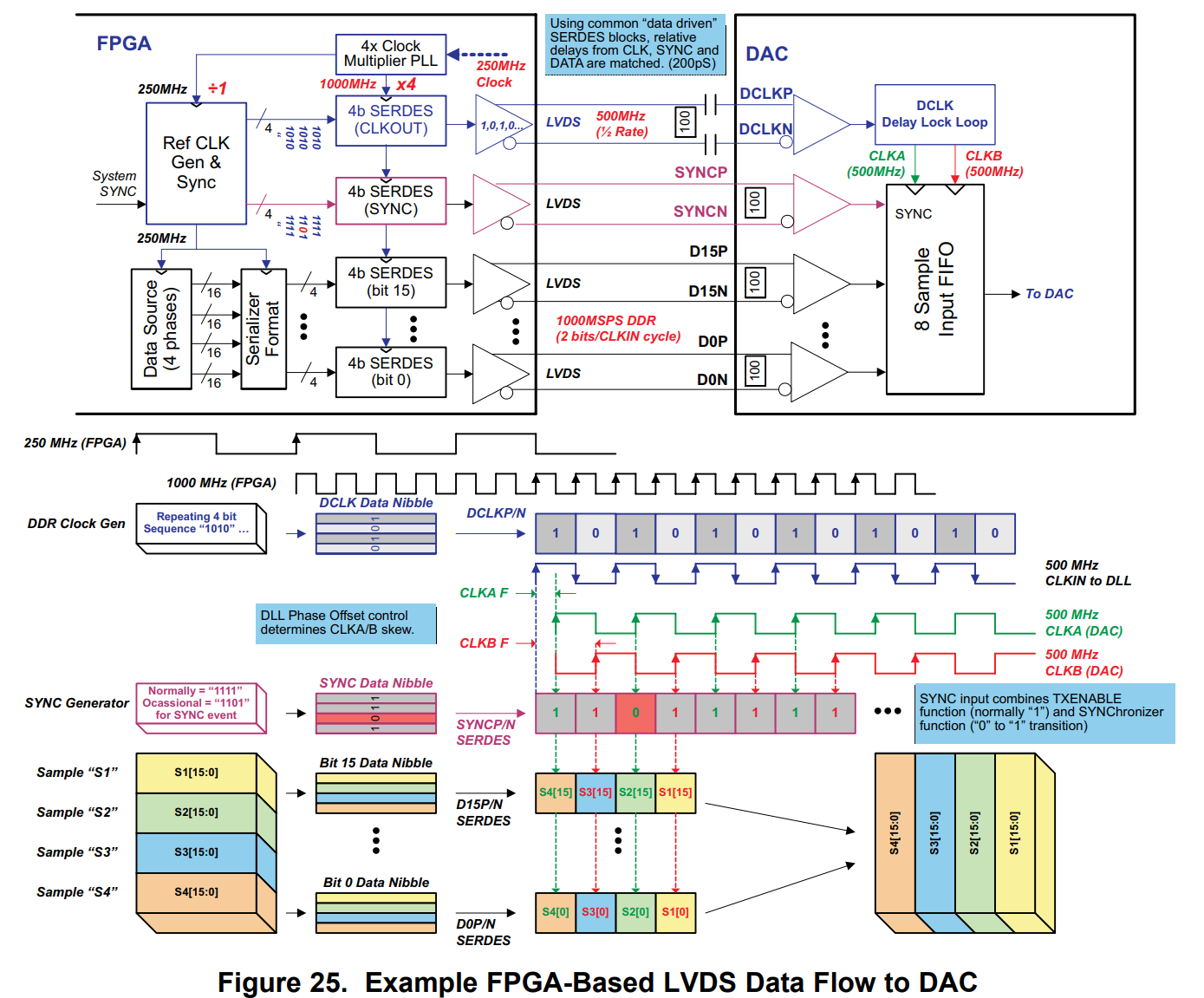
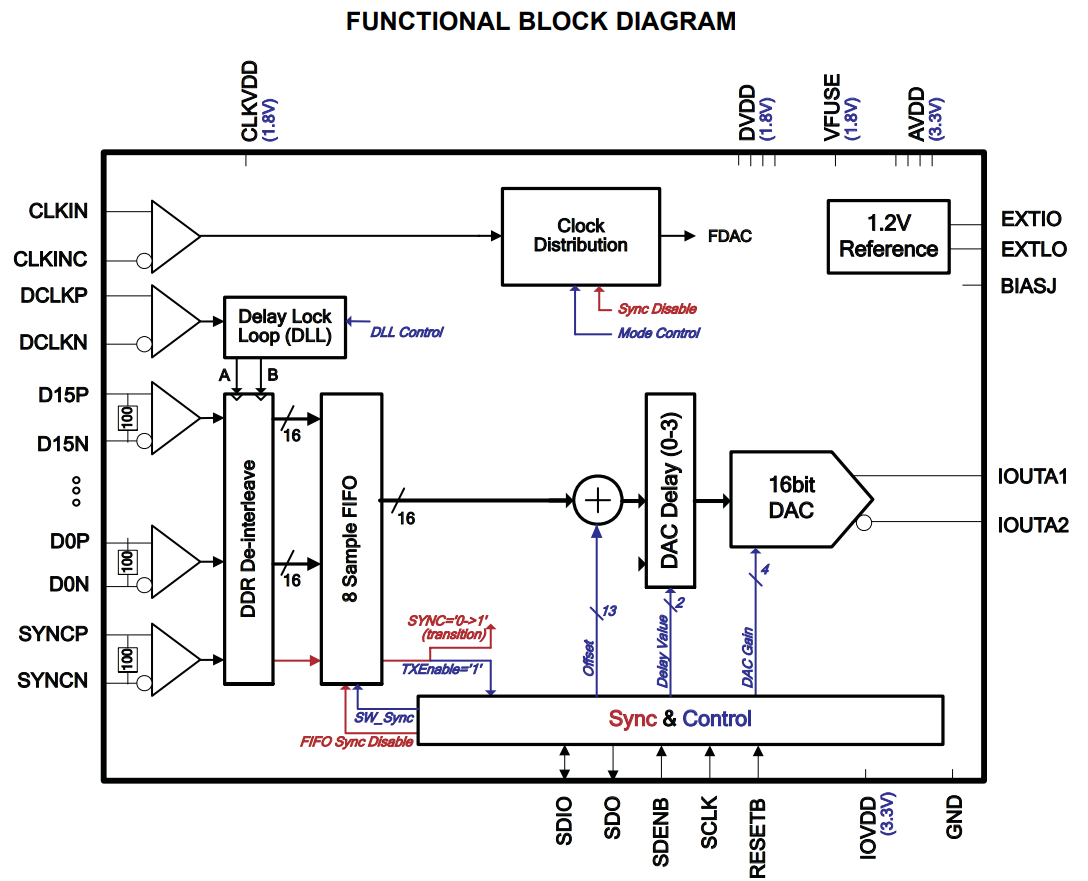
**DAC5681示例工程**

|  |  |  |  |
| --- | --- | --- | --- |
| Document Version Control | | | |
| Version | Date | Author | Changes |
| 1.0.0 | 2024.10.17 | 王雨霄 | 实现了DAC5681示例工程 |

# 一、功能概述

DAC5681是TI公司产出的16bit@1GSPS高速DAC，采用LVDS接口作为芯片的数据传输接口（此DAC具有两个时钟输入，CLKP/N作为采样时钟输入，由外部时钟芯片提供；DCLKP/N作为LVDS数据时钟，和LVDS数据一起由FPGA通过OSERDES提供）：



本设计旨在提供基于xczu19eg-ffvc1760-2-i芯片、xc7vx690tffg1761-2、xcku115-flva1517-2-i芯片运行的DAC5681驱动示例工程，包含DAC5681及其外围器件的接口驱动、寄存器配置驱动等。

本设计主要功能特性如下：

1. 支持DAC5681的LVDS接口驱动，内置ODELAY功能并支持VAR\_LOAD模式下的delay值调节（zu、ku芯片中使用E3系列原语，v7芯片中使用E2系列原语）；
2. 内置输出频率可控的并行DDS源，支持对DAC5681进行单音信号输出测试；
3. 支持DAC5681及HMC7044的3线SPI寄存器配置，并提供寄存器配置样例；
4. zu工程使用PYNQ作为PS控制器，提供ipynb配置程序；v7、ku工程使用Microblaze软核作为PS控制器，提供cpp配置程序；

