

All-Digital Phase-Locked Loop In 90-nm Process

目錄

一、	電路功能、規格及架構	3
1.	電路功能	3
2.	電路規格	3
3.	電路架構及目標	3
二、	子電路實現及合成	4
1.	相位及頻率偵測器(Phase/Frequency Detector，簡稱 PFD).....	4
2.	PLL 控制器(PLL Controller)	5
3.	控制碼轉換器(Converter).....	6
4.	數位控制震盪器(Digital Controlled Oscillator，簡稱 DCO).....	6
5.	除頻器(Frequency Divider).....	7
三、	階層式繞線	8
1.	Unconstaint domain(PFD+Converter+DCO)	8
2.	Constaint domain(Controller+Frequency Divider)	8
3.	整合 APR	8
四、	Eda Cloud.....	9
1.	DRC 檢查	9
2.	Post-layout transistor level simulation	9

圖目錄

圖 1、	ADPLL 電路架構圖	3
圖 2、	PLL Loop Simulation	4
圖 3、	Phase/Frequency Detector 架構圖	5
圖 4、	SYN_RTL.sdc(控制器).....	5
圖 5、	SYN_RTL.sdc(Converter).....	6
圖 6、	Digital Controlled Oscillator 架構圖	6
圖 7、	SYN_RTL.sdc(除頻器，最高 400MHz)	7
圖 8、	Avoid IPO changes the netlist.....	8

圖 9、CHIP. globals 設定	9
圖 10、debug pins	9
圖 11、PLL Post-layout Simulation with REF_CLK = 50MHz and M = 1	9
圖 12、PLL Post-layout Simulation with REF_CLK = 50MHz and M = 1(Cont'd).....	10
圖 13、PLL Post-layout Simulation with REF_CLK = 50MHz and M = 2	10
圖 14、PLL Post-layout Simulation with REF_CLK = 50MHz and M = 2(Cont'd).....	10
圖 15、PLL Post-layout Simulation with REF_CLK = 50MHz and M = 3	11
圖 16、PLL Post-layout Simulation with REF_CLK = 50MHz and M = 3(Cont'd).....	11
圖 17、PLL Post-layout Simulation with REF_CLK = 50MHz and M = 4	11
圖 18、PLL Post-layout Simulation with REF_CLK = 50MHz and M = 4(Cont'd).....	12
圖 19、PLL Post-layout Simulation with REF_CLK = 50MHz and M = 5	12
圖 20、PLL Post-layout Simulation with REF_CLK = 50MHz and M = 5(Cont'd).....	12
圖 21、PLL Post-layout Simulation with REF_CLK = 50MHz and M = 6	13
圖 22、PLL Post-layout Simulation with REF_CLK = 50MHz and M = 6(Cont'd).....	13
圖 23、PLL Post-layout Simulation with REF_CLK = 50MHz and M = 7	13
圖 24、PLL Post-layout Simulation with REF_CLK = 50MHz and M = 7(Cont'd).....	14
圖 25、PLL Post-layout Simulation with REF_CLK = 62.5MHz and M = 4	14
圖 26、PLL Post-layout Simulation with REF_CLK = 62.5MHz and M = 4(Cont'd)	14
圖 27、PLL Post-layout Simulation with REF_CLK = 31.25MHz and M = 5	15
圖 28、PLL Post-layout Simulation with REF_CLK = 31.25MHz and M = 5(Cont'd)	15
圖 29、PLL Post-layout Simulation with REF_CLK = 50MHz and M = 1 ~ 7	15

表目錄

表 1、ADPLL 電路規格.....	3
表 2、PFD DeadZone.....	5

一、 電路功能、規格及架構

1. 電路功能

全數位鎖相迴路(All-Digital Phase-Locked Loop，簡稱 ADPLL)，其工作原理為將參考頻率與除頻器輸出的回授頻率，利用相位及頻率偵測器(Phase/Frequency Detector，簡稱 PFD)產生 up/down 的訊號，並傳送給 PLL 控制器產生 dco_code 讓數位控制震盪器(Digital Controlled Oscillator，簡稱 DCO)改變輸出頻率，最後透過除頻器輸出並與參考頻率做比較，重複以上動作維持並持續追蹤參考訊號與輸出訊號之相位、頻率的倍頻關係。

2. 電路規格

Parameter	Description
Target Process	90nm process
Reference Clock (MHz)	
Output Clock (MHz)	
Programmable Input and Feedback Divider	M = 1 ~ 7
Lock-in Time (# cycle)	
Power Consumption (mW)	

表 1、ADPLL 電路規格

3. 電路架構及目標

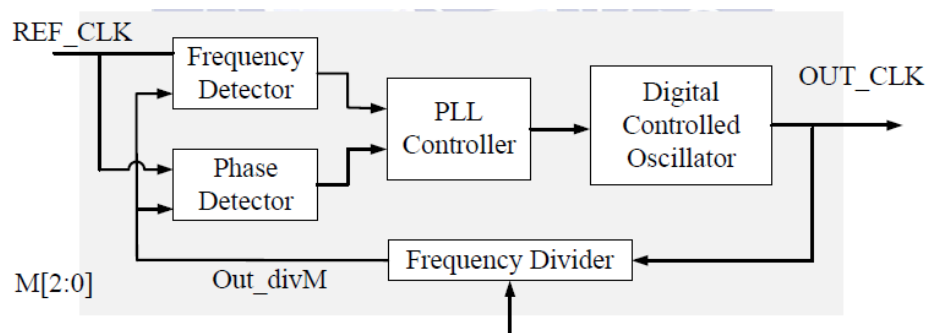


圖 1、ADPLL 電路架構圖

此次專案架構包含了 Phase/Frequency Detector、PLL Controller、Digital Controlled Oscillator、Frequency Divider，以及減少 Jitter 所使用的 Digital Loop Filter(直接將功能內嵌在 PLL Controller 內)，最後目標為在 90nm 製程下通過 post-layout transistor level simulation。

