

HW2 -- ICS

PB22111679 孙婧雯

T1

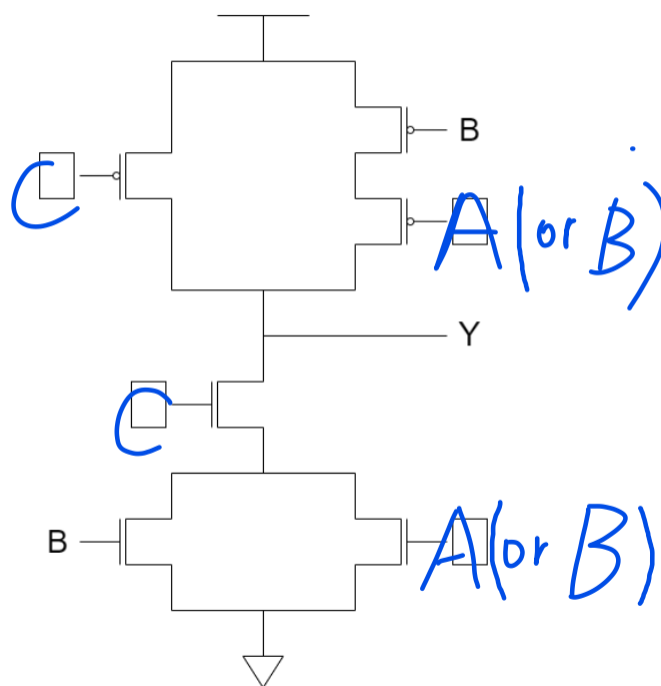
本题使用 CMOS 搭建晶体管电路，并要求我们根据已知的真值表信息补全 transistor 的信息和真值表。

由 $A = 0, B = 0, C = 1, Y = 1$ 可以推测可能的标记方式，此时要求电路的下半部分不能导通，则距 Y 最近的 N-type transistor 应标记 C，电路右下方的 N-type transistor 应标记 A 或 B。

由 $A = 1, B = 1, C = 0, Y = 1$ 可以验证上面分析正确，另外要求此时 Y 与高电平连接，则电路左上部分的 P-type transistor 应标记为 C。

再返回第一种真值情况验证得，右上部分的 P-type transistor 应标记为 A 或 B。

从而补全电路图和真值表如下：



A	B	C	Y
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1

A	B	C	Y
1	1	1	0

T2

根据逻辑表达式运算规则有：

$$\overline{A} = A \text{ NAND } A$$

$$AB = \overline{\overline{AB}} = \overline{A \text{ NAND } B}$$

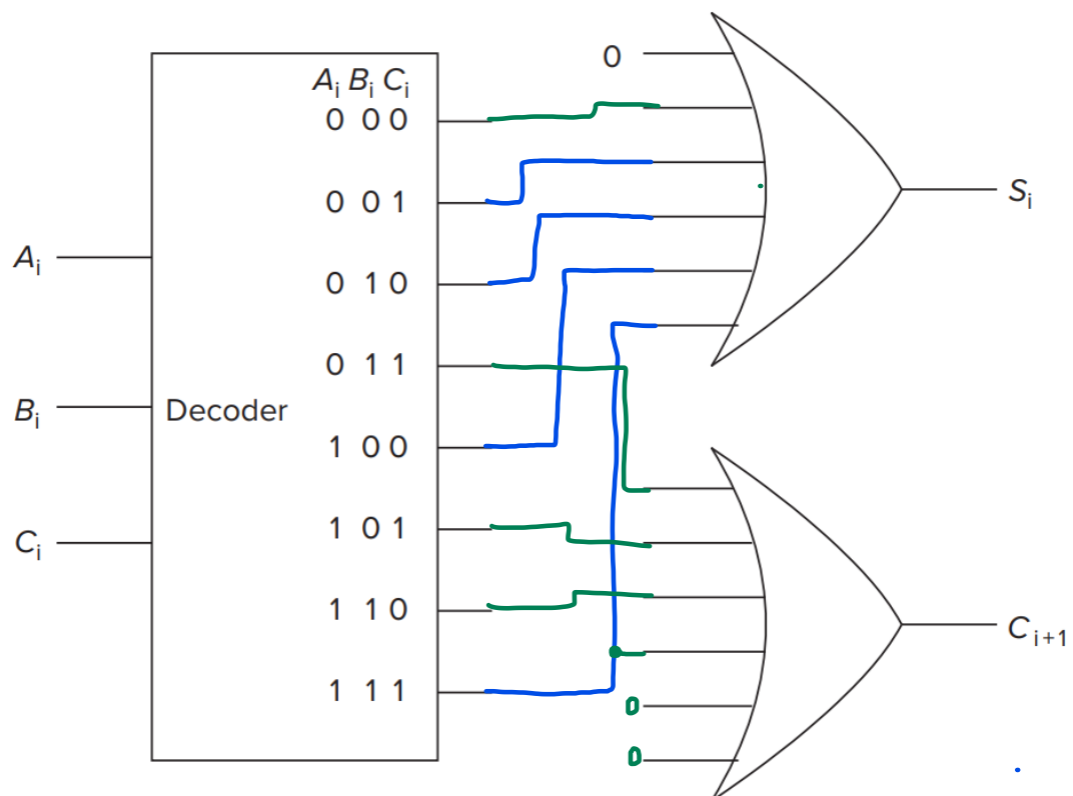
$$A + B = \overline{\overline{A} \overline{B}}$$