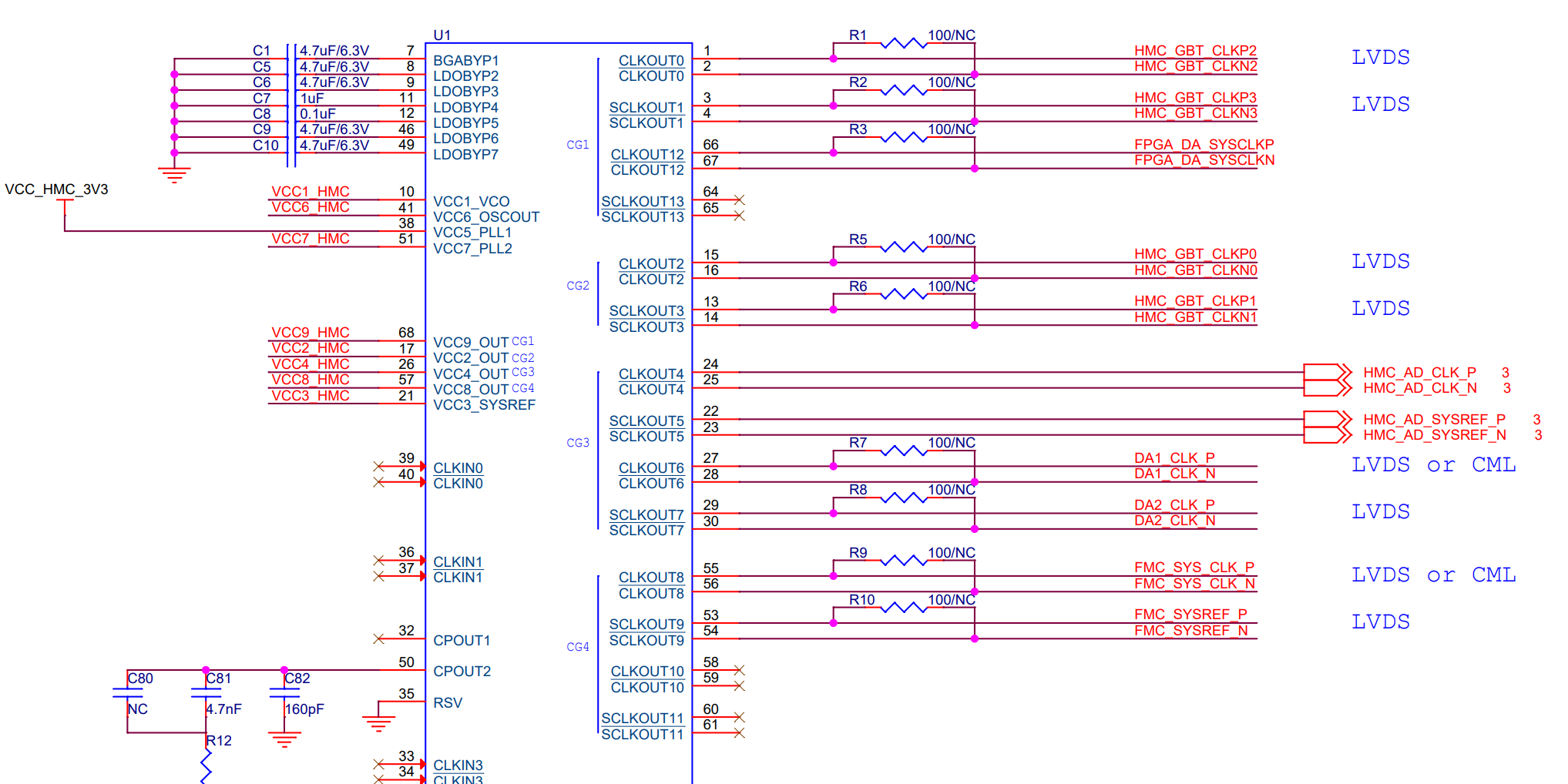


注：

1. 本设计依托的子卡中包含2片DAC5681芯片，2片芯片独立工作，驱动方式完全相同，故上图仅体现了单片的驱动结构（本设计实际支持驱动2片DAC5681芯片）。
2. 上图重在体现xczu19eg工程中的LVDS驱动方式及其时钟网络，用于进行功能控制的GPIO等结构未在上图中体现。

# 二、时钟规划

本设计中使用的子卡采用7044+DAC5681的结构，通过7044为DAC5681提供采样时钟、为FPGA提供参考时钟：



**FPGA参考时钟**

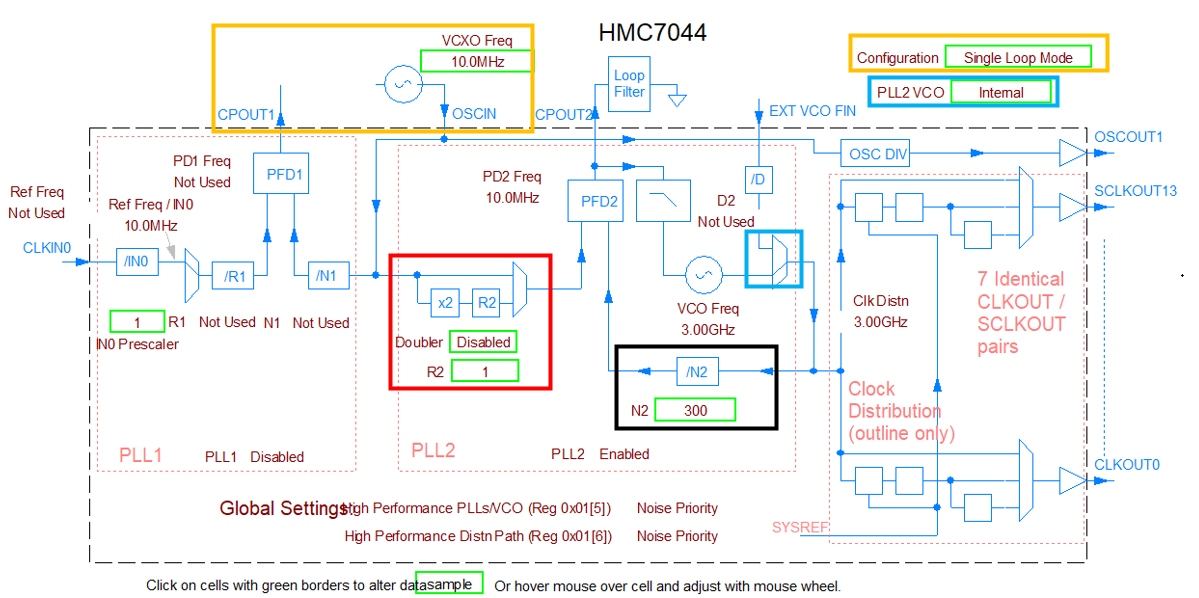
**DAC采样时钟**

本设计令DAC5681以1GSPS最大采样率工作，故采样时钟CLK需为1GHz，且16bit数据对应的OESREDS的输出线速率也应为1Gbps；若这些OSERDES采用8位并行接口+DDR输出模式，则并行数据输入对应的时钟频率为1G/8=125MHz，串行数据输出对应的时钟频率为1G/2=500MHz；此外，DAC5681的DCLK时钟采用DDR模式对LVDS数据进行采样，故DCLK时钟频率也为500MHz，而DCLK是通过OSERDES循环输出0、1序列产生的，故这一OSERDES的输出线速率也为1Gbps，同样需要125MHz及500MHz时钟；根据DAC5681手册中的Figure 25建议，此时应为FPGA提供250MHz的参考时钟，故时钟规划如下：



# 三、HMC7044配置

首先，使用ADIsimCLK 1.70软件计算PLL环路参数（软件中绿色框内的参数可供用户修改）：



1. 右上角橙色框中的“Configuration”参数用于设置7044工作在单环/双环模式：当7044工

作在双环模式时，左上角橙色框内会出现一条连线，将PLL1的CPOUT1连接至PLL2的OSCIN（即PLL2的输入时钟由PLL1产生）；当7044工作在单环模式时（本设计采用此模式），CPOUT1与OSCIN断连，PLL2的输入时钟来自外部，直接设置VCXO Freq参数等于晶振频率（在本设计中为10MHz）。

1. 中间红色框内存在一个二倍频器（可设置开闭）及分频器（R2为分频系数，可在1-4095

取值），将VCXO Freq转为PFD2的输入频率PD2 Freq：当打开二倍频器时，PD2 Freq = VCXO Freq\* 2 / R2；在本设计中，为保证时钟质量关闭了二倍频器，并设置R2 = 1，故PD2 Freq = 10MHz。

1. 黑色框中的N2用于设置VCO频率（7044内置VCO频率需在2.4GHz~3.2GHz范围内），

关系为：VCO Freq / N2 = PD2 Freq，即VCO Freq = PD2 Freq \* N2；在本设计中，VCO目标频率为3GHz，故N2设置为300。

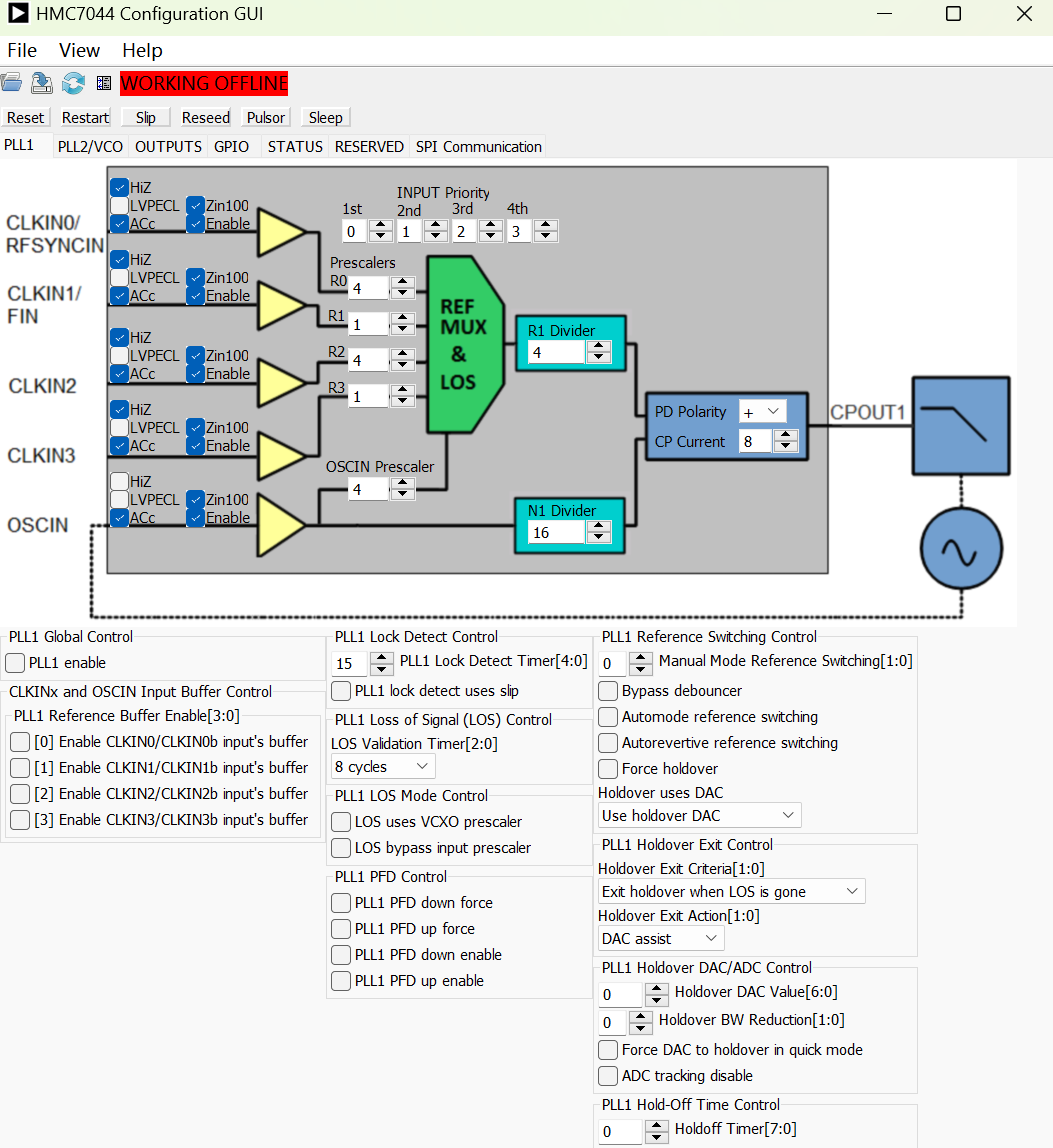
1. 右上角蓝色框中的PLL2 VCO参数用于设置PLL2使用外部/内部VCO，当设置为Internal