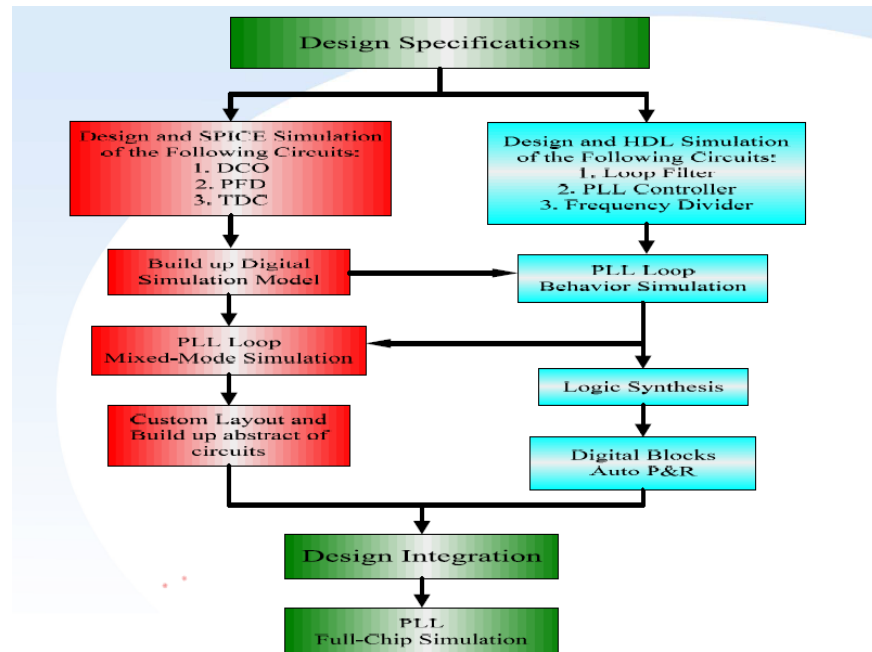


圖 2、PLL Loop Simulation

二、子電路實現及合成

1. 相位及頻率偵測器(Phase/Frequency Detector，簡稱 PFD)

此次專案 PFD 使用的是 Cell-based, three-state, bang-bang PFD 架構，一開始針對 IN 及 FB 的頻率做調整，待頻率追蹤完畢後接著做相位追蹤，利用輸出的 flagU 及 flagD 作為資訊提供給後續電路對數位控制震盪器做加速或減速。此電路的 DeadZone 受到前面 D-Flip/Flop clear pin 的最小 pulse width 所限制，而下圖中的紅色方塊圖是 digital pulse amplifier，可將進來的 phase error 信號放大，讓後一級的 D-Flip/Flop 更容易可以偵測到。

這裡我使用多個並聯的 nand gate 以增強訊號推動後面 digital pulse amplifier 的能力。另外為了使 PFD 傳送給控制器的 flagU、flagD 與 phase clk 不要太靠近，否則容易有 setup time violation，故在 phase clk 訊號前加入一些 buffer 以拉大 phase clk 與 flagU/flagD 的距離。

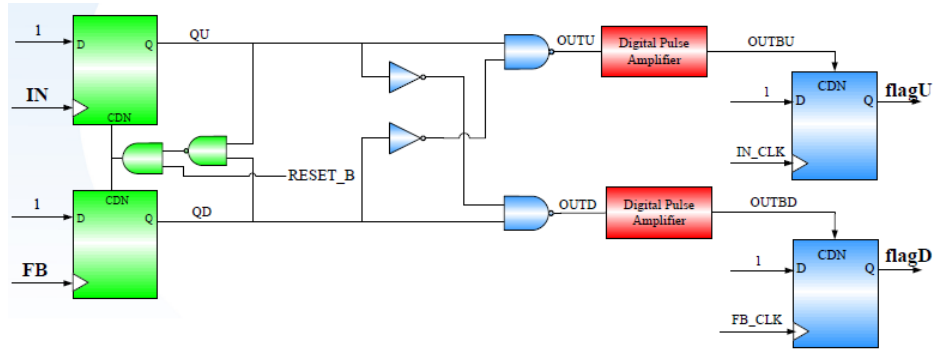


圖 3、Phase/Frequency Detector 架構圖

	(SS, 1.62V, 125°C)	(TT, 1.8V, 25°C)	(FF, 1.98V, 0°C)
DeadZone(單位:ps)	30	19	13

表 2、PFD DeadZone

2. PLL 控制器(PLL Controller)

藉由 PFD 所產生的 up/down 訊號，將領先或落後的資訊傳入 PLL Controller 對 dco_code 進行調整，當 polarity 轉換時，會將 step 除 2 以慢慢達成鎖定。dco_code 的 bits 數量為[7:0]，因應震盪器產生 128 種不同頻率的輸出，為防止 dco_code 溢位，設定當下一次的 dco_code 加減 step 會超過上下限時直接將 dco_code 調至上限或下限值，另外數位濾波器之功能已包含在內，使用的是 Motorola 的架構，針對每次領先或落後的資訊在必要時更新 anchor register 的值。

