VSDSquadron FPGA Mini Board - LED Control Implementation

1. Understanding the Verilog Code

Module Overview

The top module is responsible for controlling an RGB LED using an internal oscillator. It includes:

- **LED Outputs:** led_red, led_blue, led_green (for RGB control).
- **Clock Input:** hw_clk (hardware oscillator, though unused in the current code).
- **Test Output:** testwire (debug signal driven by the frequency counter).

Internal Components

1. **Internal Oscillator (SB_HFOSC)**
   - Configured with CLKHF_DIV = "0b10" (divides clock frequency).
   - Generates a clock signal (int_osc) that drives the counter.
2. **Frequency Counter**

- frequency_counter_i (28-bit register) increments on every rising edge of int_osc.
- testwire is assigned frequency_counter_i[5], generating a test signal.

3. **RGB LED Driver (SB_RGBA_DRV)**
   - Controls the hardware LED outputs (led_red, led_green, led_blue).
   - **Current settings:**
     - RGB0_CURRENT = "0b000001" (Red)
     - RGB1_CURRENT = "0b000001" (Green)
     - RGB2_CURRENT = "0b000001" (Blue)
   - **LED Behavior:**
     - Red and Green LEDs are off (RGB0PWM = 0, RGB1PWM = 0).
     - Blue LED is on (RGB2PWM = 1).

---

2. Creating the PCF File (Pin Mapping)

The **Pin Constraint File (PCF)** assigns FPGA pins to module signals.

Signal	Pin	Function
led_red	39	Controls Red LED
led_blue	40	Controls blue LED
led_green	41	Controls green LED
hw_clk	20	External clock input
testwire	17	Debug/test signal

Verifying Pin Assignments

- The **PCF file** matches the **VSDSquadron FPGA Mini board** specifications.
- These mappings ensure the LEDs and test signal are correctly routed to the board's pins.

3. Integrating with the VSDSquadron FPGA Mini Board

Board Setup

1. Connect the FPGA board to a computer via **USB-C**.
2. Ensure FTDI drivers are installed (for serial communication).

Building & Flashing Steps

Run the following commands in the project directory:

```
make clean     # Clears previous builds
make build     # Compiles the Verilog design
sudo make flash  # Programs the FPGA
```

- If successful, the **blue LED should turn on**, confirming correct operation.

---

4. Final Summary

Key Observations

- **Verilog Code:** Generates an internal clock and drives an RGB LED.
- **PCF File:** Correctly assigns FPGA pins to module signals.
- **Board Programming:** The **blue LED turns on**, validating the design.

Challenges & Solutions

Challenge	Solution
FPGA not detected during flashing	Ensured correct USB connection and FTDI setup