模式时，中间蓝色框中的MUX将选用7044内置VCO的输出送入后级。

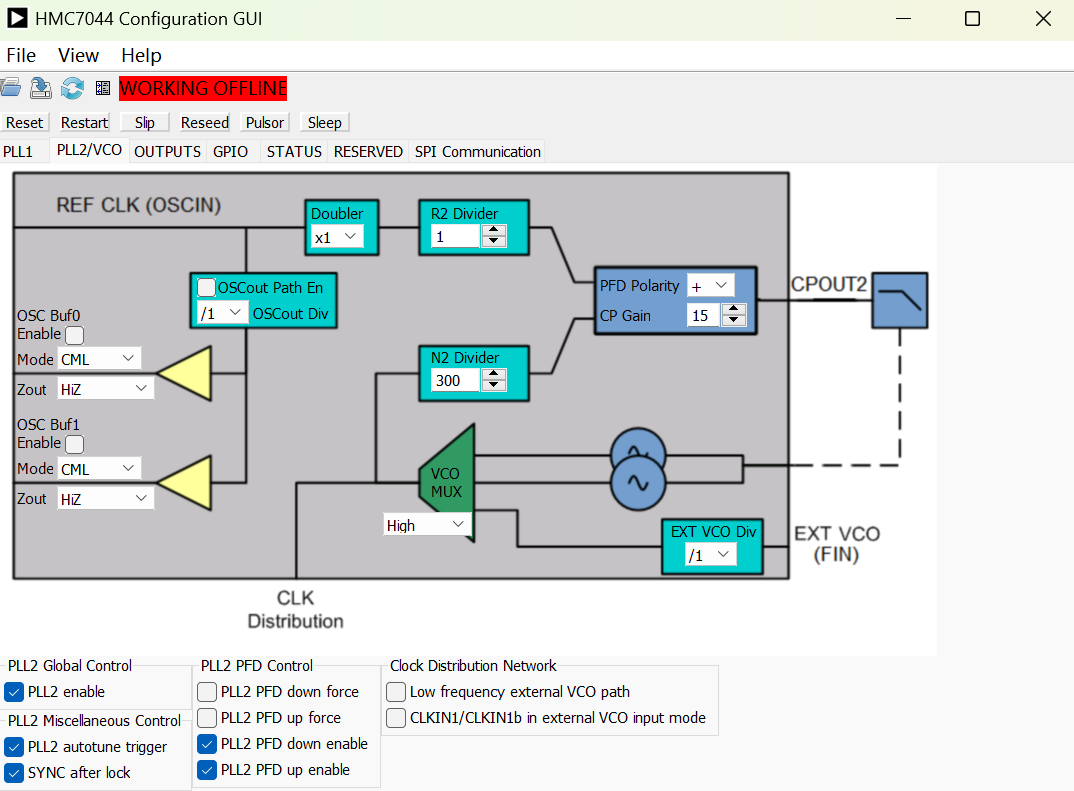
1. 最终的输出时钟由VCO Freq分频得到，分频系数可设置为2-4096间的任意偶数或1、

3、5这三种奇数（在HMC704X\_GUI软件中设置，没有体现在ADIsimCLK 1.70软件中）；

其次，使用HMC704X\_GUI软件进行详细配置，并导出寄存器：

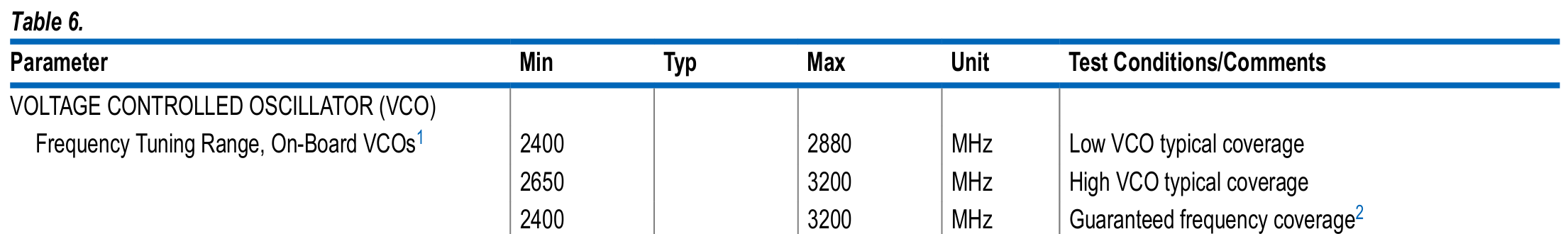


本项目中7044工作在单环模式，故关闭PLL1各项功能；

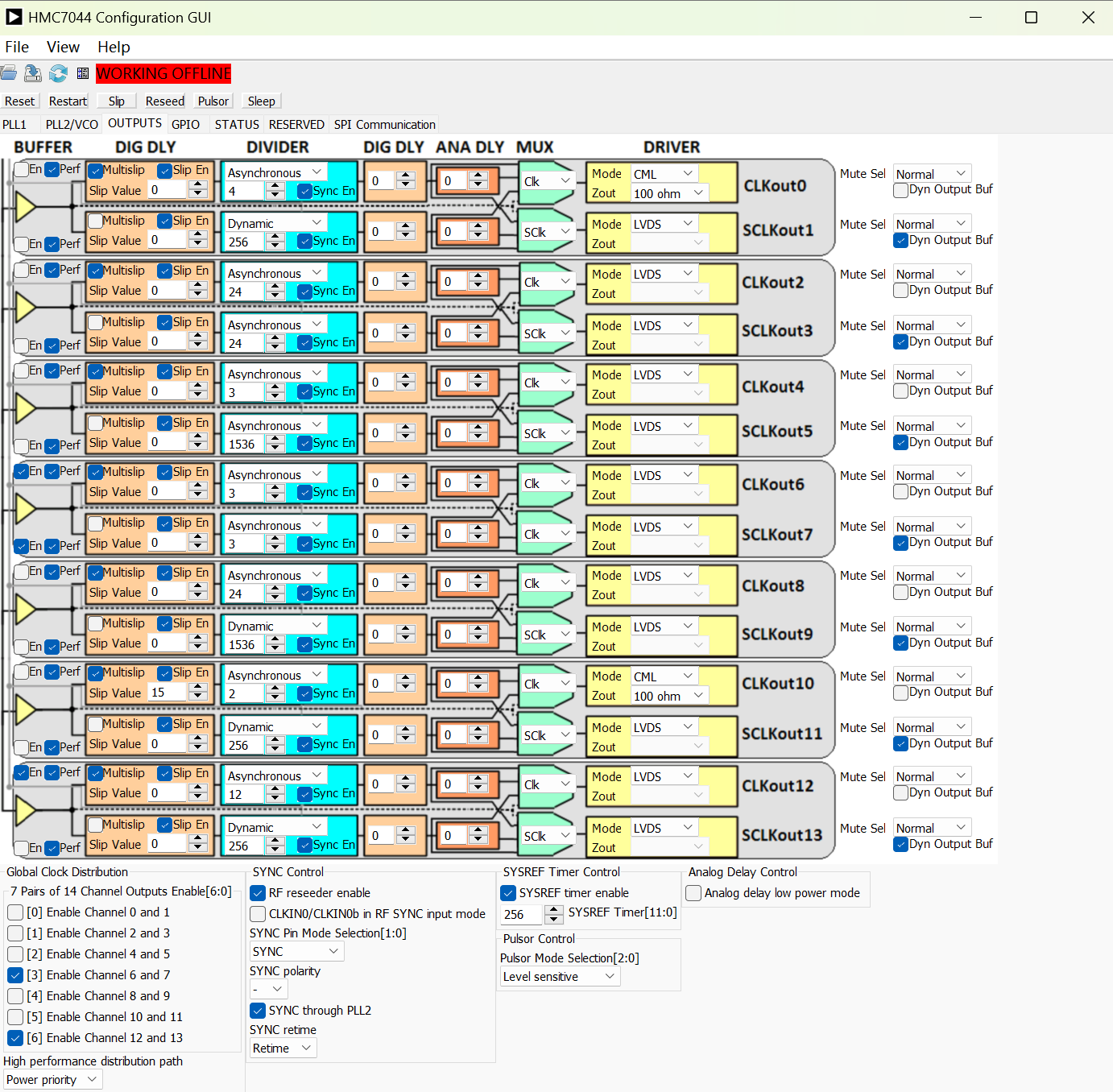


1、修改“Doubler”、“R2 Divider”、“N2 Divider”三个参数，使其与上一步在ADIsimCLK 1.70软件中设置的相应参数保持一致；

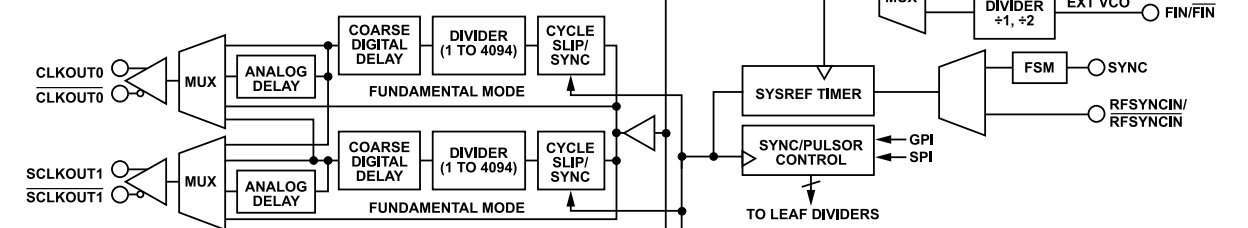
2、依照当前VCO频率选择红色框中的参数为“High”或“Low”：



本项目中设置的VCO频率为3GHz，故选择“High”；



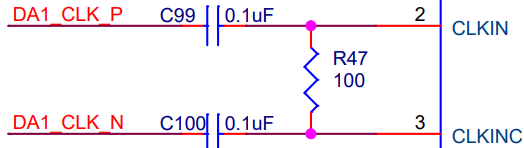
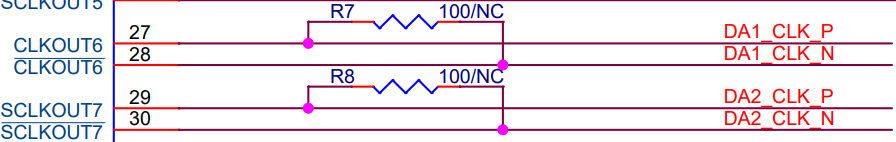
1. 7044共14个输出时钟，分为7组，每组包含一个CLKout（DCLK）及SCLKout（SYSREF）；