電路合成時除了 clk 的資訊外，因為還要與其他電路做整合，所以增加了 INPUT_DELAY、OUTPUT_DELAY、PORT_LOADING，如下圖 SYN_RTL.sdc 所示：

```

SYN_RTL.sdc
~/05_ADPLL_90nm/CONTROLLER/SYN

#####
## Set Design Constraints
#####
set CLK_PERIOD 3.7
set CLK_RISING_EDGE 0
set CLK_FALLING_EDGE [expr $CLK_PERIOD/2.0]
set INPUT_DELAY 1.5
set OUTPUT_DELAY 1.0
set PORT_LOADING 0.2

## Clock Constraints ##
create_clock phase_clk -period $CLK_PERIOD -waveform "$CLK_RISING_EDGE $CLK_FALLING_EDGE"

## Set I/O Timing ##
set input_delay $INPUT_DELAY -clock phase_clk [all_inputs -no_clocks]
set output_delay $OUTPUT_DELAY -clock phase_clk [all_outputs]

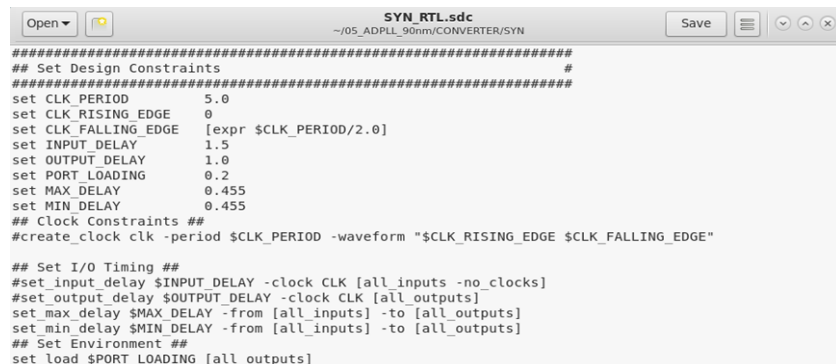
## Set Environment ##
set_load $PORT_LOADING [all_outputs]

```

圖 4、SYN_RTL.sdc(控制器)

3. 控制碼轉換器(Converter)

此為一將 8-bit dco_code 轉換為 128-bit DCO 控制訊號的純組合邏輯電路，因為是組合電路，故不需要下 clk constraint，只要以 MAX_DELAY、MIN_DELAY 取代(設為 0 讓合成最佳化)，再將實際合成出來的 delay 寫回原來的 sdc 檔做合成即可(平衡各個輸入到輸出的 delay)，如下圖所示：



```
#####  
## Set Design Constraints  
#####  
set CLK_PERIOD 5.0  
set CLK_RISING_EDGE 0  
set CLK_FALLING_EDGE [expr $CLK_PERIOD/2.0]  
set INPUT_DELAY 1.5  
set OUTPUT_DELAY 1.0  
set PORT_LOADING 0.2  
set MAX_DELAY 0.455  
set MIN_DELAY 0.455  
## Clock Constraints ##  
#create_clock clk -period $CLK_PERIOD -waveform "$CLK_RISING_EDGE $CLK_FALLING_EDGE"  
  
## Set I/O Timing ##  
#set_input_delay $INPUT_DELAY -clock CLK [all_inputs -no_clocks]  
#set_output_delay $OUTPUT_DELAY -clock CLK [all_outputs]  
set max_delay $MAX_DELAY -from [all_inputs] -to [all_outputs]  
set min_delay $MIN_DELAY -from [all_inputs] -to [all_outputs]  
## Set Environment ##  
set_load $PORT_LOADING [all_outputs]
```

圖 5、SYN_RTL.sdc(Converter)

4. 數位控制震盪器(Digital Controlled Oscillator，簡稱 DCO)

此次專案使用的是 Tri-state inverters matrix 架構，使用並聯的 Tri-state inverter，透過每個 Tri-state inverter 的開或關，可以調整對輸出充放電的速度，雖然這樣的架構可以提供一個高解析度的 delay line，但相對的代價就是較大的面積與功率，這裡我另外在輸出端多串了 35 個 BUFX2，以控制住此 DCO 的最高震盪頻率，避免進到後面的除頻器時脈過快而導致其無法正常運作，在(TT, 1.8V, 25°C)的模擬下，輸出頻率為 188MHz ~ 299MHz。

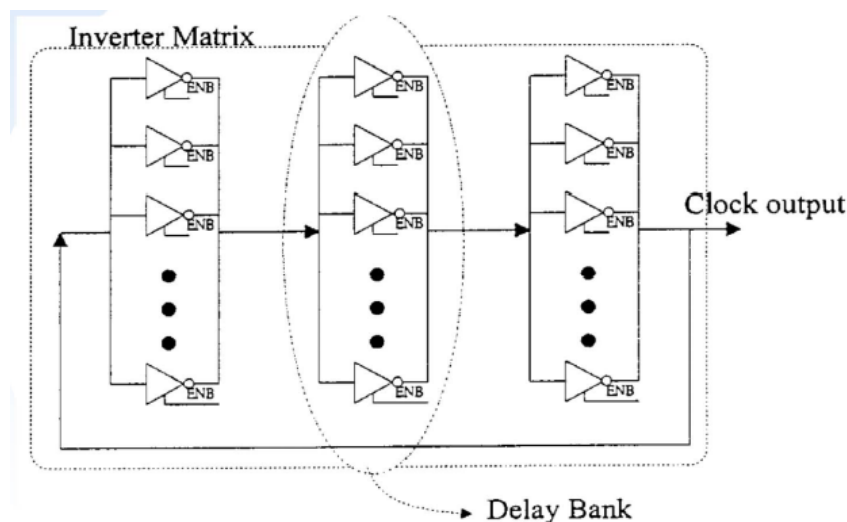
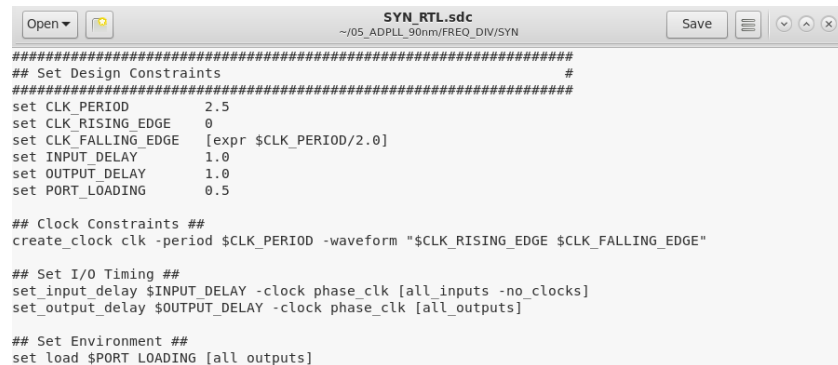


