FPGA Note

FPGA 笔记

吾生也有涯，而知也无涯. 以有涯随无涯，殆已！已而为知者，殆而已矣！

整理：Yu Wang

整理时间：May 14, 2018

Email: wyu0725@mail.ustc.edu.cn

Version: 1.0

目　　录

说明

最近看到两篇有意思的 Xilinx White Paper，意识到也该把平时遇到的有意思的设计方面的内容记录下来，内容会慢慢的扩充.

本文档使用的模板来自于 ElegantBook，网址已经停止维护了，但是其中一个作者的博客还在更新Ethlisan，感谢作者提供模板

部分 I

FPGA 介绍

部分 II

Xilinx White Paper

Xilinx 公司有一系列的 White Paper，口气不小，不过 Xilinx 工程师确实是有这个实力，写 White Paper 的工程师定是扫地僧式的人物。

第 1 章 WP 275: Get your Priorities Right - Make your Design Up to 50% Smaller

文档下载地址，一开始学 Verilog 的时候把 Verilog 当成 C 一样的语言来学习，逐步了解到代码的背后是有硬件结构支持，也不是所有的代码都是可综合的，这一点暂时不讨论。这篇文档刷新了认识，代码背后不仅由硬件结构，还有基本的 FPGA 单元. 这篇文档从 LUT 和 D 触发器入手介绍编译器是如何处理代码的。

1.1 基本的逻辑实现

1.1.1 Single Level Logic

首先考虑一个 4 输入与门的实现，代码如下

Verilog

```
always @(posedge clk) begin
  data_out <= a & b & c & d;
end
```

VHDL

```
process (clk)
begin
        if clk' event and clk=' 1'  then
                data_out <= a and b and c and d;
        end if;
end process
```

其实现如1.1所示，使用最简单的一个 D 触发器和一个 4 输入 LUT 即可实现

1.1.2 Two Level Logic

当逻辑门的输入个数增加之后，比如说增加到 6 个之后，LUT 输入的个数显然不够了，这个时候就要使用两个 LUT，如下1.2所示，这种情况无论如何 2 个 LUT 是免不了的。

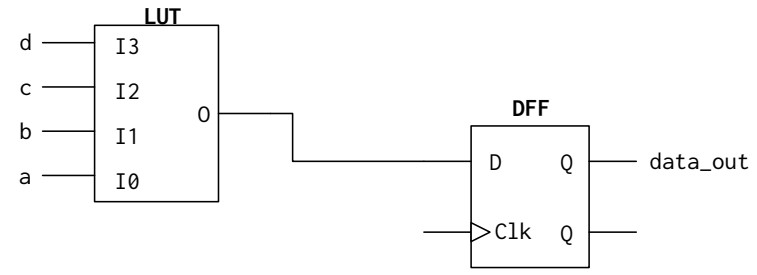

图 **1.1:** Single level logic use 4 input LUT and D Flip-Flop

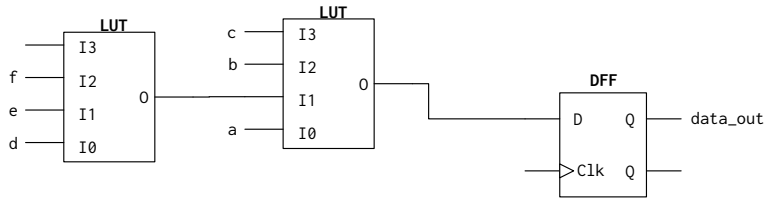

图 **1.2:** Two Level Logic use two 4-input LUT and a D Flip-Flop

1.2 D 触发器中的同步和异步

首先先在如下的 VHDL 代码中添加异步复位的功能异步复位代码

再看下一个例子，添加 Reset 和 Enable 的功能，如下代码所示 Enable 和 Reset 功能的代码。图下面对这种情况进行解释，首先请回忆