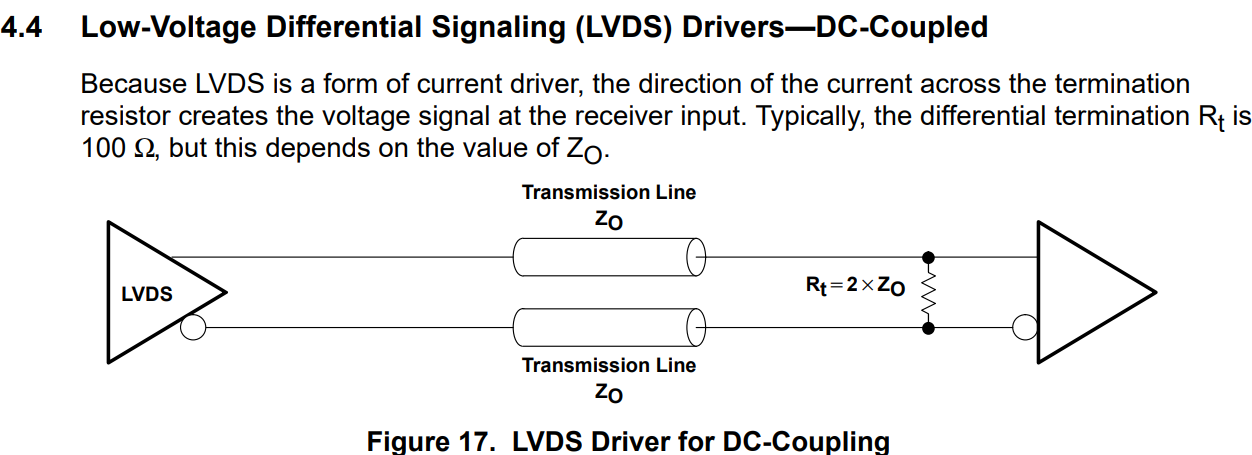


CLKout及SCLKout均连接至时钟选择器上，可各自选择Device CLK信号或SYSREF信号进行输出；本设计中不涉及JESD204接口，故三个输出时钟均设置为Clk模式。

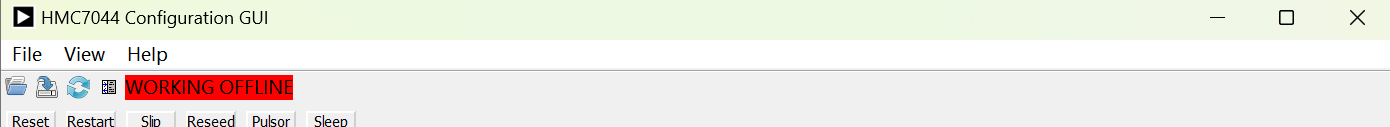
1. 左下角红色框中的7个选项用于控制各组的输出使能，各组内的“En”选项用于设置组内具体通道的输出使能（推荐关闭未使用的时钟通道）；
2. 7044最终输出由VCO Freq分频得到，中间黑色框中的数字用于设置输出端分频系数，分频系数可设置为2-4096间的任意偶数或1、3、5这三个奇数；

4、右侧黑色框中可设置相应通道的电平输出标准，在本设计中均采用LVDS标准；

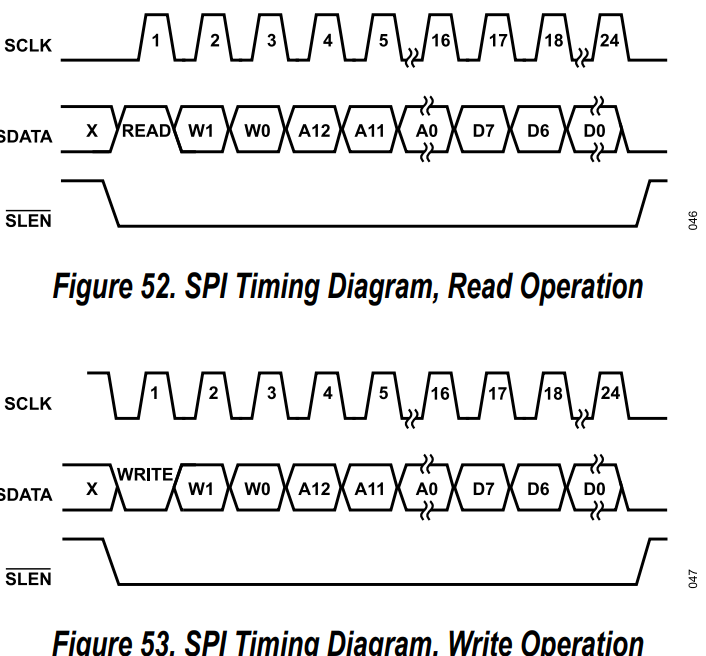




配置完成后，点击菜单栏第二个按钮，保存当前寄存器配置值，以用于SPI配置：



HMC7044采用三线SPI，特性如下：

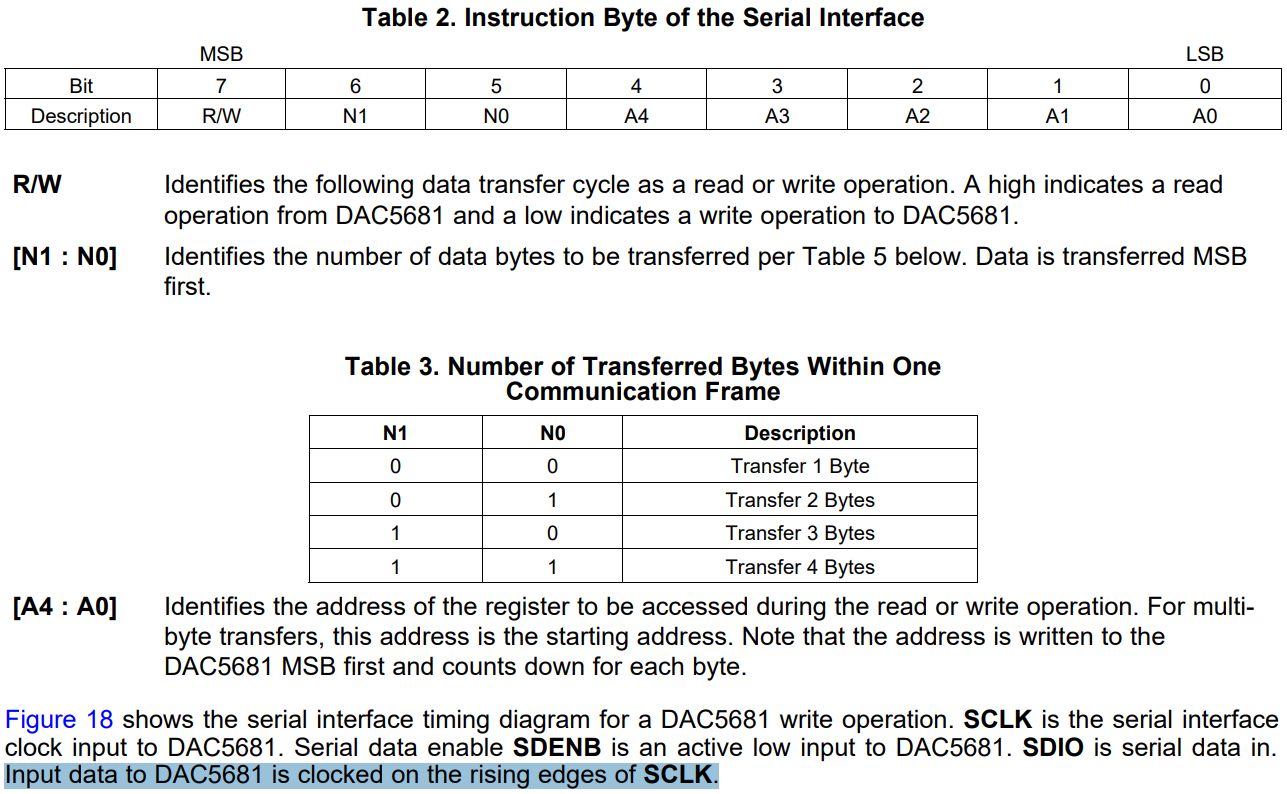
****

1. .信号在SCK下降沿改变，在SCK上升沿被采样，CPOL=0，CPHA=0；
2. .W1及W0应设为0；
3. .SPI最大速率为10MHz（本设计中采用5MHz）；