首先用与非门表示非门，再用非门和与非门表示与门，最后用与门和非门表示或门。这说明与门、或门、非门都可以用与非门表示。

而与、或、非门（的组合）是逻辑完备的，因此与非门是逻辑完备的。

T3

将运算后有进位的 (A, B, C) 连接到下方的或门，将运算后本位为 1 的连接到上方的或门。如图所示：



T4

1. maximum value = 3 (11 in binary)
2. maximum value = 3 (11 in binary)

3. maximum possible value = 9
4. number of required bits is 4, since 9 is 1001 in binary
5. the truth table is as follows:

A[1]	A[0]	B[1]	B[0]	Y[3]	Y[2]	Y[1]	Y[0]
0	0	x	x	0	0	0	0
0	1	x	y	0	0	x	y
1	0	0	0	0	0	0	0
1	0	0	1	0	0	1	0
1	0	1	0	0	1	0	0
1	0	1	1	0	1	1	0
1	1	0	0	0	0	0	0
1	1	0	1	0	0	1	1
1	1	1	0	0	1	1	0
1	1	1	1	1	0	0	1

6.

$$\begin{aligned}
 Y[2] &= A_1 \overline{A_0} B_1 + A_1 A_0 B_1 \overline{B_0} + B_1 \overline{B_0} A_1 + B_1 B_0 A_1 \overline{A_0} \\
 &= A_1 \overline{A_0} B_1 + B_1 \overline{B_0} A_1 \\
 &= (A_1 \text{ AND } (\text{NOT } A_0) \text{ AND } B_1) \\
 &\quad \text{OR} \\
 &\quad (B_1 \text{ AND } (\text{NOT } B_0) \text{ AND } A_1)
 \end{aligned}$$

