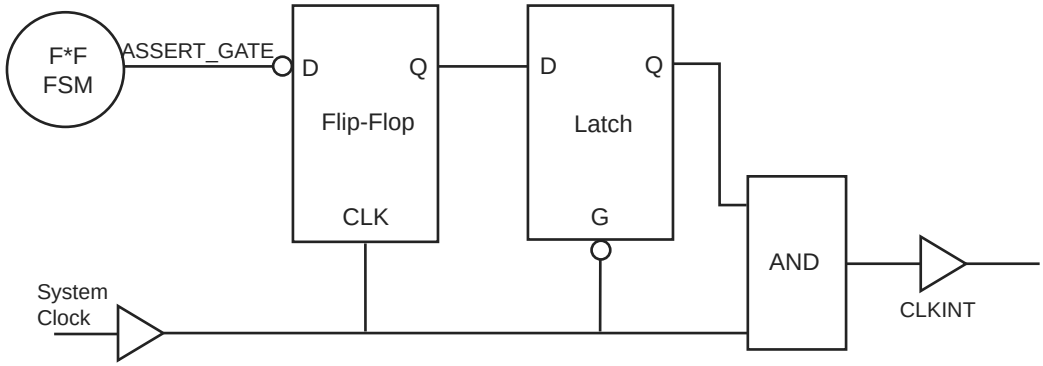

Figure 2-12 • Clock Gating Circuit

After initiating the clock gating circuit, the FSM will assert and hold the LSICC signal (active High), feeding the ULSICC macro. This will initiate the 1 µs entrance into Flash*Freeze mode.

Upon deassertion of the Flash*Freeze pin, the FSM will set ASSERT_GATE High. Once the I/O banks become active, the clock will enter the device and register the ASSERT_GATE signal, cleanly releasing the clock gate.

Design Flow [1]

Microsemi has developed a convenient and intuitive design flow for configuring and integrating Flash*Freeze technology into an FPGA design. Flash*Freeze type 1 is implemented by instantiating the INBUF_FF macro in the top level of a design. Flash*Freeze type 2 with management IP can be generated by the Libero core generator or SmartGen and instantiated as a single block in the user's design. This single block will include an INBUF_FF macro and the optional Flash*Freeze management IP, which includes the ULSICC macro. If designers do not wish to use this core generator, the INBUF_FF macro and the optional ULSICC macro may be instantiated in the design, and custom Flash*Freeze management IP can be developed by the user. The remainder of this section will cover configuration details of the INBUF_FF macro, the ULSICC macro, and the Flash*Freeze management IP.

Additional information on the tools discussed within this section may be found in the Libero online help.

INBUF_FF

The INBUF_FF macro is a special-purpose input buffer macro that is interpreted downstream in the design flow by Microsemi's Designer software. When this macro is used, the top-level port will be forced to the dedicated FF pin in the FPGA, and Flash*Freeze mode will be available for use in the device. The following are the design rules for INBUF_FF:

If INBUF_FF is not used in the design, the device will not be configured to support Flash*Freeze mode.

When the INBUF_FF macro is used, the FF pin will establish a hardwired connection to the Flash*Freeze technology circuit in the device, as shown in Figure 2-1 on page 25, Figure 2-3 on page 27, and Figure 2-10 on page 37, and described in the "Flash*Freeze Type 1: Control by Dedicated Flash*Freeze Pin" section on page 24.

1. This section applies to Libero / Designer software v8.3 and later. Microsemi recommends that designs created in earlier versions of the software be modified to accommodate this flow by instantiating the INBUF_FF macro or the Flash*Freeze management IP. Refer to the Libero / Designer software v8.3 release notes and the Libero online help for more information on migrating designs from older software versions.