圖 6、Digital Controlled Oscillator 架構圖

5. 除頻器(Frequency Divider)

此次專案使用的是 Counter-based Frequency Divider，不論任何數都可以除，因為在此專案中除頻器之輸出並無 duty cycle 之要求，故並無對 duty cycle 特別做修正，但這可以很簡單的使用 Verilog 做實現，端看實際應用端是否有這項需求。

電路合成時除了 clk 的資訊外，因為還要與其他電路做整合，所以增加了 INPUT_DELAY、OUTPUT_DELAY、PORT_LOADING，如下圖 SYN_RTL.sdc 所示：



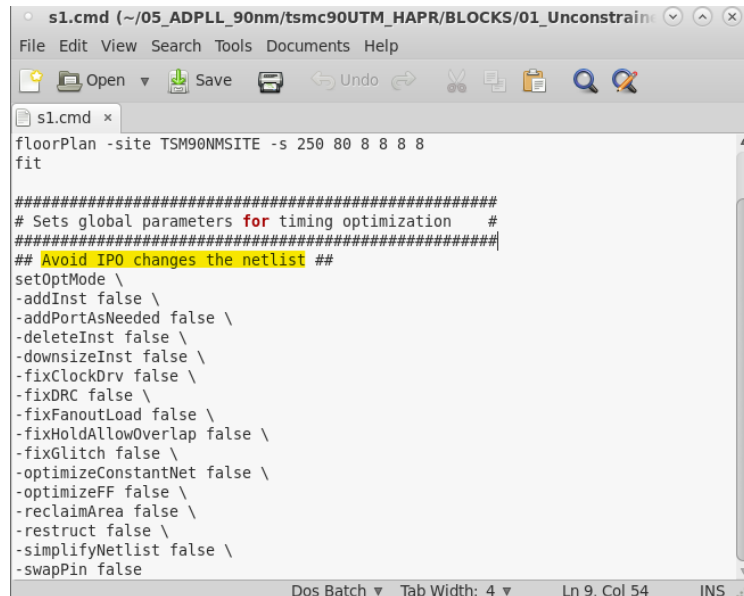
```
#####  
## Set Design Constraints #####  
#####  
set CLK_PERIOD 2.5  
set CLK_RISING_EDGE 0  
set CLK_FALLING_EDGE [expr $CLK_PERIOD/2.0]  
set INPUT_DELAY 1.0  
set OUTPUT_DELAY 1.0  
set PORT_LOADING 0.5  
  
## Clock Constraints ##  
create_clock clk -period $CLK_PERIOD -waveform "$CLK_RISING_EDGE $CLK_FALLING_EDGE"  
  
## Set I/O Timing ##  
set_input_delay $INPUT_DELAY -clock phase_clk [all_inputs -no_clocks]  
set_output_delay $OUTPUT_DELAY -clock phase_clk [all_outputs]  
  
## Set Environment ##  
set_load $PORT_LOADING [all_outputs]
```

圖 7、SYN_RTL.sdc(除頻器，最高 400MHz)

三、階層式繞線

1. Unconstaint domain(PFD+Converter+DCO)

在此 ADPLL 架構中，由於 PFD 及 DCO 的電路中會有 gate 並聯的情況，現有的數位模型無法很好的模擬這樣的 delay，故需要跑 spice 去模擬這樣的非線性現象，並且在 APR 階段，此兩部份的電路會只能做接線，而不能再針對內部電路進行修改(包括換 gate 或改變現有的邏輯閘 size)，並命名為”CORE1”，如下圖指令設定所示。



```
s1.cmd (~/.05_ADPLL_90nm/tsmc90UTM_HAPR/BLOCKS/01_Unconstrain
File Edit View Search Tools Documents Help
s1.cmd x
floorPlan -site TSM90NMSITE -s 250 80 8 8 8 8
fit
#####
# Sets global parameters for timing optimization #
#####
## Avoid IPO changes the netlist ##
setOptMode \
-addInst false \
-addPortAsNeeded false \
-deleteInst false \
-downsizeInst false \
-fixClockDrv false \
-fixDRC false \
-fixFanoutLoad false \
-fixHoldAllowOverlap false \
-fixGlitch false \
-optimizeConstantNet false \
-optimizeFF false \
-reclaimArea false \
-restruct false \
-simplifyNetlist false \
-swapPin false
Dos Batch Tab Width: 4 Ln 9, Col 54 INS
```

圖 8、Avoid IPO changes the netlist

2. Constraint domain(Controller+Frequency Divider)

將控制器及除頻器兩個需要有 clk constraint 的部份一起做繞線，並命名為”CORE2”，讓後續整合時可以直接使用現階段繞線出來的結果。

3. 整合 APR

將前面兩部份做整合，注意此時繞線時不能再看進前面的 block 做調整，僅能做接線動作，此外為了避免設計變成 pad limit design，故只留少數訊號接 IO PAD，其餘需要 debug 的訊號，將其用 buf 方式另外加入，並且繞線時使用 placement blockage 限制這些 std cell 的擺放區域(避免在延伸接電源時碰到 Hardblock)，最終在送上 Eda Cloud 做完整 Layout 檢查前須通過 DRC 檢查。