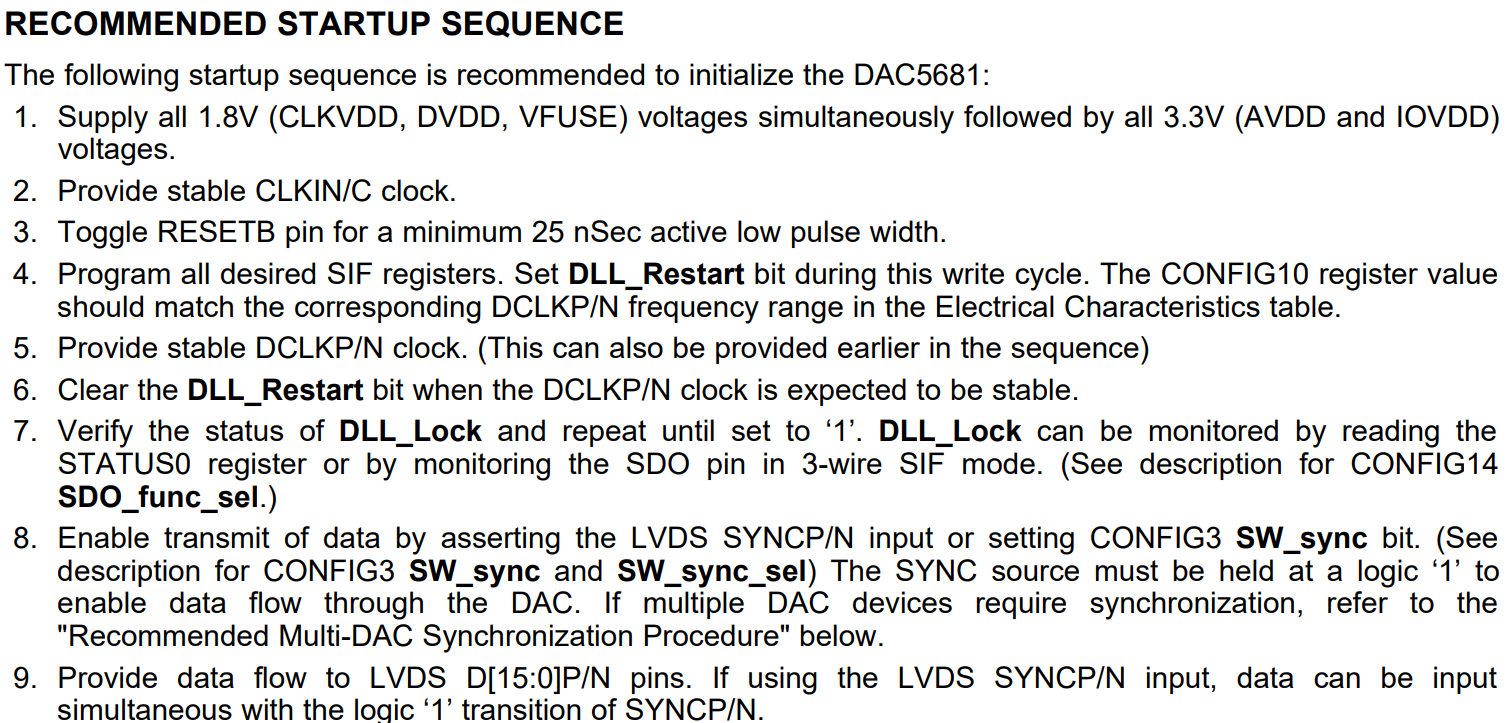
注：HMC7044寄存器数量较多，配置顺序在此不详细列出，具体请参考configs文件夹中dac5681.ipynb文件对HMC7044寄存器的操作行为。

# 四、DAC5681配置

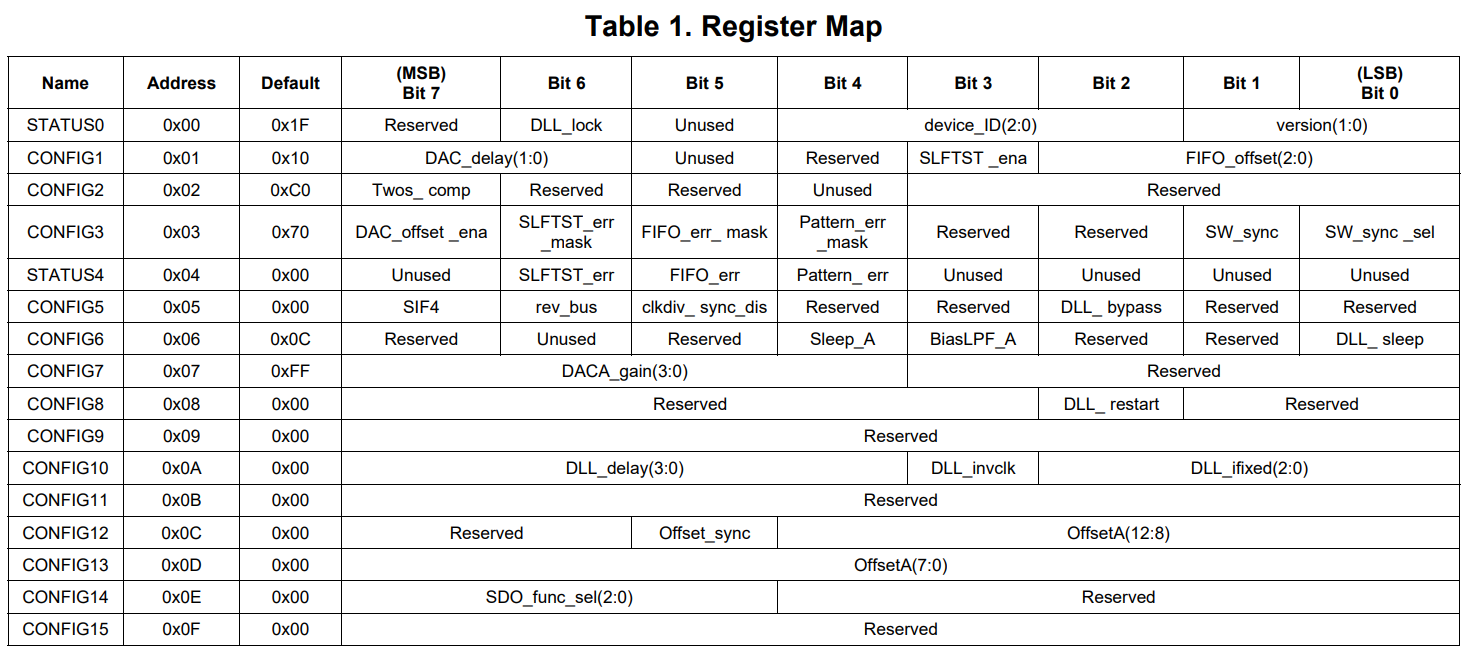
DAC5681可采用三线或四线SPI（可通过寄存器切换，默认模式为三线，本设计中采用默认模式），特性如下：



推荐启动顺序如下：



需要注意的是，SYNC信号具备同步、LVDS传输使能两个功能——进行多片同步时，该信号需要由外部引脚提供；单片工作时，该信号仅具备传输使能的功能，可由SPI寄存器配置来替代外部引脚输入。本设计令DAC5681单片工作，并支持以上两种SYNC控制方式。



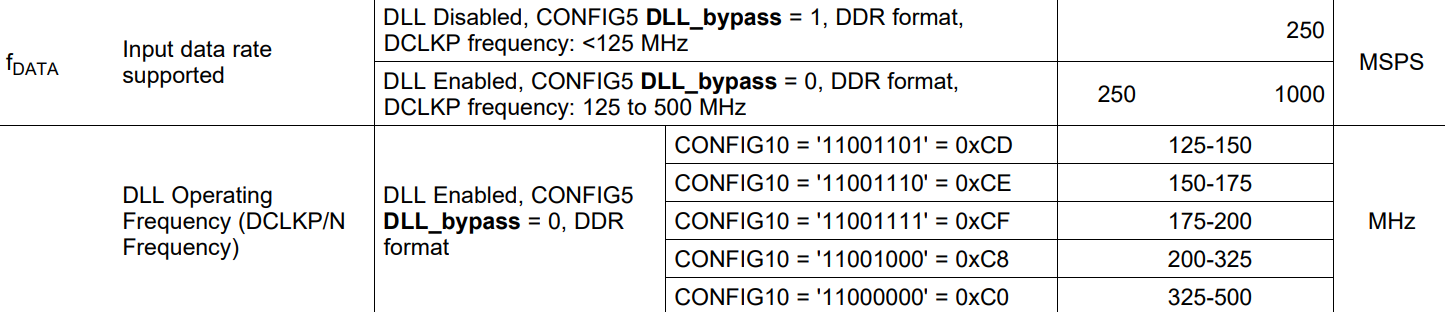
当使用内置SYNC寄存器时，DAC5681具体配置顺序如下：

|  |
| --- |
| HMC7044时钟稳定后，通过Reset引脚进行硬复位（低电平复位，复位持续时间>25ns） |
| 依次配置0x01~0x0F寄存器 |
| 通过GPIO开启DCLK |
| 等待1ms |
| 配置0x08寄存器，解除DLL复位 |
| 循环读取0x00寄存器，直至DLL锁定 |
| 配置0x03寄存器，启动内置SYNC |
| 通过GPIO开启LVDS数据传输 |

