

LSP 英文考试样题 (含答案)

考试信息

- 文件: Eng_SampleOfExam.pdf
 - 语言: 英语
 - 内容: 往年英文考试样题及解答
-

第 1 题 - De Morgan 展开 (4 分)

题目: 将 XOR 转换为只含 AND、OR、NOT 的形式 [English] Convert XOR to form containing only AND, OR, NOT

$$F1 = A \text{ xor } (B \text{ or } C)$$

$$F1 \leq (A \text{ and not } B \text{ and not } C) \text{ or } (\text{not } A \text{ and } (B \text{ or } C));$$

$$F2 = \text{not } (A \text{ and } (B \text{ xor } C))$$

$$F2 \leq \text{not } A \text{ or } ((\text{not } B \text{ or } C) \text{ and } (B \text{ or not } C));$$

第 2 题 - RS 锁存器仿真 (5 分)

输入序列:

A = 1|1|0|0|1
B = 0|1|1|1|1
C = 0|0|0|1|1
t0|t1|t2|t3|t4

答案:

X = 0|1|1|1|0
Y = 1|0|0|1|1

第 3 题 - Shannon 展开 (6 分)

题目: 将 $X=f(A,B,C,X)$ 分解为 [English] Decompose $X=f(A,B,C,X)$ into

$$X = (\text{not } X \text{ and } f(A,B,C)) \text{ or } (X \text{ and } f(A,B,C))$$

答案 (卡诺图):

f0:	B			f1:	B			
	A	0	1		A	0	1	
C 0	1	0	1	0	1	0	1	1
1	0	1	0	1	1	1	0	1

第 4 题 - 二进制数解释 (2 分)

题目: 12 位数 1000 0000 0111 的十进制值 [English] Decimal value of 12-bit number 1000 0000 0111

计算: - 二进制: $1000\ 0000\ 0111 = 2^{11} + 7 = 2048 + 7 = 2055$

答案: - a) unsigned: 2055 - b) signed: -2041 (2055 - 4096 = -2041)

第 5 题 - 等价逻辑函数 (6 分)

```
f1 <= ((C and not B) or (C and B and A));
f2 <= (A xor C) or (A and not C);
f3 <= (A or B) and (not A or B or C);
f4 <= (not A or not C) and (C or A);
f5 <= (not C and A) or (not A and C);
f6 <= (A and not B) xor (A and not B and C);
```

答案: f4 f5 (都是 A XOR C)

第 6 题 - 主从 D 触发器 (7 分)

题目: 用 4 个 NAND 门和 3 个 NOT 门构建主从 D 触发器 [English] Build a master-slave D flip-flop using 4 NAND gates and 3 NOT gates

结构:

DATA	NAND1	NAND2	NAND3	Q
Clock		NOT		
	NAND4			

说明: - 主锁存器: Clock 高电平时透明 - 从锁存器: Clock 低电平时透明 - 整体在 Clock 上升沿采样

第 7 题 - 上下计数器接线 (15 分)

题目: 机械传感器检测产品通过, Remove 按钮移除产品 [English] Mechanical sensor detects product passing, Remove button removes product

需要添加的元件: 1. 上拉电阻 (10-100kΩ) 2. 去抖动电容 (0.047μF) 3. 限流电阻 (10kΩ)

电路要点: - 传感器 → Plus1 (计数 +1) - Remove 按钮 → Minus1 (计数-1) - 需要消抖处理

第 8 题 - VHDL 代码分析 (15 分)

代码:

```
library ieee; use ieee.std_logic_1164.all;
entity XXX is port(A, B, C, D : in std_logic; Q : out std_logic); end;
architecture rtl of XXX is
    constant E: std_logic := '0';
begin
    process(A, D)
        variable qv: std_logic;
    begin
        if D='0' then
            qv := E; -- 异步清零
```

```

    elsif rising_edge(A) then
        if C='1' then
            qv := B;  -- 加载
        else
            qv := not qv;  -- 翻转
        end if;
    end if;
    Q <= qv;
end process;
end rtl;

```

答案: 带异步清零的 T 触发器/可加载触发器

功能: - D='0': 异步清零 - C='1': 加载 B - C='0': 输出翻转 (T 触发器行为)