圖 9、CHIP. globals 設定

```

BUFX2 B1(.Y(dco_code[0]), .A(c_dco_code[0]));
BUFX2 B2(.Y(dco_code[1]), .A(c_dco_code[1]));
BUFX2 B3(.Y(dco_code[2]), .A(c_dco_code[2]));
BUFX2 B4(.Y(dco_code[3]), .A(c_dco_code[3]));
BUFX2 B5(.Y(dco_code[4]), .A(c_dco_code[4]));
BUFX2 B6(.Y(dco_code[5]), .A(c_dco_code[5]));
BUFX2 B7(.Y(dco_code[6]), .A(c_dco_code[6]));
BUFX2 B8(.Y(dco_code[7]), .A(c_dco_code[7]));

BUFX2 B9 (.Y(flagU), .A(c_flagU));
BUFX2 B10(.Y(flagD), .A(c_flagD));
BUFX2 B11(.Y(polarity), .A(c_polarity));
BUFX2 B12(.Y(phase_clk), .A(c_phase_clk));

```

圖 10、debug pins

四、Eda Cloud

1. DRC 檢查

由於國家晶片中心將實際 90nm 製程電晶體 layout 的資訊鎖在 eda cloud 上，故須將我們在工作站上面繞線後的結果透過 ftp 上傳至 eda cloud 做完整的 DRC 檢查。

2. Post-layout transistor level simulation

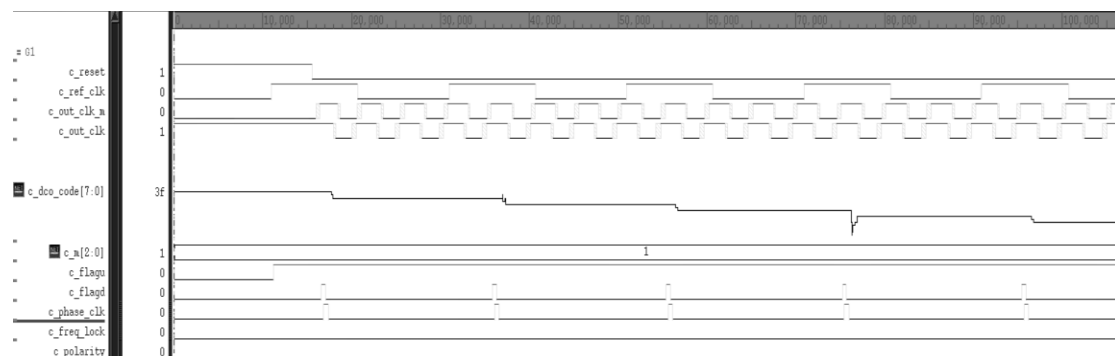


圖 11、PLL Post-layout Simulation with REF_CLK = 50MHz and M = 1

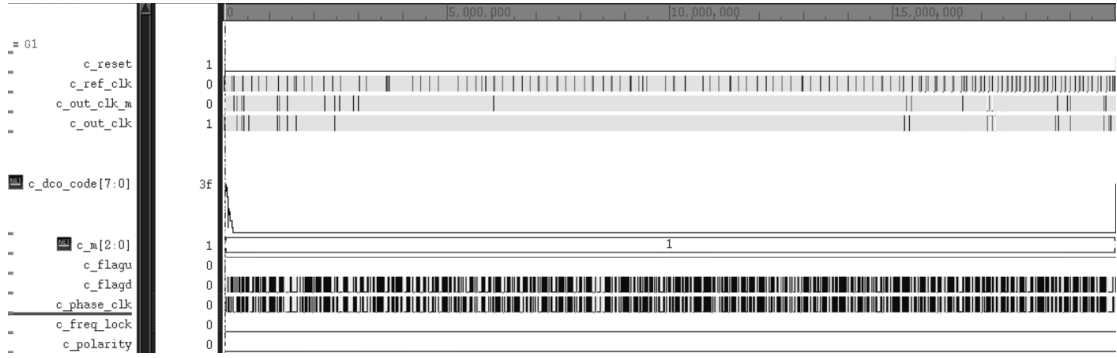


圖 12、PLL Post-layout Simulation with REF_CLK = 50MHz and M = 1(Cont'd)

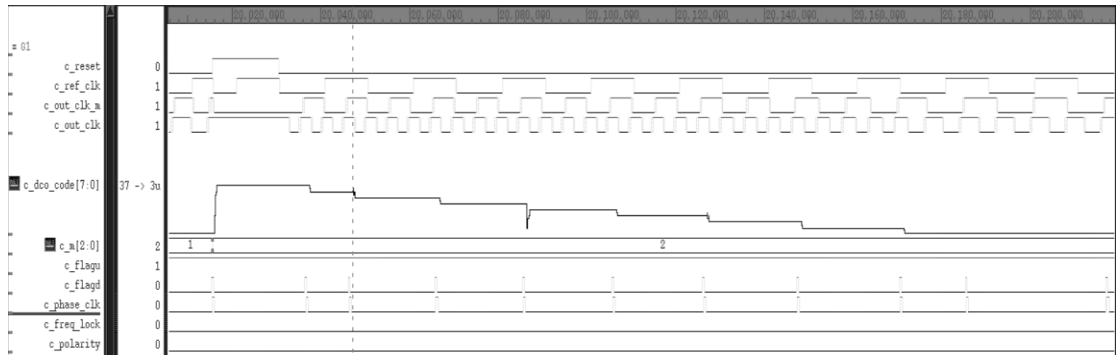


圖 13、PLL Post-layout Simulation with REF_CLK = 50MHz and M = 2

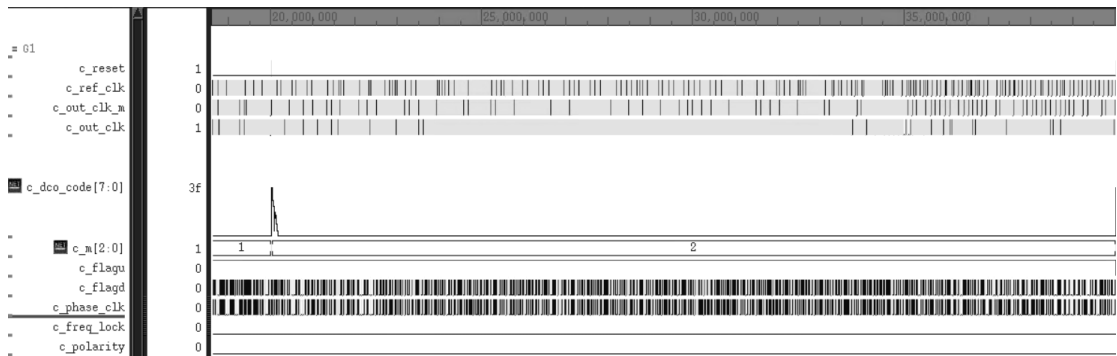


圖 14、PLL Post-layout Simulation with REF_CLK = 50MHz and M = 2(Cont'd)