当使用外置SYNC引脚时，DAC5681具体配置顺序如下：

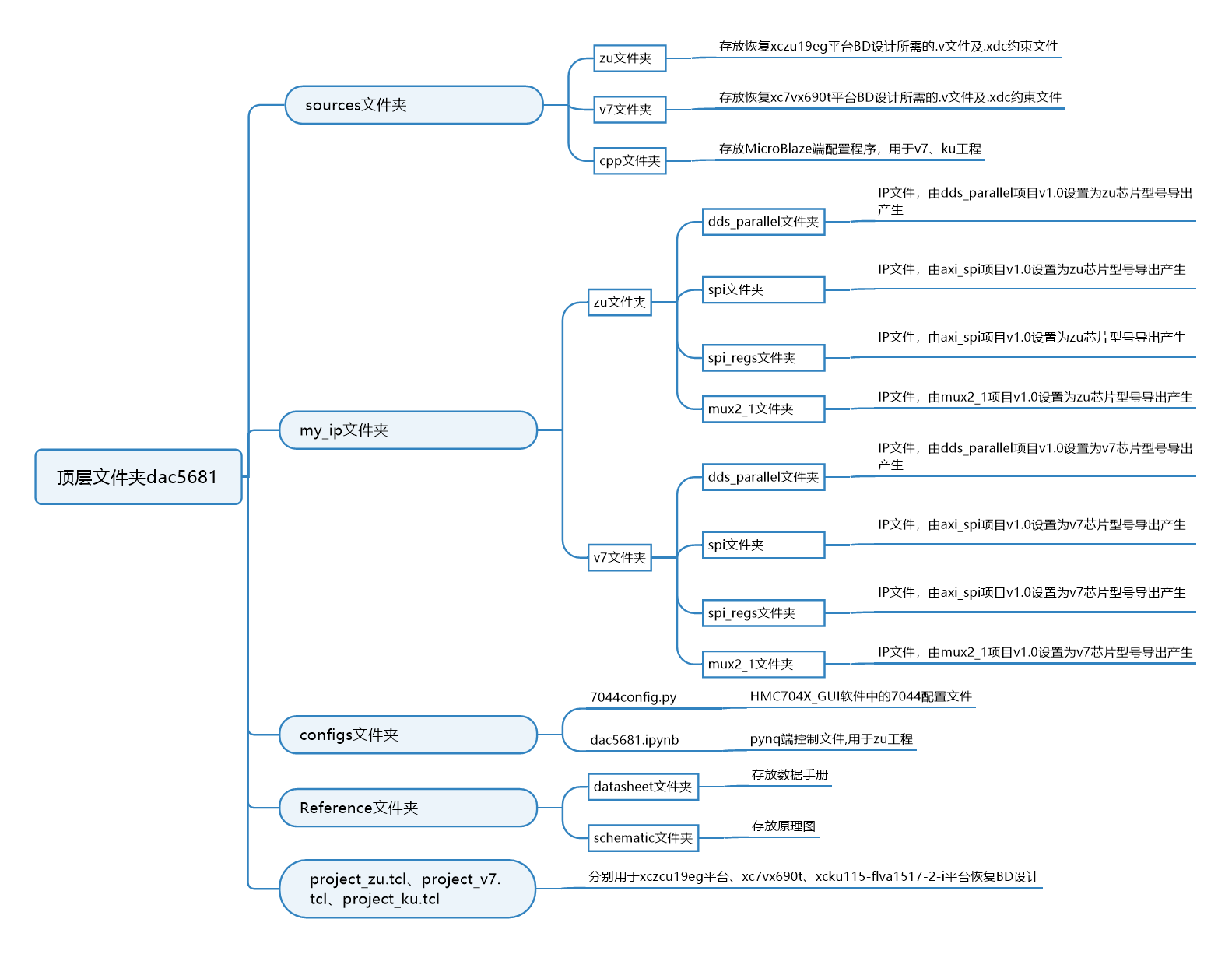
|  |
| --- |
| HMC7044时钟稳定后，通过Reset引脚进行硬复位（低电平复位，复位持续时间>25ns） |
| 依次配置0x01~0x0F寄存器 |
| 通过GPIO开启DCLK |
| 等待1ms |
| 配置0x08寄存器，解除DLL复位 |
| 循环读取0x00寄存器，直至DLL锁定 |
| 通过GPIO开启外置SYNC |
| 通过GPIO开启LVDS数据传输 |

其中，0x0A寄存器（CONFIG10）的值由DAC实际采样率查表得到（对应下表）：

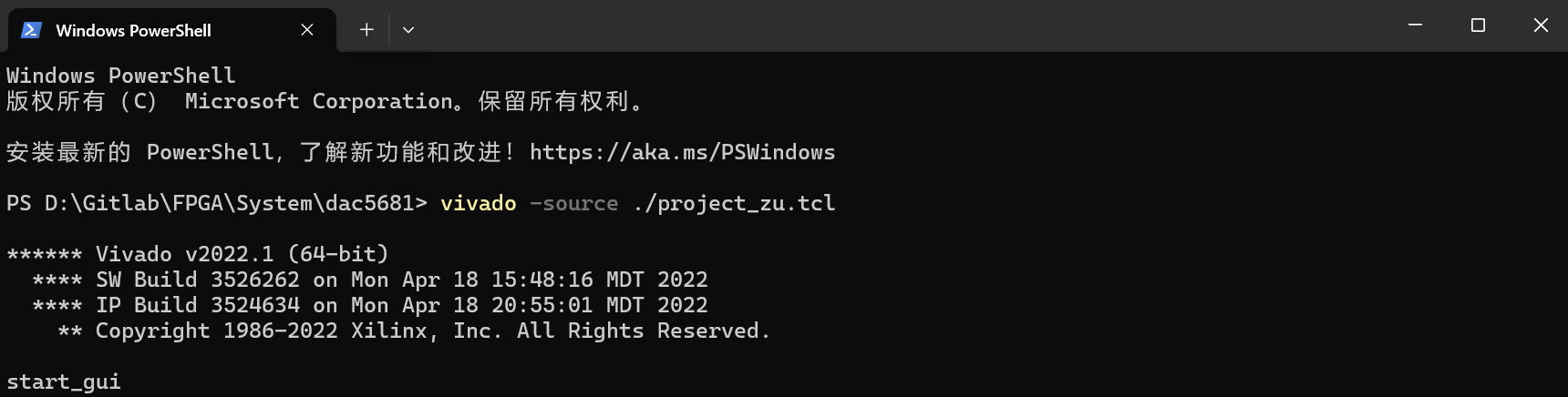


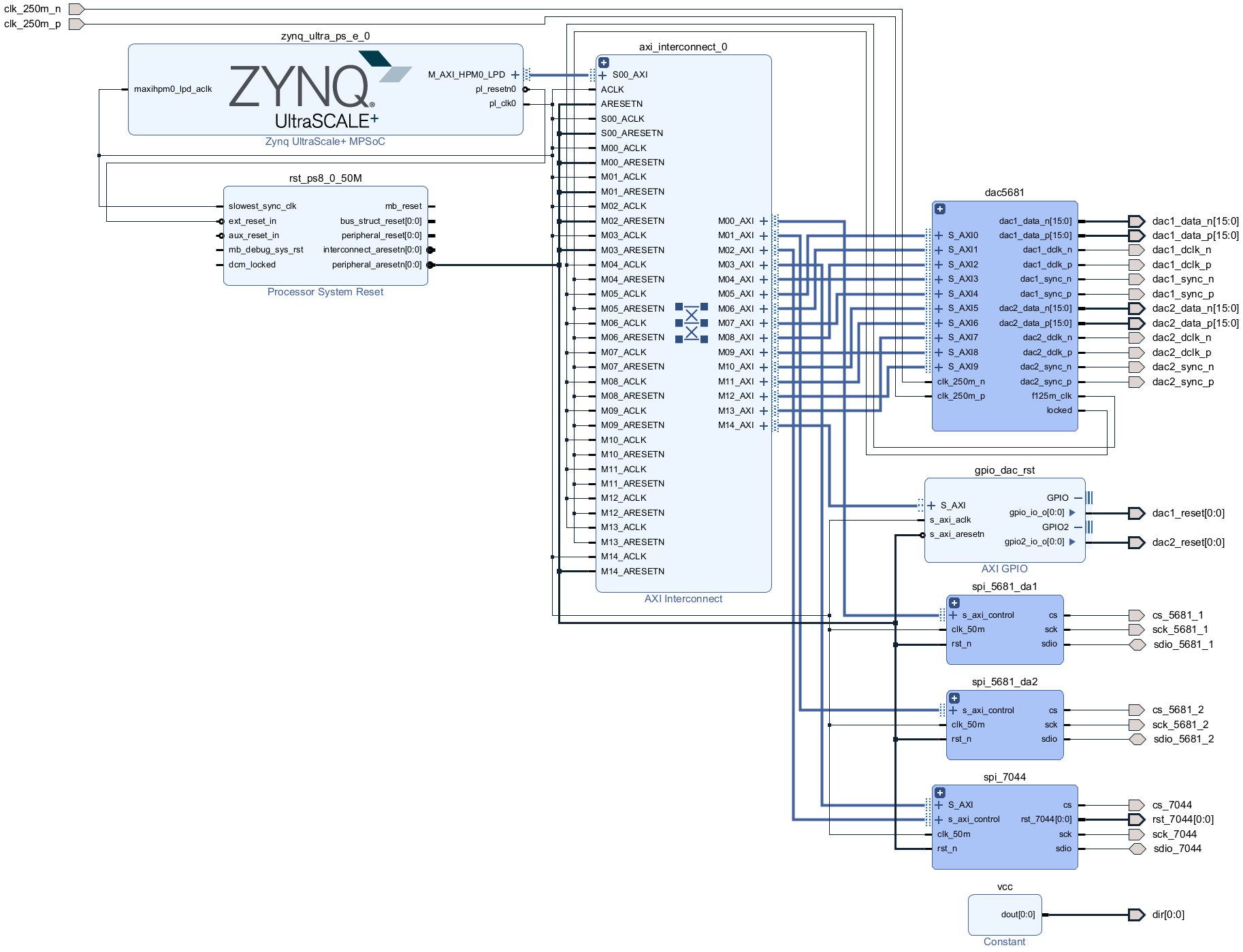
# 五、示例工程及使用方式

本工程的文件结构目录如下图所示（ku工程相关设计文件的存放结构未在下图体现）：



在顶层文件夹路径下运行“vivado -source ./project\_<zu/v7/ku>.tcl”命令，即可在Vivado 2022.1中恢复完整BD设计：