2021 年 12 月 21 日考试附加题

第 1 题 - 等价逻辑函数 (8 分)

```

y1 <= (D and not C) or (not C and A) or (D and B);
y2 <= (D and C) xor (B and A);
y3 <= (D or A) and (not C or B) and (D or not C);
y4 <= (D and B) or (D and not C and not B) or ((D xor A) and not C);

```

答案: y1 y3

第 4 题 - 用 NAND 实现 XOR (3 分)

$A \text{ XOR } B = ((A \text{ NAND } (A \text{ NAND } B)) \text{ NAND } (B \text{ NAND } (A \text{ NAND } B)))$

需要 4 个 NAND 门

第 5 题 - 全加器 (3 分)

$\text{Sum} = A \oplus B \oplus \text{Cin}$
 $\text{Cout} = (A \cdot B) + (\text{Cin} \cdot (A \oplus B))$

第 6 题 - +1 加法器 (3 分)

$Y0 = \text{NOT } A0$
 $Y1 = A1 \text{ XOR } A0$
 $Y2 = A2 \text{ XOR } (A1 \text{ AND } A0)$
 $Y3 = A3 \text{ XOR } (A2 \text{ AND } A1 \text{ AND } A0)$

第 7 题 - CMOS 电机控制 (10 分)

题目: 一次按下开启电机, 再次按下关闭电机, 上电默认关闭 [English] One press turns motor on, another press turns it off, default off at power-up

需要添加: 1. DFF 的 D 连接到 Q (Toggle 功能) 2. 按钮到 CLK (通过去抖动电路) 3. 上电复位电路到 CLR_N 4. 继电器控制电路

第 8 题 - VHDL 代码分析 (10 分)

代码:

```

library IEEE; use IEEE.STD_LOGIC_1164.all;
entity xxx is port (a, b, c : in std_logic; d : out std_logic); end;
architecture rtl of xxx is

```

```

begin
  process(a, b, c)
    variable z: std_logic_vector(0 to 3);
  begin
    if a = '0' then
      z := "0001";
    elsif rising_edge(b) then
      z := z(1 to 3) & z(0); -- 左循环移位
      if c='1' then
        d <= not z(3);
      else
        d <= z(3);
      end if;
    end if;
  end process;
end rtl;

```

答案：带预置的 4 位环移位寄存器

功能：- a='0': 预置为"0001"- b 上升沿：左循环移位 - c: 控制输出是否取反

重要知识点总结

1. De Morgan 定律

$\text{NOT}(A \text{ AND } B) = (\text{NOT } A) \text{ OR } (\text{NOT } B)$
 $\text{NOT}(A \text{ OR } B) = (\text{NOT } A) \text{ AND } (\text{NOT } B)$
 $A \text{ XOR } B = (A \text{ AND NOT } B) \text{ OR } (\text{NOT } A \text{ AND } B)$
 $A \text{ XNOR } B = (A \text{ AND } B) \text{ OR } (\text{NOT } A \text{ AND NOT } B)$

2. 12 位有符号数

- 范围: -2048 ~ 2047
- 转换: $\text{val} > 2047 \rightarrow \text{val} - 4096$

3. 用 NAND 实现基本门

- $\text{NOT } A = A \text{ NAND } A$
- $A \text{ AND } B = (A \text{ NAND } B) \text{ NAND } (A \text{ NAND } B)$
- $A \text{ OR } B = (A \text{ NAND } A) \text{ NAND } (B \text{ NAND } B)$
- $A \text{ XOR } B = \text{需要 4 个 NAND 门}$

4. 全加器公式

$\text{Sum} = A \oplus B \oplus \text{Cin}$
 $\text{Cout} = (A \cdot B) + (\text{Cin} \cdot (A \oplus B))$
 $\quad = (A \cdot B) + (A \cdot \text{Cin}) + (B \cdot \text{Cin}) \quad \text{-- MAJ 函数}$

5. +1 加法器规律

$Y[i] = A[i] \text{ XOR } (A[i-1] \text{ AND } A[i-2] \text{ AND } \dots \text{ AND } A[0])$

即：当前位翻转，当且仅当所有低位都是 1