Step 1

Run Synthesis with default options. The Synplicity log shows the following device utilization:

Cell usage:

	cell count	area	count*area
DFN1E1C1	1536	2.0	3072.0
BUFF	278	1.0	278.0
INBUF	10	0.0	0.0
VCC	9	0.0	0.0
GND	9	0.0	0.0
OUTBUF	6	0.0	0.0
CLKBUF	3	0.0	0.0
PLL	2	0.0	0.0
TOTAL	1853		3350.0

Step 2

Run Compile with the Promote regular nets whose fanout is greater than option selected in Designer; you will see the following in the Compile report:

```
Device utilization report:
==========================
CORE                    Used:     1536   Total:    13824     (11.11%)
IO (W/ clocks)      Used:       19   Total:      147     (12.93%)
Differential IO     Used:        0   Total:       65     (0.00%)
GLOBAL                  Used:        8   Total:       18     (44.44%)
PLL                     Used:     2   Total:        2    (100.00%)
RAM/FIFO                Used:     0   Total:       24     (0.00%)
FlashROM                Used:     0   Total:        1     (0.00%)

The following nets have been assigned to a global resource:
Fanout  Type         Name
----------------------------
1536    INT_NET            Net   : EN_ALL_c
                               Driver: EN_ALL_pad_CLKINT
                               Source: AUTO PROMOTED
1536    SET/RESET_NET  Net   : ACLR_c
                               Driver: ACLR_pad_CLKINT
                               Source: AUTO PROMOTED
256     CLK_NET            Net   : QCLK1_c
                               Driver: QCLK1_pad_CLKINT
                               Source: AUTO PROMOTED
256     CLK_NET            Net   : QCLK2_c
                               Driver: QCLK2_pad_CLKINT
                               Source: AUTO PROMOTED
256     CLK_NET            Net   : QCLK3_c
                               Driver: QCLK3_pad_CLKINT
                               Source: AUTO PROMOTED
256     CLK_NET            Net   : $1N14
                               Driver: $1I5/Core
                               Source: ESSENTIAL
256     CLK_NET            Net   : $1N12
                               Driver: $1I6/Core
                               Source: ESSENTIAL
256     CLK_NET            Net   : $1N10
                               Driver: $1I6/Core
                               Source: ESSENTIAL
```

Designer will promote five more signals to global due to high fanout. There are eight signals assigned to global networks.

During Layout, Designer will assign two of the signals to quadrant global locations.

Step 3 (optional)

You can also assign the QCLK1_c and QCLK2_c nets to quadrant regions using the following PDC commands:

```
assign_local_clock  net QCLK1_c   type quadrant UL
assign_local_clock  net QCLK2_c   type quadrant LL
```

Step 4

Import this PDC with the netlist and run Compile again. You will see the following in the Compile report:

```
The following nets have been assigned to a global resource:
Fanout  Type         Name
---------------------------
1536    INT_NET      Net   : EN_ALL_c
                             Driver: EN_ALL_pad_CLKINT
                             Source: AUTO PROMOTED
1536    SET/RESET_NET Net  : ACLR_c
                             Driver: ACLR_pad_CLKINT
                             Source: AUTO PROMOTED
256     CLK_NET      Net   : QCLK3_c
                             Driver: QCLK3_pad_CLKINT
                             Source: AUTO PROMOTED
256     CLK_NET      Net   : $1N14
                             Driver: $1I5/Core
                             Source: ESSENTIAL
256     CLK_NET      Net   : $1N12
                             Driver: $1I6/Core
                             Source: ESSENTIAL
256     CLK_NET      Net   : $1N10
                             Driver: $1I6/Core
                             Source: ESSENTIAL
The following nets have been assigned to a quadrant clock resource using PDC:
Fanout  Type         Name
---------------------------
256     CLK_NET      Net   : QCLK1_c
                             Driver: QCLK1_pad_CLKINT
                             Region: quadrant_UL
256     CLK_NET      Net   : QCLK2_c
                             Driver: QCLK2_pad_CLKINT
                             Region: quadrant_LL
```

Step 5

Run Layout.