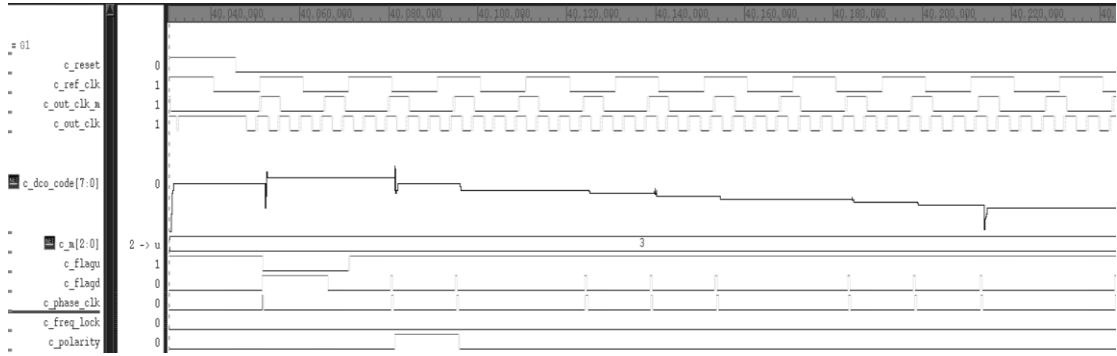


圖 15、PLL Post-layout Simulation with REF_CLK = 50MHz and M = 3

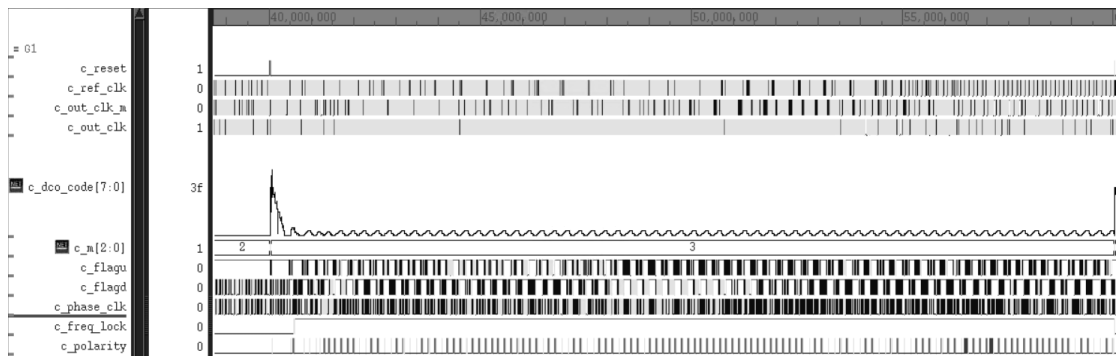


圖 16、PLL Post-layout Simulation with REF_CLK = 50MHz and M = 3(Cont'd)

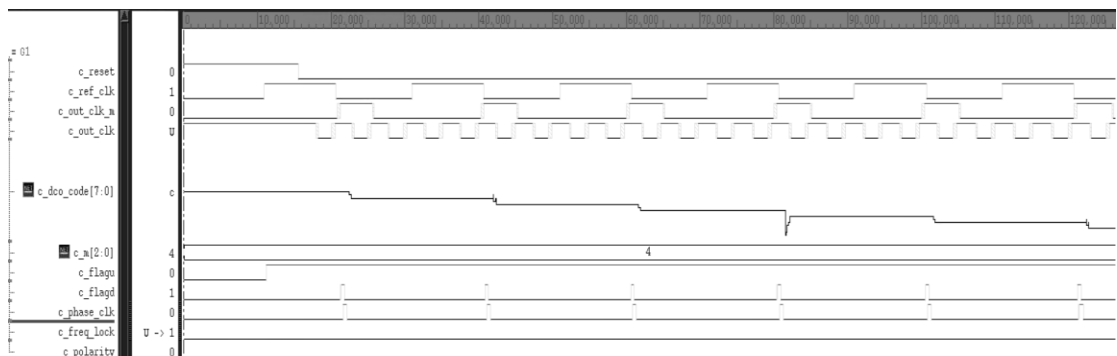


圖 17、PLL Post-layout Simulation with REF_CLK = 50MHz and M = 4

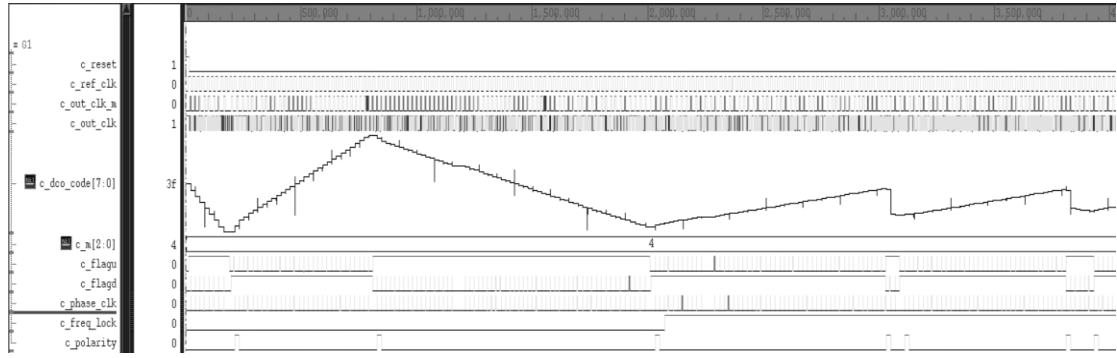


圖 18、PLL Post-layout Simulation with REF_CLK = 50MHz and M = 4(Cont'd)

