LSP Exam – 2015年1月30日

CVUT FEL (CVUT) – České vysoké učení technické v Praze | Czech Technical University in Prague

中文版 | English | Čeština

AI推演版本 – 以下为解析参考

题目1: 逻辑表达式化简

题目: 重写逻辑表达式F，使NOT运算符只出现在变量前面，用卡诺图验证！ **[English]** Rewrite the logic expression F so that NOT operators only appear before variables, verify with Karnaugh map!

```
F = not ( (A or not B or not C) or ( (A or B) and not (A or not C)) )
```

解答步骤

1. 应用德摩根定律: `not(X or Y) = not X and not Y`
2. 逐步化简...

```
F = not(A or B or C) and not((A or B) and not(A or C))
  = Ā·B·C and (not(A or B) or (A or C))
  = Ā·B·C and ((Ā·B) or A or C)
```

继续化简得到最终结果...

卡诺图验证

F	C=0	C=1
AB=00		
AB=01		
AB=11		
AB=10		

题目2: RS锁存器电路仿真　常考

题目: 输入A, B, C在时刻t0, t1, t2, t3的值如下，写出Q输出值。 **[English]** Given inputs A, B, C values at times t0, t1, t2, t3 as shown, write the Q output value.

```
A = ..0..|..1..|..1..|..1..|
B = ..0..|..0..|..0..|..1..|
C = ..1..|..1..|..0..|..0..|

t0   t1   t2   t3

Q = _____|_____|_____|_____|
```

假设输入变化之间的间隔足够长，可以忽略门延迟。

题目3: 香农展开　常考

题目: 将题目2中的函数 `Q=f(A,B,C,Q)` 分解为： **[English]** Decompose the function Q=f(A,B,C,Q) from question 2 into:

```
Q = (not Q and f0(A,B,C)) or (Q and f1(A,B,C))
```

写出f0和f1的卡诺图。

解题方法

1. 令Q=0，求f0
2. 令Q=1，求f1
3. 画卡诺图

题目4: 有符号/无符号数位值　常考

题目: 10位二进制数 `10 0000 1111` 的十进制值是多少？ **[English]** What is the decimal value of the 10-bit binary number 10 0000 1111?

a) **无符号数(unsigned)**: _____

b) **有符号数(two's-complement)**: _____

解答

```
10 0000 1111 (  )

a)   :
   = 2^9 + 2^3 + 2^2 + 2^1 + 2^0
   = 512 + 8 + 4 + 2 + 1
   = 527

b)   ( ):
     1
    1: = -2^9 + (   )
       = -512 + 15 = -497

    2:   1
       01 1111 0000 + 1 = 01 1111 0001 = 497
          = -497
```

题目5: 等价逻辑函数　常考

题目: 标记所有与其他函数等价的逻辑函数： **[English]** Mark all logic functions that are equivalent to other functions:

```
f1 <= (A xor C) or (A and not C);
f2 <= (B or C) and (not A or B or C);
f3 <= ((C and not B) or (B and A));
f4 <= (A or C) and (not A or not C);
```

```
f5 <= (A and not B) xor (A and C);
f6 <= (A and not C) or (C and not A);
```

解题方法

画出每个函数的卡诺图：

f1: (A xor C) or (A and not C) = A or (A xor C) = A or C‾C…

f4: (A or C) and (not A or not C) = A⊕C

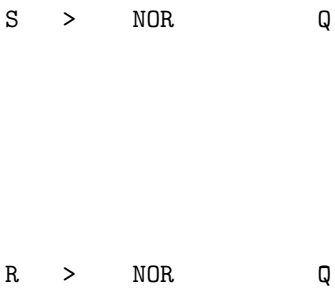
f6: (A and not C) or (C and not A) = A⊕C

结论: f4 ≡ f6 (都是 A XOR C)

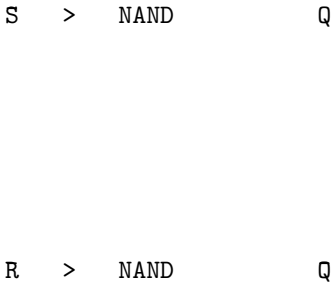
题目6: RS锁存器画图　常考

题目：仅使用NOR门画出RS锁存器，仅使用NAND门画出RS锁存器。 **[English]** Draw RS latch using only NOR gates, and draw RS latch using only NAND gates.

NOR型 RS锁存器 (高电平有效)



NAND型 RS锁存器 (低电平有效)



题目7: 异步18分频器设计

题目: 在未完成的图中添加门和连线，创建CLK时钟信号的**18分频**异步分频器，带异步复位ACLRN。 **[English]** Add gates and connections to the incomplete diagram to create an asynchronous divide-by-18 divider of CLK clock signal with asynchronous reset ACLRN.

设计思路

- 18分频需要计数0–17，共18个状态
- 17的二进制: 10001
- 检测到17时复位所有触发器

关键连接

- 使用5个D触发器级联
- 用AND门检测状态10001
- 检测输出连接到所有DFF的CLRN

题目8: VHDL代码分析

题目: 分析格式错误的VHDL代码，画出对应的逻辑电路图。 **[English]** Analyze the poorly formatted VHDL code and draw the corresponding logic circuit diagram.

```
library IEEE; use IEEE.STD_LOGIC_1164.all;
entity test20140214 is port (a, b, c, d : in std_logic; e : out std_logic); end;
architecture rtl of test20140214 is begin
process(a, b) variable z:std_logic_vector(0 to 3); begin
if b = '0' then z:=(others=>'0');
elsif rising_edge(a) then
  if c='1' then z:=d & z(0 to 2);
  else z:=z(3) & z(0 to 2);
  end if;
end if;
e<=z(3);
end process;
end rtl;
```

格式化代码

```
library IEEE;
use IEEE.STD_LOGIC_1164.all;

entity test20140214 is
    port (a, b, c, d : in std_logic;
          e : out std_logic);
end;

architecture rtl of test20140214 is
begin
    process(a, b)
        variable z: std_logic_vector(0 to 3);
    begin
```

```
        if b = '0' then
            z := (others => '0');      --
        elsif rising_edge(a) then
            if c = '1' then
                z := d & z(0 to 2);    --    d
            else
                z := z(3) & z(0 to 2); --
            end if;
        end if;
        e <= z(3);
    end process;
end rtl;
```

功能分析

- **电路名称**: 4位可控移位寄存器
- **a**: 时钟信号 (上升沿触发)
- **b**: 异步清零 (b='0'时清零)
- **c**: 模式控制
  - c='1': 串行输入模式，d进入z(0)
  - c='0': 循环移位模式
- **e**: 输出z(3)

知识点总结

题号	题型	难度
1	逻辑表达式化简	
2	RS锁存器仿真	
3	香农展开	
4	有符号/无符号数	
5	等价函数判断	
6	RS锁存器画图	
7	分频器设计	
8	VHDL分析	