Global Management in PLL Design

This section describes the legal global network connections to PLLs in the low power flash devices. For detailed information on using PLLs, refer to "Clock Conditioning Circuits in Low Power Flash Devices and Mixed Signal FPGAs" section on page 77. Microsemi recommends that you use the dedicated global pins to directly drive the reference clock input of the associated PLL for reduced propagation delays and clock distortion. However, low power flash devices offer the flexibility to connect other signals to reference clock inputs. Each PLL is associated with three global networks (Figure 3-5 on page 52). There are some limitations, such as when trying to use the global and PLL at the same time:

- If you use a PLL with only primary output, you can still use the remaining two free global networks.
- If you use three globals associated with a PLL location, you cannot use the PLL on that location.
- If the YB or YC output is used standalone, it will occupy one global, even though this signal does not go to the global network.

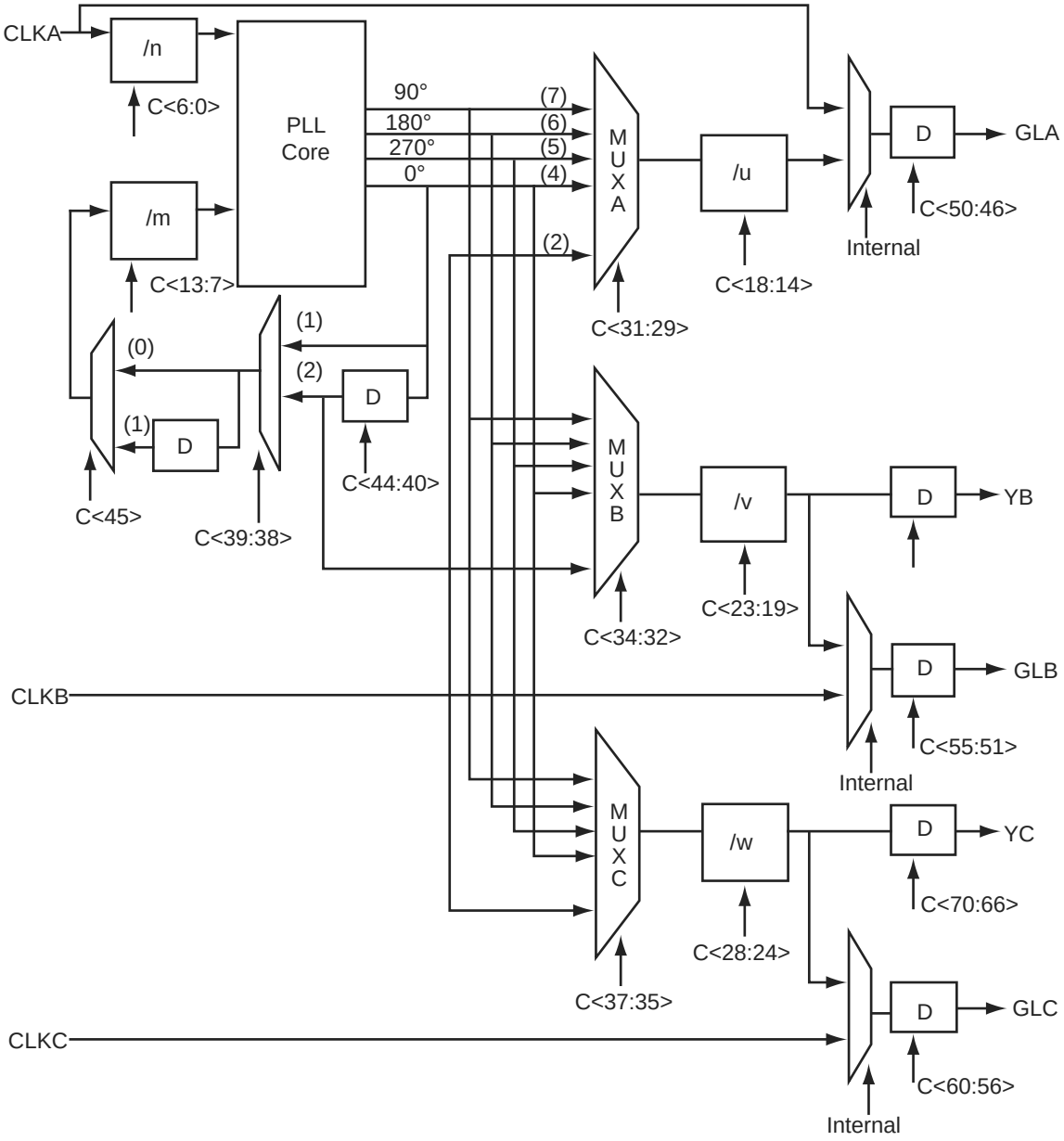

Figure 4-22 • CCC Block Control Bits Graphical Representation of Assignments

Use quadrant global region assignments by finding the clock net associated with the CCC macro under the Nets tab and creating a quadrant global region for the net, as shown in Figure 4-33.

