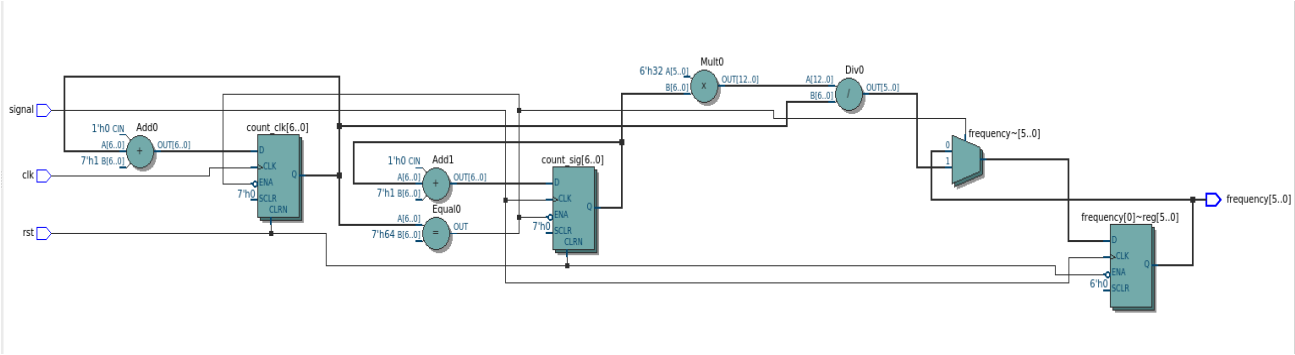
“4. Frequency Calculator”

Design:

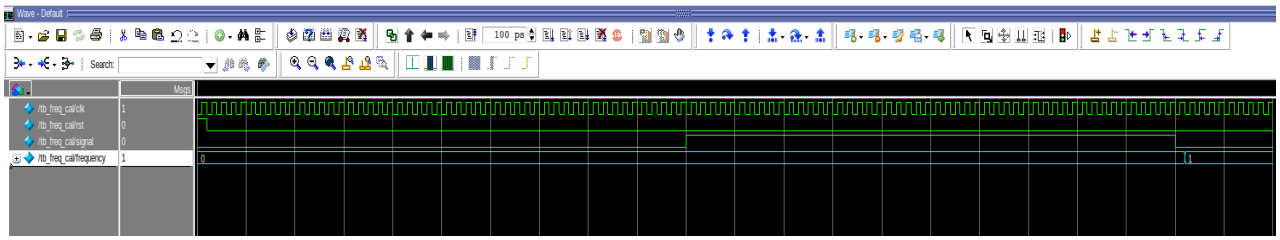
- The module measures input signal frequency by counting signal edges relative to a reference clock.
- A clock counter increments until 100 cycles, then freezes and updates the frequency output.
- A signal counter increments on each negative edge of the input signal within the same window
- The ratio of signal edges to clock cycles is computed to estimate frequency in MHZ range.
- Reset clears counters and output forproper initialization before frequency calculation begins.



Verification (calculation in action):

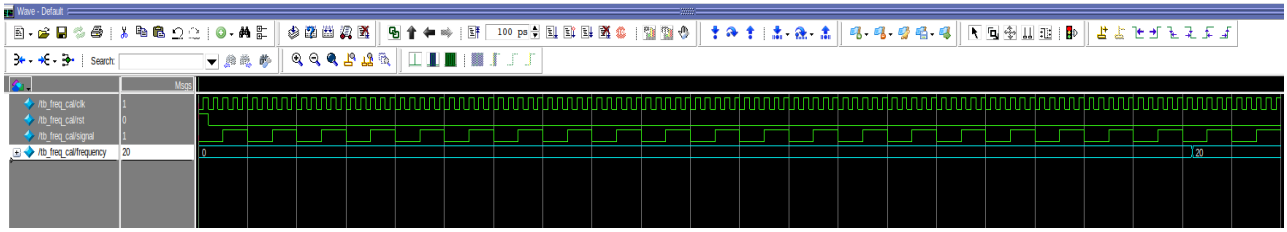
- The testbench instantiates the freq_cal module and drives clock, reset, and signal inputs for measurement.
- A parameter SIG_HALF_FREQ controls the toggling rate of the input signal, allowing flexible frequency generation.
- By adjusting SIG_HALF_FREQ, different unknown signal frequencies can be tested against the frequency calculator.
- The simulation runs for 1100 ns, after which the calculated frequency is displayed in MHz.
- Using this setup, signal frequencies of 1 MHz, 20 MHz, 35 MHz, and 50 MHz were successfully verified.

1. *Input Signal with “Time Period = 1000ns”:*



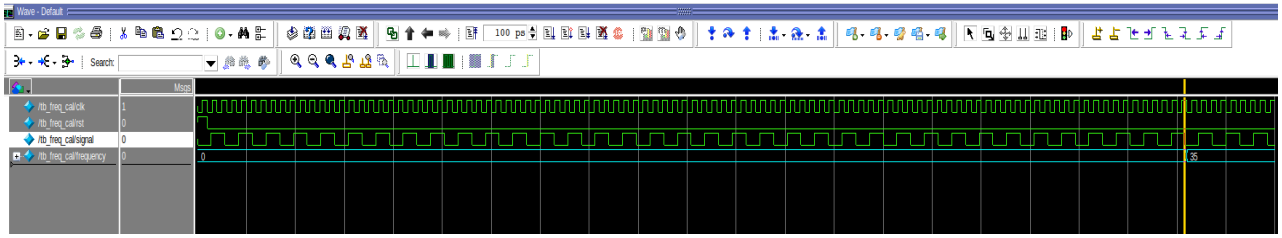
```
VSIM 32> run -all
# Signal frequency is : 1MHz
```

2. *Input Signal with “Time Period = 51ns”:*



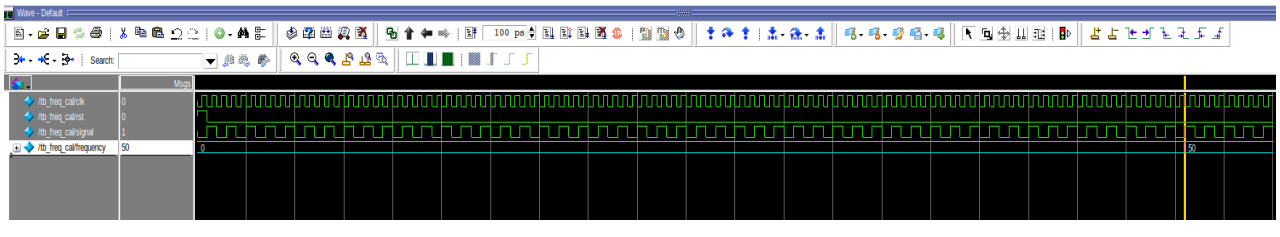
```
VSIM 28> run -all
# Signal frequency is : 20MHz
```

3. *Input Signal with “Time Period = 29ns”:*



```
VSIM 36> run -all
# Signal frequency is : 35MHz
```

4. *Input Signal with “Time Period = 20ns”:*



```
VSIM 40> run -all
# Signal frequency is : 50MHz
```