

# ICS HomeWork-3

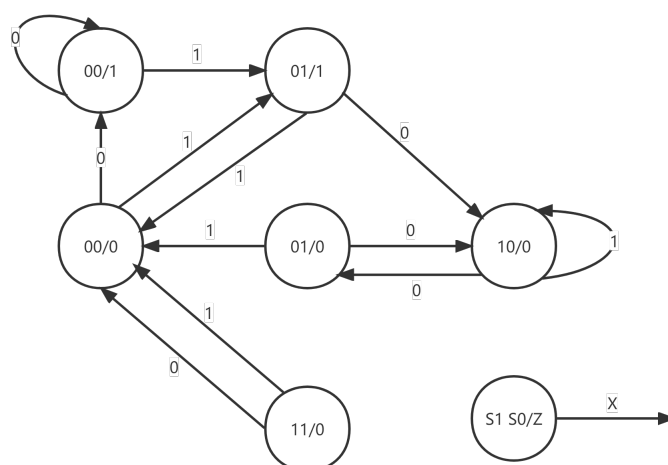
PB20000113 孔浩宇

October 31, 2022

**T1**

如图

$S1$	$S0$	$X$	$Z$	$S1'$	$S0'$
0	0	0	1	0	0
0	0	1	1	0	1
0	1	0	0	1	0
0	1	1	0	0	0
1	0	0	0	0	1
1	0	1	0	1	0
1	1	0	0	0	0
1	1	1	0	0	0



## T2

如果  $R_1 + R_2 \geq 0$ , 则下条指令执行  $x3039$ .

## T3

- (a)  $Bits[15:12]$  表示指令的操作码, 即指令做什么
- (b)  $Bits[11:0]$  表示操作数, 即指令对谁操作

## T4

- (1) 取指令: 需要读取内存中数据, 即  $MDR < -M[MAR]$ , 需 100 个时钟周期。
- (2) 译码: 从 IR 寄存器中读  $IR[15:12]$ , 需要 1 个时钟周期
- (3) 地址计算: 不需要进行
- (4) 取操作数: 从 IR 寄存器中读取, 需 1 个时钟周期
- (5) 执行: 需 1 个时钟周期
- (6) 存放结果: 将结果写入寄存器  $R_6$ , 需要 1 个时钟周期

共需  $100+1+1+1+1=104$  个周期。

## T5

操作数需要 6 位来表示, 寄存器地址需要 6 位, 则立即数至多为 14 位, 表示范围为  $-2^{13} \sim 2^{13} - 1$ .

## T6

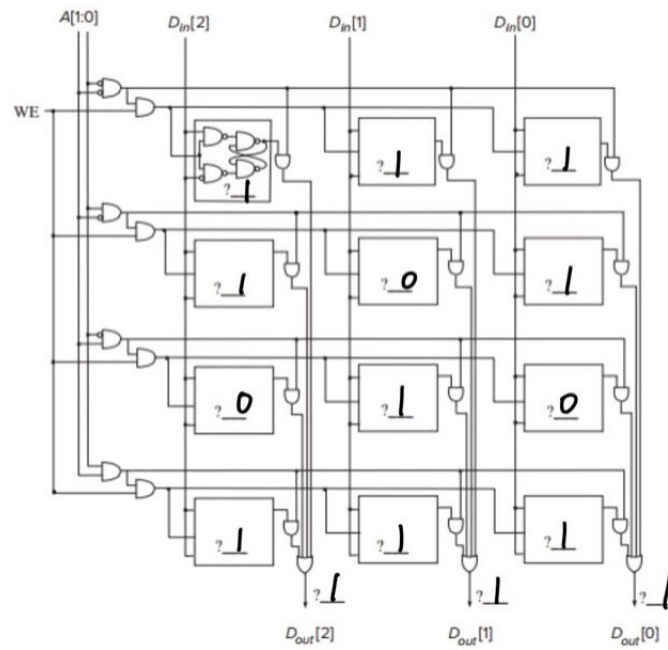
- (a) 对 1001(*NOT*) 没有好处; 对 0001, 0101 可以增加立即数表示位数, 扩大立即数表示范围.
- (b) 对 0010(*LD*), 0011(*ST*), 偏移量位数增加, 可寻址范围增加.
- (c) 对 0000(*BR*) 没有好处.

## T7

	fetch instruction	decode	evaluate address	fetch data	execute	store result
PC	All				JMP	
IR	All					
MAR	All		STR	All		
MDR	All		STR	All		

## T8

如图



## T9

(a)  $MAR : x2$       $MDR : 01010000$

(b)  $MDR : 00111001$

## T10

	R/W	MAR	MDR
Operation 1	W	x4000	1 1 1 1 0
Operation 2	R	x4003	1 0 1 1 0
Operation 3	W	x4001	1 0 1 1 0
Operation 4	R	x4002	0 1 1 0 1
Operation 5	W	x4003	0 1 1 0 1

## T11

(a) 至少需要 8 位来表示操作码

(b) 至少需要 7 位来表示寄存器

(c) 最大位数为 3 位

## T12

(a) 每秒产生的机器周期数为

$$n = \frac{1}{T} = \frac{1}{2 \times 10^{-9}} = 5 \times 10^8.$$

(b) 每秒处理的指令最多为

$$n = \frac{5 \times 10^8}{8} = 6.25 \times 10^7.$$

(c) 每秒能执行指令

$$n = 6 \times 6.25 \times 10^7 = 3.75 \times 10^8.$$

## T13

- (1) 取指令：将当前 PC 装入 MAR, PC 增量；将地址 MAR 对应的数据  $M[MAR]$  写入 MDR, 并将 MDR 的内容写入 IR 寄存器。
- (2) 译码：通过 IR 寄存器中的前四位来判断该指令是什么操作。
- (3) 地址计算：如果涉及地址的计算，在这一拍进行。
- (4) 取操作数：从 IR 中读取该指令操作的对象。
- (5) 执行：对操作数执行该指令对应的操作。
- (6) 存放结果：将结果写入最后结果对应的寄存器。