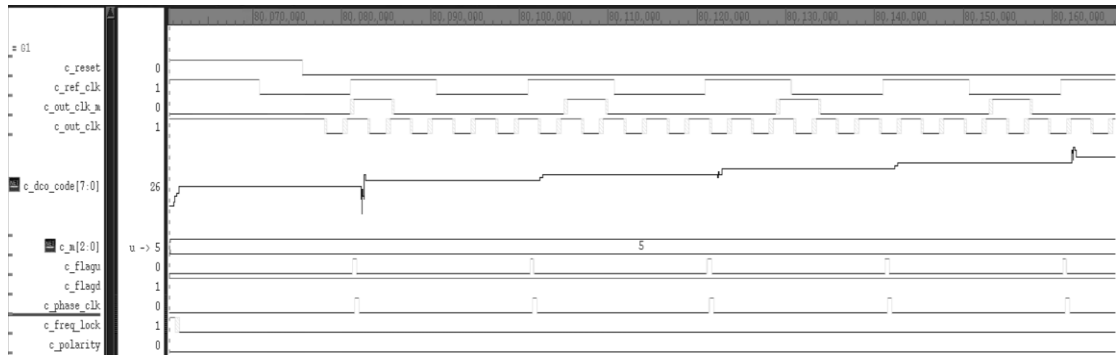


圖 19、PLL Post-layout Simulation with REF_CLK = 50MHz and M = 5

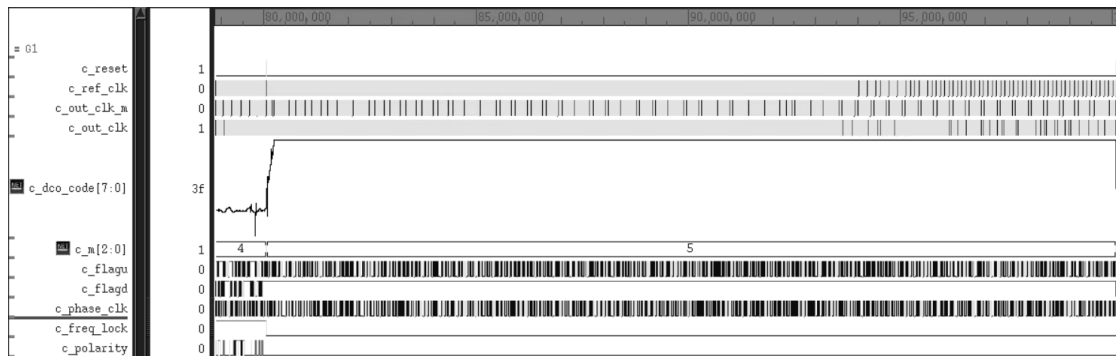


圖 20、PLL Post-layout Simulation with REF_CLK = 50MHz and M = 5(Cont'd)

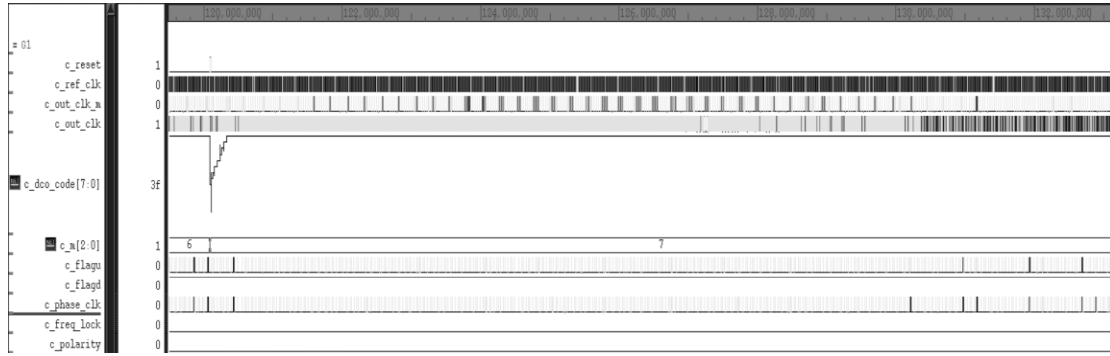


圖 24、PLL Post-layout Simulation with REF_CLK = 50MHz and M = 7(Cont'd)