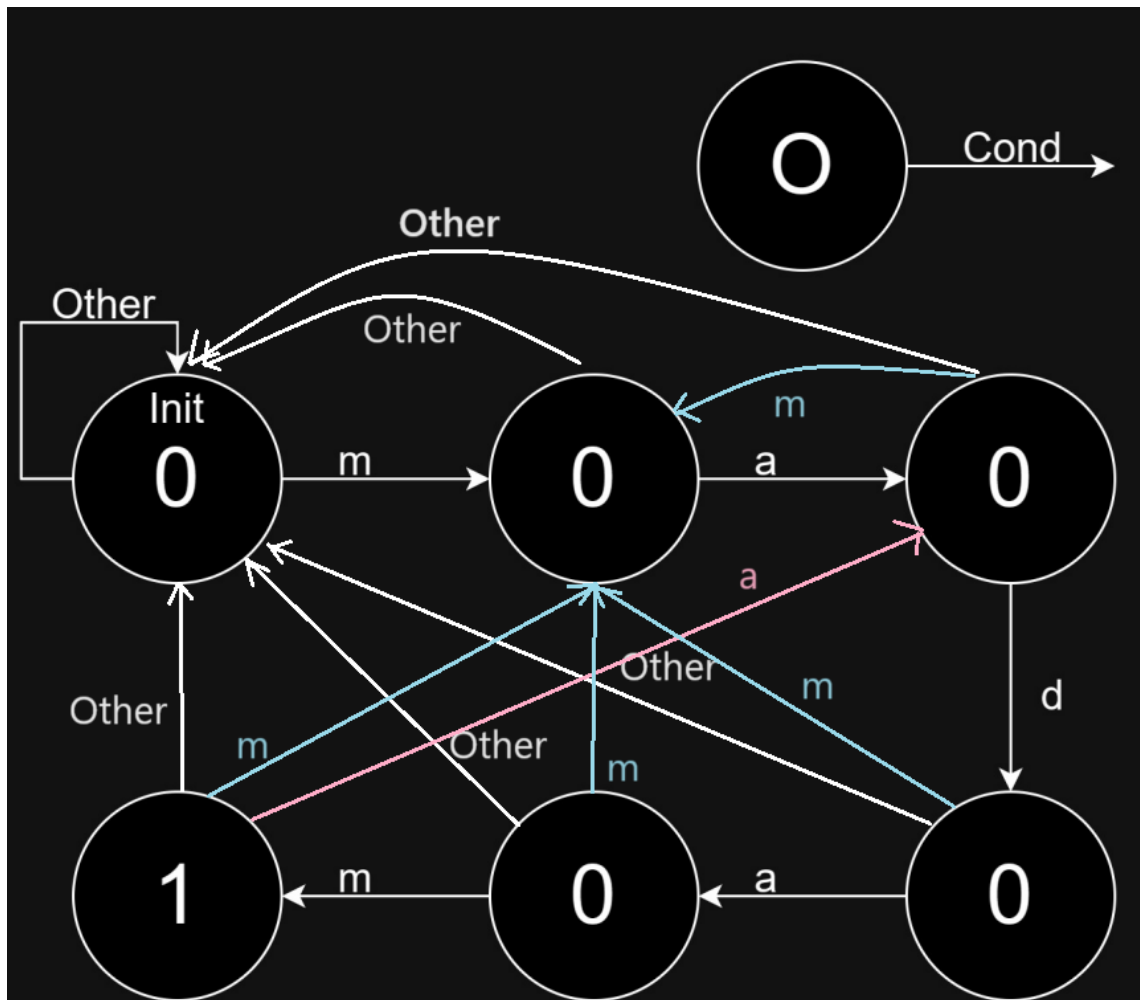
T5

$$\begin{aligned}
 a \text{ XOR } b &= a\overline{b} + \overline{a}b \\
 &= (a + b)(\overline{a} + \overline{b}) \\
 &= (a + b)\overline{ab} \\
 &= \overline{\overline{a \cdot \overline{ab}} + \overline{b \cdot \overline{ab}}} \\
 &= \overline{\overline{a \cdot \overline{ab}} \cdot \overline{b \cdot \overline{ab}}} \\
 &= (a \text{ NAND } (a \text{ NAND } b)) \text{ NAND } (b \text{ NAND } (a \text{ NAND } b))
 \end{aligned}$$

T7

1. 分析：对 m a d a m 五个状态，若下一个为 others 则都回到 `init`（这些转移线用白色标注）；若下一个为 m，除第二个 a 之外都回到第一个 m（这些转移线用蓝色标注；若在第二个 m 后为 a 则回到第一个 a 之后（因为此时 ma 后缀与前缀相同，这条转移线用粉色标注）。

补全状态图如下：



2. 至少使用 5 个，以存储 5 个不同的时序状态。

T8

1. 由于该内存中所有的内存一共 2^a 个地址 (address)，且寻址能力 (addressability) 为 b ，则至多需要 $Sp = \lceil 2^a / b \rceil$ (上取整) 寻址空间 (address space)。

$$address \ space = \lceil \frac{2^a}{b} \rceil$$

2. 至多可以存储 2^a bit。

T9

1. 向第一个位置写数据，则 $A[1:0] = 00$ ，使能信号 $WE = 1$ 。
2. 增加寻址能力到 k bit，反映到图中，应将 D 的列数增加到 k 列。
3. $2^8 < 257 < 2^9$ ，则 PC 位宽为 9，原来地址总线宽度为 3，现在需要增加 6。增加后寻址能力为 $2^9 = 512$ 。

T10

1. 共有 $100 * 4 * 240 * 101 = 9696000$ 个状态，对 2 取对数并上取整可得至少需要 24 bit。
2. 分别对 100、4、240、100 取 2 的对数 (上取整)，相加得至少需要 24 bit。

3. 第二种表示方法更好，这使得 4 种需要表示的状态量之间相互独立，且对储存空间的浪费较小。

备注

T8、T9 的最后一问觉得有些问题，应该是概念还没理解清楚.....qwq