**zu工程中，红框内的dir[0:0]信号用于控制QT7024载板中电平转换芯片的数据流向，与DAC5681驱动无关。**

本设计中，通过多个AXI-GPIO实现系统功能控制，各从机地址如下图所示：



具体控制方式如下：

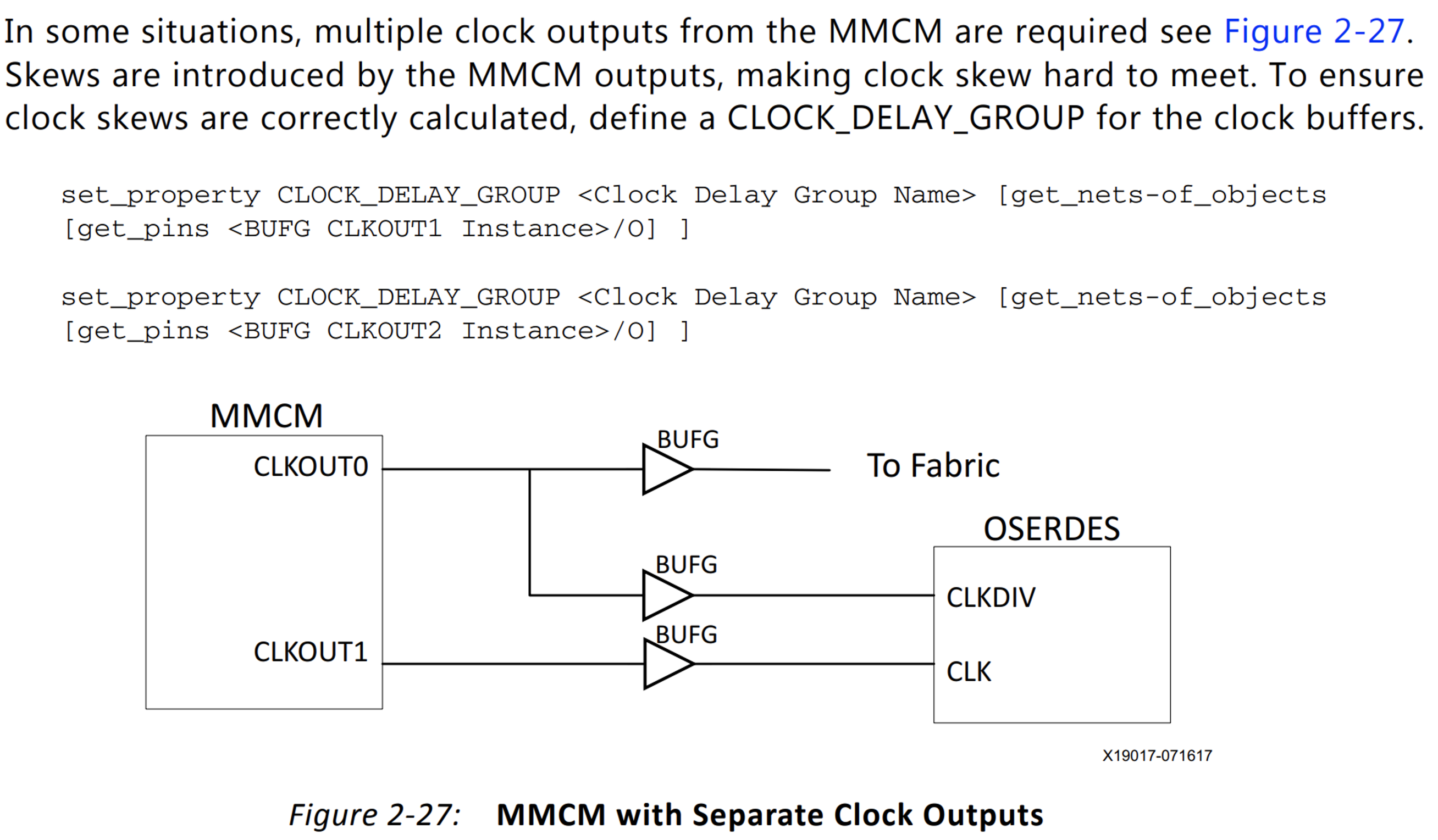
|  |  |  |  |
| --- | --- | --- | --- |
| 基地址 | 内部地址 | 寄存器数据位宽（bit） | 意义 |
| 0x8000\_0000 |  |  | SPI - DAC1 |
| 0x8001\_0000 |  |  | SPI – DAC2 |
| 0x8003\_0000 |  |  | SPI – HMC7044 |
| 0x8002\_0000 | 0x0000 | 1（default value = 1） | HMC7044硬复位（高有效） |
| 0x800e\_0000 | 0x0000 | 1（default value = 0） | DAC1硬复位（低有效） |
| 0x0008 | 1（default value = 0） | DAC2硬复位（低有效） |
| DAC1控制 | | | |
| 0x8004\_0000 | 0x0000 | 16（default value = 0x051F） | DDS1（用于DAC1）频率控制字 |
| 0x0008 | 1（read only） | 该寄存器值为1时，允许更新0x0000寄存器中的频率控制字 |
| 0x8006\_0000 | 0x0000 | 16（default value = 0） | DAC1的用户输入数据源（16bit补码，MSB:LSB=sign, data） |
| 0x0008 | 1（default value = 0） | 该寄存器值为0时，DAC1数据源来自DDS1；  该寄存器值为1时，DAC1数据源来自0x0000寄存器的用户输入（在PL逻辑中，16bit用户输入将被复制8次后拼接为128bit送入lvds\_out模块，也就是每个用户输入将被DAC1输出8次）。 |
| 0x8008\_0000 | 0x0000 | 1（default value = 1） | DAC1-DCLK使能信号（低有效） |
| 0x0008 | 1（default value = 1） | DAC1-数据使能信号（低有效） |
| 0x8007\_0000 | 0x0000 | 2（default value = 0） | {1’b mode，1’b sync }  mode为0时，关闭SYNC信号；  mode为1时，启动SYNC信号，当SYNC信号启动完成后，sync上升沿可触发一次SYNC同步行为。 |
| 0x0008 | 1（read only） | 该寄存器值为1时，标志SYNC信号启动完成 |
| 0x8005\_0000 | 0x0000 | 9（default value = 0） | DAC1-DCLK的odelay延迟值（0-511） |
| 0x0008 | 10（read only） | {9’b delay\_monitor, 1’b delay\_rd}  高9比特用于监测当前的odelay延迟值（采用VAR\_LOAD模式控制延迟，延迟值的步进应小于8）；  最低位为1时，允许更新0x0000寄存器中的延迟值。 |
| DAC2控制 | | | |
| 0x800a\_0000 | 0x0000 | 16（default value = 0x051F） | DDS2（用于DAC2）频率控制字 |
|  | 0x0008 | 1（read only） | 该寄存器值为1时，允许更新0x0000寄存器中的频率控制字 |
| 0x8009\_0000 | 0x0000 | 16（default value = 0） | DAC2的用户输入数据源（16bit补码，MSB:LSB=sign, data） |
| 0x0008 | 1（default value = 0） | 该寄存器值为0时，DAC2数据源来自DDS2；  该寄存器值为1时，DAC2数据源来自0x0000寄存器的用户输入（在PL逻辑中，16bit用户输入将被复制8次后拼接为128bit送入lvds\_out模块，也就是每个用户输入将被DAC2输出8次）。 |
| 0x800c\_0000 | 0x0000 | 1（default value = 1） | DAC2-DCLK使能信号（低有效） |
| 0x0008 | 1（default value = 1） | DAC2-数据使能信号（低有效） |
| 0x800d\_0000 | 0x0000 | 2（default value = 0） | {1’b mode，1’b sync }  mode为0时，关闭SYNC信号；  mode为1时，启动SYNC信号，当SYNC信号启动完成后，sync上升沿可触发一次SYNC同步行为。 |
| 0x0008 | 1（read only） | 该寄存器值为1时，标志SYNC信号启动完成 |
| 0x800b\_0000 | 0x0000 | 9（default value = 0） | DAC1-DCLK的odelay延迟值（0-511） |
| 0x0008 | 10（read only） | {9’b delay\_monitor, 1’b delay\_rd}  高9比特用于监测当前的odelay延迟值（采用VAR\_LOAD模式控制延迟，延迟值的步进应小于8）；  最低位为1时，允许更新0x0000寄存器中的延迟值。 |

推荐按照如下顺序启动（详细代码请见dac5681.ipynb文件）：

|  |
| --- |
| 系统上电，烧录比特 |
| 配置HMC7044时钟输出 |
| 设置DAC数据源（用户输入/内置DDS源切换、用户输入值、DDS源频率控制字） |
| 以内置SYNC模式或外置SYNC模式启动DAC（详情请见“四、DAC5681配置”） |
| 按需调节DAC-DCLK信号的odelay延迟值 |

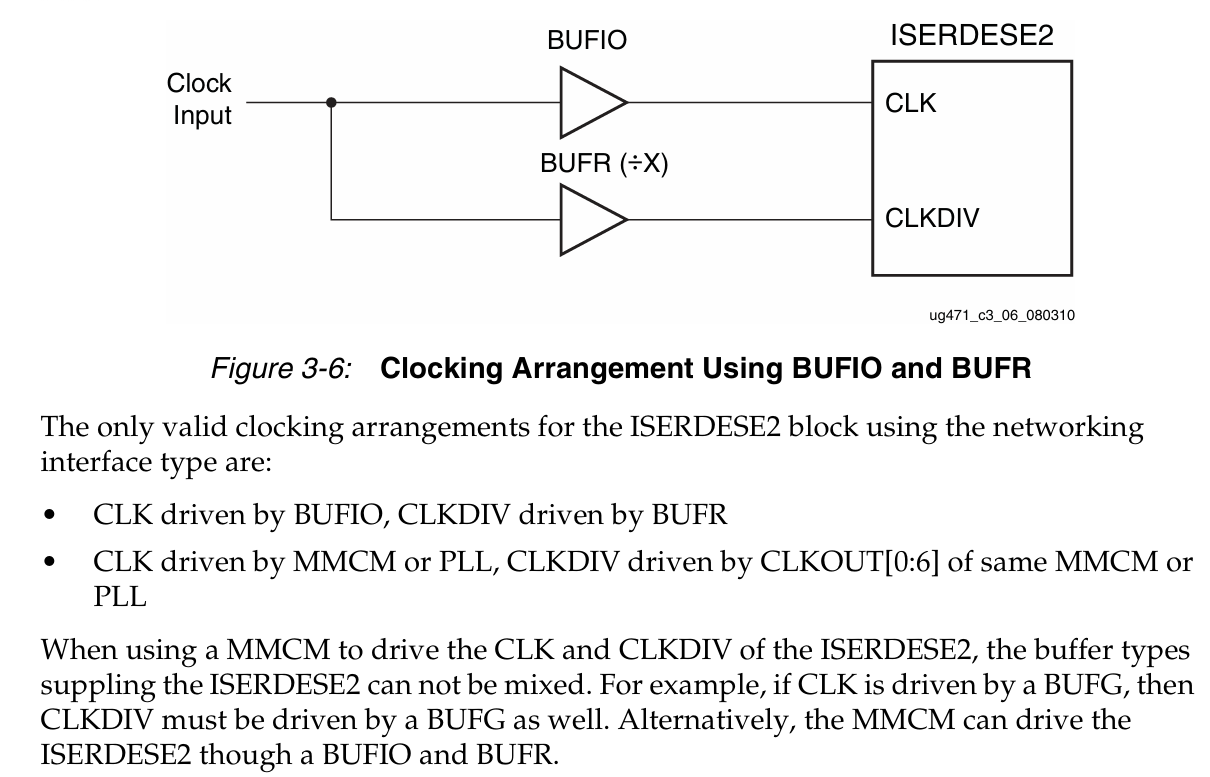
# 附录1：

根据UG571，OSERDESE3的2个时钟输入最好不要通过clk wizard的两个out来产生，而是要将一个out经过BUFG和BUFGCE\_DIV来产生，否则容易引起skew违例：