Figure 4-33 • Quadrant Clock Assignment for a Global Net

External I/O Driven CCCs

The above-mentioned recommendation for proper layout techniques will ensure the correct assignment. It is possible that, especially with External I/O Driven CCC macros, placement of the CCC macro in a desired location may not be achieved. For example, assigning an input port of an External I/O Driven CCC near a particular CCC location does not guarantee global assignments to the desired location. This is because the clock inputs of External I/O Driven CCCs can be assigned to any I/O location; therefore, it is possible that the CCC connected to the clock input will be routed to a location other than the one closest to the I/O location, depending on resource availability and placement constraints.

Clock Placer

The clock placer is a placement engine for low power flash devices that places global signals on the chip global and quadrant global networks. Based on the clock assignment constraints for the chip global and quadrant global clocks, it will try to satisfy all constraints, as well as creating quadrant clock regions when necessary. If the clock placer fails to create the quadrant clock regions for the global signals, it will report an error and stop Layout.

The user must ensure that the constraints set to promote clock signals to quadrant global networks are valid.

Cascading CCCs

The CCCs in low power flash devices can be cascaded. Cascading CCCs can help achieve more accurate PLL output frequency results than those achievable with a single CCC. In addition, this technique is useful when the user application requires the output clock of the PLL to be a multiple of the reference clock by an integer greater than the maximum feedback divider value of the PLL (divide by 128) to achieve the desired frequency.

For example, the user application may require a 280 MHz output clock using a 2 MHz input reference clock, as shown in Figure 4-34 on page 126.

User I/O Naming Convention

IGLOO and ProASIC3

Due to the comprehensive and flexible nature of IGLOO and ProASIC3 device user I/Os, a naming scheme is used to show the details of each I/O (Figure 7-19 on page 207 and Figure 7-20 on page 207). The name identifies to which I/O bank it belongs, as well as pairing and pin polarity for differential I/Os.

I/O Nomenclature = FF/Gmn/IOuxwBy

Gmn is only used for I/Os that also have CCC access i.e., global pins.

FF = Indicates the I/O dedicated for the Flash*Freeze mode activation pin in IGLOO and ProASIC3L devices only

G = Global

m = Global pin location associated with each CCC on the device: A (northwest corner), B (northeast corner), C (east middle), D (southeast corner), E (southwest corner), and F (west middle)

n = Global input MUX and pin number of the associated Global location m either A0, A1, A2, B0, B1, B2, C0, C1, or C2. Refer to the "Global Resources in Low Power Flash Devices" section on page 47 for information about the three input pins per clock source MUX at CCC location m.

u = I/O pair number in the bank, starting at 00 from the northwest I/O bank and proceeding in a clockwise direction

x = P or U (Positive), N or V (Negative) for differential pairs, or R (Regular single-ended) for the I/Os that support single-ended and voltage-referenced I/O standards only. U (Positive) or V (Negative) for LVDS, DDR LVDS, B-LVDS, and M-L VDS only restricts the I/O differential pair from being selected as an LVPECL pair.

w = D (Differential Pair), P (Pair), or S (Single-Ended). D (Differential Pair) if both members of the pair are bonded out to adjacent pins or are separated only by one GND or NC pin; P (Pair) if both members of the pair are bonded out but do not meet the adjacency requirement; or S (Single-Ended) if the I/O pair is not bonded out. For Differential Pairs (D), adjacency for ball grid packages means only vertical or horizontal. Diagonal adjacency does not meet the requirements for a true differential pair.

B = Bank

y = Bank number (0 3). The Bank number starts at 0 from the northwest I/O bank and proceeds in a clockwise direction.