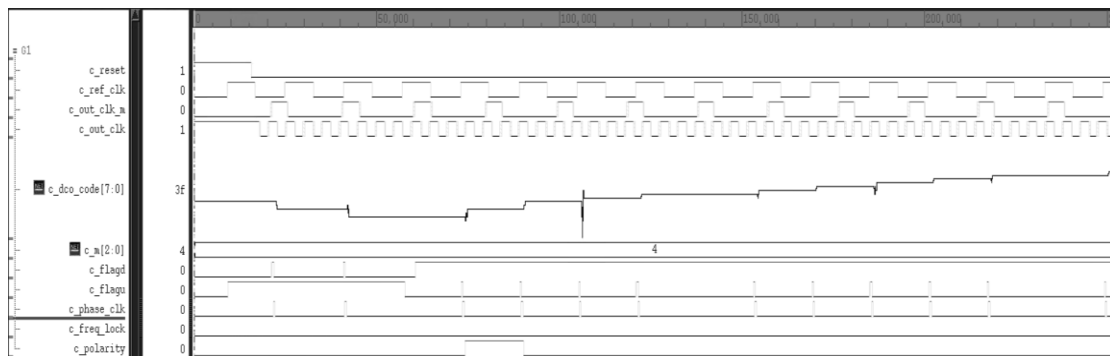


圖 25、PLL Post-layout Simulation with REF_CLK = 62.5MHz and M = 4

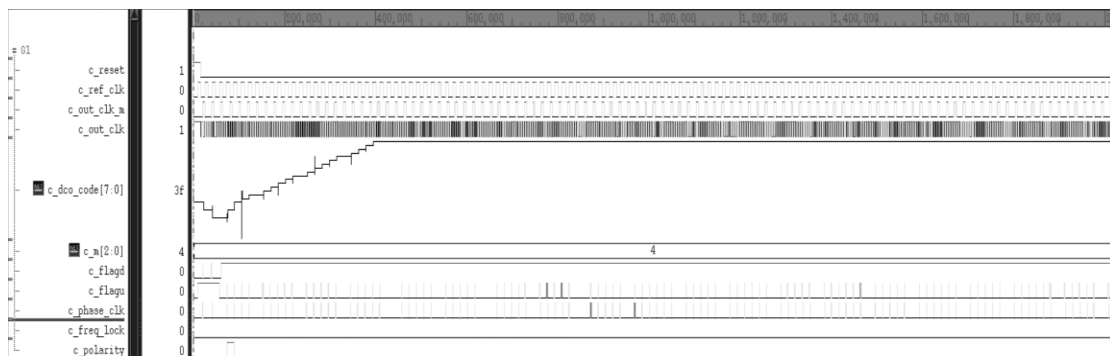


圖 26、PLL Post-layout Simulation with REF_CLK = 62.5MHz and M = 4(Cont'd)

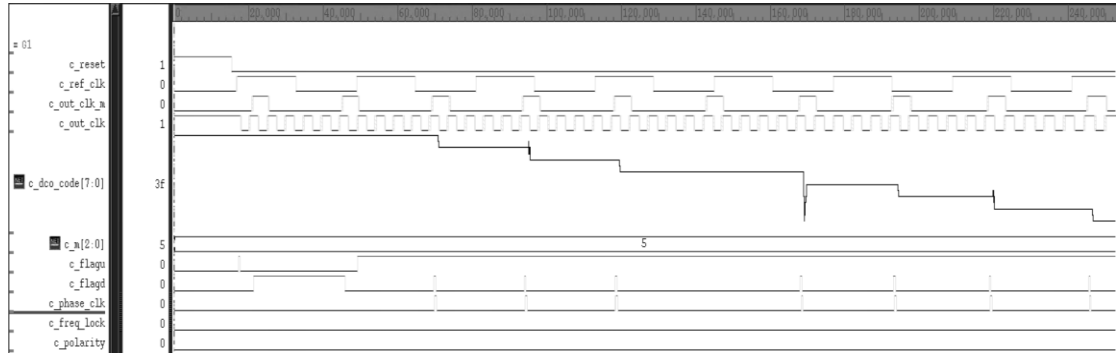


圖 27、PLL Post-layout Simulation with REF_CLK = 31.25MHz and M = 5

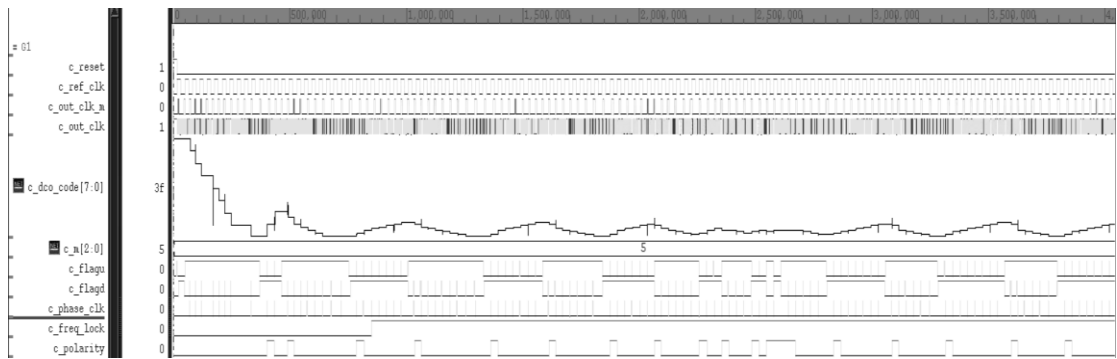


圖 28、PLL Post-layout Simulation with REF_CLK = 31.25MHz and M = 5(Cont'd)

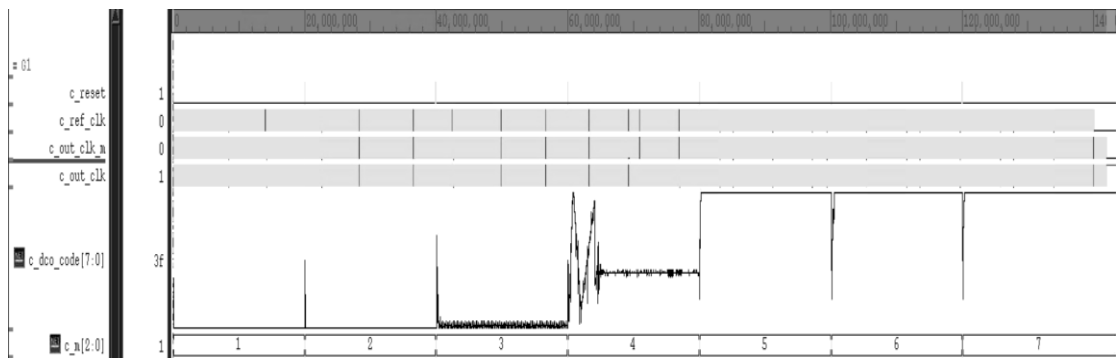


圖 29、PLL Post-layout Simulation with REF_CLK = 50MHz and M = 1 ~ 7