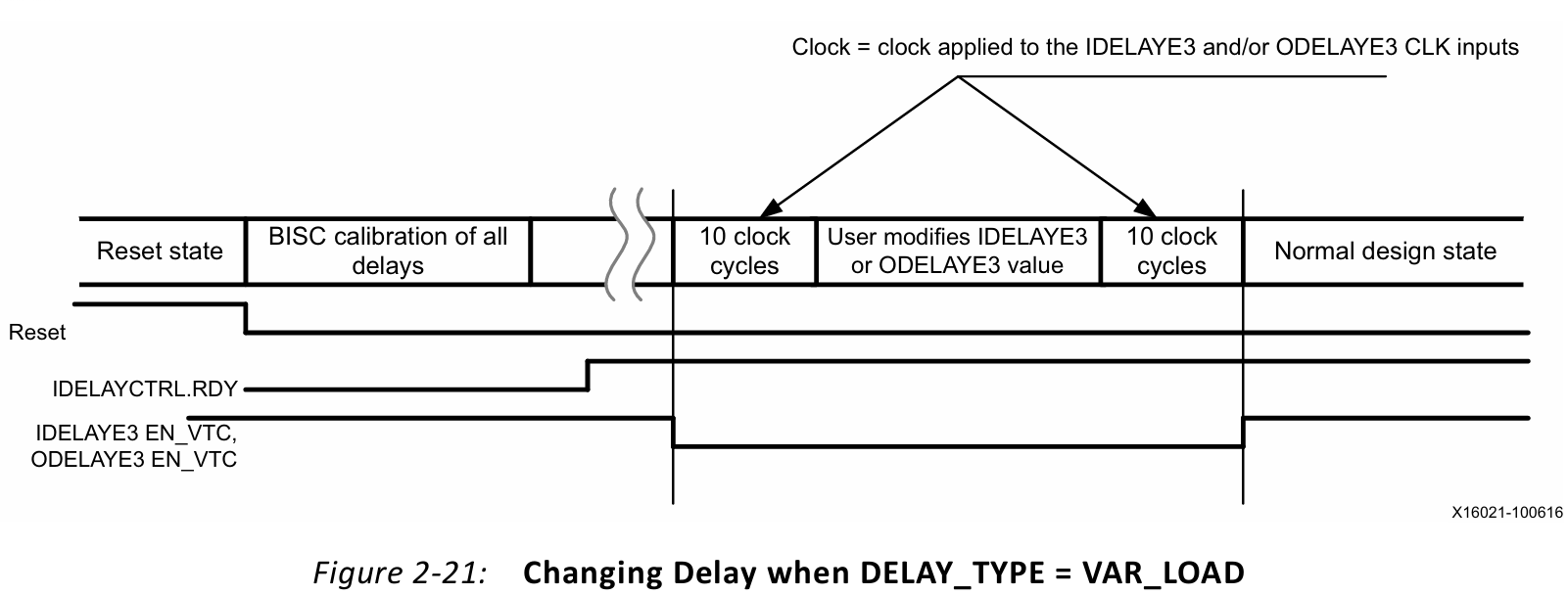
# 附录2：

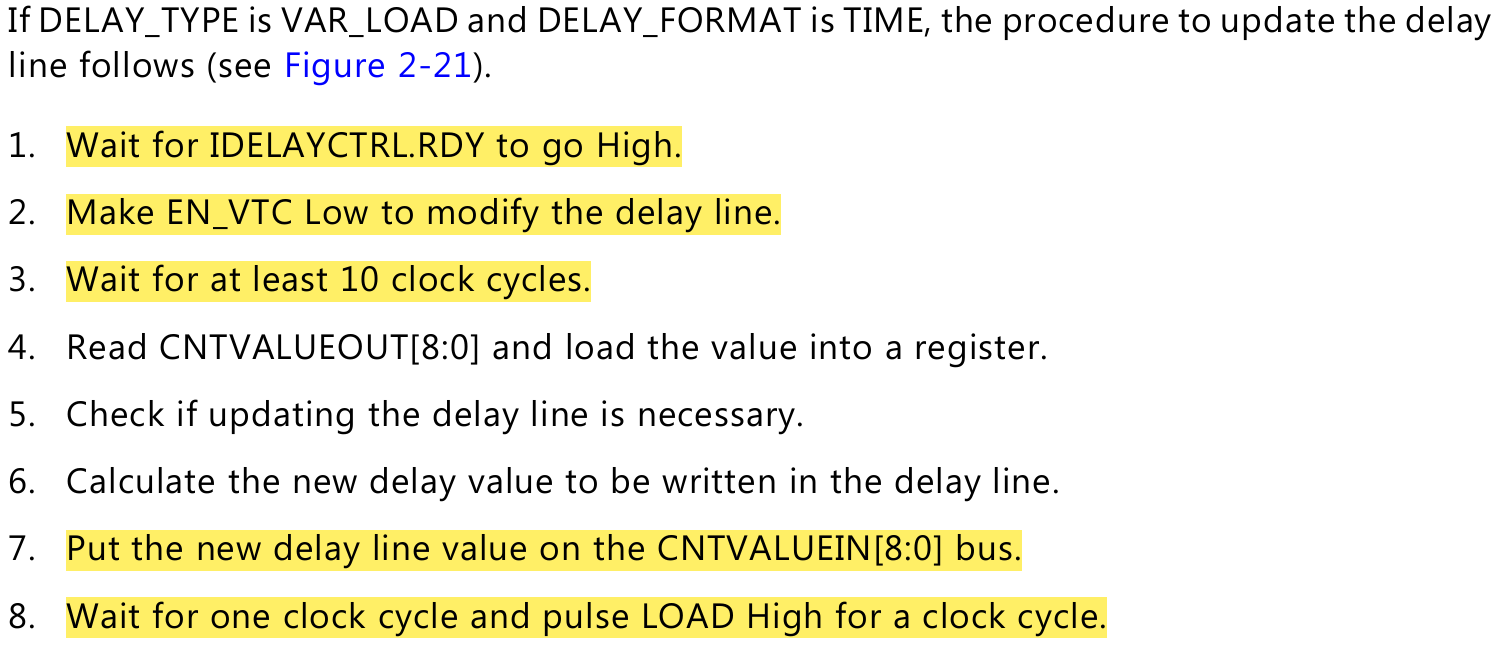
根据UG471，OSERDESE2采用如下时钟输入方式：



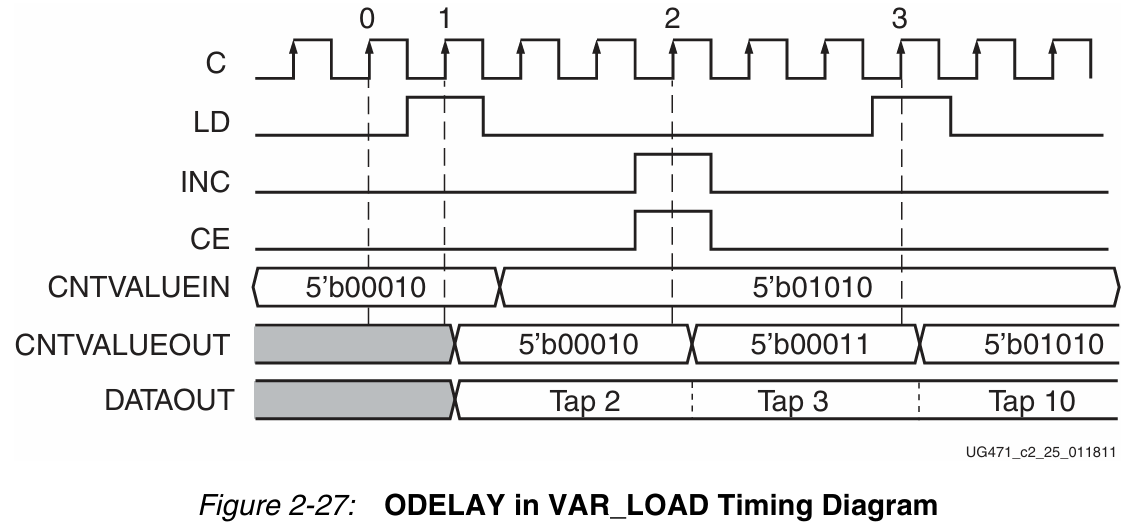
# 附录3：

根据UG571，ODELAYE3在TIME模式下通过VAR\_LOAD调节延迟值的时序如下：



根据UG471，ODELAYE2在TIME模式下通过VAR\_LOAD调节延迟值的时序如下：



可见，相比ODELAYE3，ODELAYE2的调节省去了对VTC信号的操作及等待环节，当LD信号拉高时，CNTVALUEIN传入值生效；因此，ODELAYE3的VAR\_LOAD驱动能够兼容ODELAYE2的VAR\_LOAD模式。