

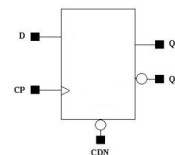
前情回顾

前情回顾

组合电路与时序电路

- **Combination Circuits**是输出仅仅于输入有关，当输入改变时，输出立即改变的电路
- **Sequential Circuit**是指整个电路只在时钟上升沿改变的电路。
- **Sequential Circuit**具有“记忆效应”

PTFDFCNDx
Always-On D Flip-Flop with Async Clear for Footer Type



Truth Table

INPUT			OUTPUT	
CDN	CP	D	Q	QN
0	x	x	0	1
1	0	0	0	1
1	0	1	1	0
1	1	x	Q	QN
1	1	x	Q	QN



H{ 4#xduw

通用异步收发传输器 (Universal Asynchronous Receiver/Transmitter, 通常称为UART)



- **Asynchronous:** 两个系统间的时钟是任意频率/相位关系
 - 时钟: D触发器的时钟
 - 频率: 指时钟震荡的速度
 - 相位: 指相位
- 一个Byte由8个bit组成, 分别是bit7->bit0, 例如0xAC=10101100
- 波特率
- 在uart中, 发送顺序应为00110101
- 分为空闲状态, 起始位, 数据位、停止位



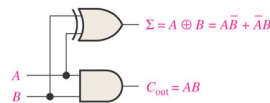
H{ 4#xduw

- | | |
|-------------|-------------|
| • 这次我们将写一个 | • 这次我们还将写一个 |
| • 可以更改发送数据。 | • 可以接收数据 |
| • 可控的发送起始时间 | • 给出接收有效信号 |
| • 可查看的发送状态 | • 可配置波特率 |
| • 可配置的波特率 | • 的接收模块 |
| • 的发射模块 | |

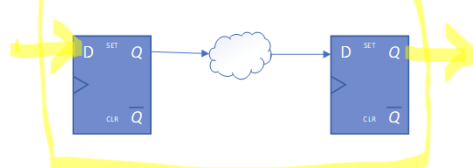
一点点基础知识（二）

数字电路

- 假设我们有与，或，非三种逻辑门
- 根据真值表我们可以写出组合逻辑电路输出与输入的关系



- 就可以构造出不同功能的电路，比如编码器、数据选择器等等。
- 再加上D触发器，可以做成时序逻辑电路。通常时序逻辑电路可以表示成



第四章 组合逻辑电路的分析与设计

组合逻辑电路的分析方法、最小化设计方法、组合逻辑电路的竞争与冒险

第五章 常用组合逻辑电路功能器件

编码器、译码器、数据选择器、分配器、数值比较器、奇偶校验位产生与校验电路、算术运算电路、ALU

第六章 触发器

锁存器、触发器、集成触发器

第七章 时序逻辑电路的分析与设计

时序逻辑电路的基本结构、描述方法，同步时序逻辑电路的分析方法和设计方法，异步电路的分析

第八章 常用的时序逻辑电路模块

寄存器、移位寄存器、计数器集成计数器及应用、序列信号发生器

第九章 半导体存储器

存储器的基本概念、RAM结构和工作原理，常用RAM简介、ROM结构、存储器扩展

第十章 可编程逻辑器件

PLD结构和表示方法、PAL16L8、PAL16R8、GAL结构、CPLD和FPGA结构

第十一章 数模与模数转换

DA基本原理，常用DA方案，DA主要技术指标；AD基本原理、常见方案、主要技术指标

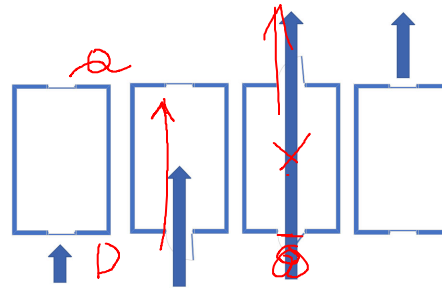
第十二章 数字系统设计

数字系统概念、算法状态机、数字系统设计举例

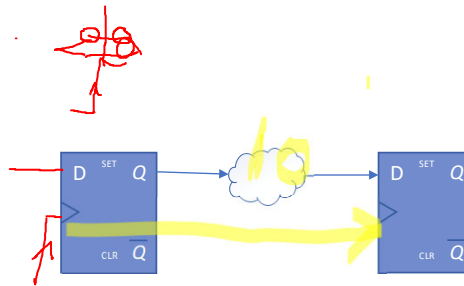


H{4#xduw

- 建立时间与保持时间



- 建立时间->影响系统频率



回到Yhulorj

- 可综合Verilog
 - 可以用物理器件实现的Verilog语言

```
always #5 clk = ~clk;
```

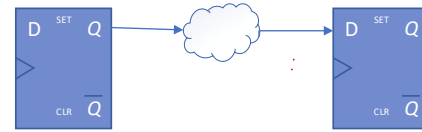
```
always@(posedge clk100M)
  clk50M <= ~clk50M
```

```
always@(posedge clk100M or negedge clk100M)
  clk25M <= ~clk25M
```



回到Yhulorj

- 一点点语法
 - 变量类型 **wire / reg**
 - 基本组成结构 **assign语句和always block**



```
always@(posedge clk or posedge reset)
  if(reset)
    c <= 1'b0;
  else
    c <= a & b;
```

~~always@(a or b)~~
~~c = a & b;~~

~~assign c = a & b;~~

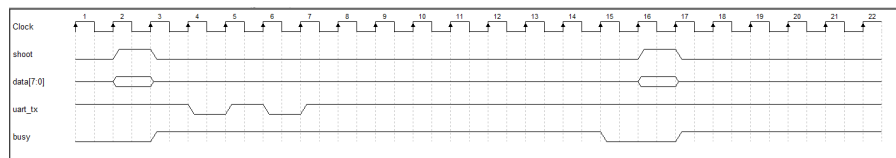
~~always@(*)~~
~~c = a & b;~~

~~always@(*)~~
~~a = a + 1;~~



回到Yhulorj

- 这次我们将写一个
 - 可以更改发送数据。
 - 可控的发送起始时间
 - 可查看的发送状态
 - 可配置的波特率
 - 的发射模块





回到Verilog

- 可选的校验位
 - 奇偶校验, 数1的个数
- Verilog的parameter, localparam和define

```
module uart_tx_op  
#(  
    parameter VERIFY_ON = 1'b0,  
    parameter VERIFY_EVEN = 1'b0  
)
```

```
)  
(  
    input      clk_i,  
    input      resetn_i,  
    input      clk_en_i,  
    input [7:0] datain_i,  
    input      shoot_i,  
    output reg  uart_tx_o = 1'b1,  
    output reg  uart_busy_o  
);
```