Nhat Doan- CSC137 - Chris Grove

***Homework #1***

5-1> a> There is one bit which is an indirect bit

There are 256k words which holds 2^8 (256KB) x 2^10 (1024 bytes/KB) = 2^18→ 18 bits for address code

There are 64 registers which is 2^6→ 6 bits for register

That is, There are 32-1-18-6= 7 bits left for op code

5-2> The difference between direct and indirect bit is:

+The direct bit has one memory reference and it gets the operand from the given address. The direct bit is 0.

+The indirect bit has two memory references: get the address of the operand from the given address and use this address to get the value of the operand.

5-3> a> S2S1S0 = 111b =7d → BUS(7) connect memory to BUS

READ→ read from memory through the AR address register.

LD(IR) → load from bus to IR

Result: IR ← M[AR]

b>S2S1S0 = 110b =6d → BUS(6) connect TR to BUS

LD(PC) → load from bus to PC

Result: PC← TR

c> S2S1S0 = 100b =4d → BUS(4) connect AC to BUS

WRITE → write to memory

LD(DR) → load from bus to DR

Result: DR ← AC , M[AR] ← AC

d> S2S1S0 = 000b =0d → BUS(0) no affecting the BUS

ADD → add DR to AC

Result: AC ← AC + DR

5-4>

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | S2 | S1 | S0 | value | LD of register | Memory | Adder |
| a | 0 | 1 | 0 | 2 | AR | - | - |
| b | 1 | 1 | 1 | 7 | IR | READ | - |
| c | 1 | 1 | 0 | 6 | - | WRITE | - |
| d | 1 | 0 | 0 | 4 | AC,DR | - | DR |

5-5> a)IR ← M[PC]

This requires 2 single clock to read memory from PC to AR and then load that value to register IR

* AR ← PC
* IR ← M[AR].

b) AC ← AC +TR

This require 2 single clock to load TR to DR and then add DR to AC

* DR ← TR
* AC ← AC + DR

c) DR ← DR +AC

+TR ← AC ( Save AC )

+AC ← AC + DR

+ DR ← TR

+ AC ← DR, DR ← AC ( swapback)

5-6>

a) 0001 0000 0010 0100b

→ 1024h

This is a direct add instruction at the address 24

AC ← M[AR] + AC

b)1011 0001 0010 0100b

→ B124h

This is an indirect STA at the address 124

AR ← M[AR]

M[AR] ← AC

c) 0111 0000 0010 0000b

→ 7020h

This is a direct INC

AC ← AC +1

5-7)

The 2 instructions that help to make flip-flop E is

CLE (Clear E )and CME (complement E)

5-9>

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | AC | E | PC | AR | IR |
| initial | A937 | 1 | 21 | - | - |
| CLA | 0 | 1 | 22 | 800 | 7800 |
| CLE | A937 | 0 | 22 | 400 | 7400 |
| CMA | 56C8 | 1 | 22 | 200 | 7200 |
| CME | A937 | 0 | 22 | 100 | 7100 |
| CIR | D49B | 1 | 22 | 80 | 7080 |
| CIL | 526F | 1 | 22 | 40 | 7040 |
| INC | A938 | 1 | 22 | 20 | 7020 |
| SPA | A937 | 1 | 22 | 10 | 7010 |
| SNA | A937 | 1 | 22 | 8 | 7008 |
| SZA | A937 | 1 | 22 | 4 | 7004 |
| SZE | A937 | 1 | 22 | 2 | 7002 |
| HLT | A937 | 1 | 22 | 1 | 7001 |

5-10>

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | IR | AR | DR | AC | PC |
| innital | - | - | - | A937 | 21 |
| ADD | 0083 | 083 | B8F2 | A832 | 22 |
| AND | 1083 | 083 | B8F2 | 6229 | 22 |
| LDA | 2083 | 083 | B8F2 | B8F2 | 22 |
| STA | 3083 | 083 | A937 | A937 | 22 |
| BUN | 4083 | 083 | - | A937 | 083 |
| ISZ | 6083 | 083 | B8F3 | A9F2 | 22 |
| BSA | 5083 | 083 | - | A937 | 084 |

5-11>

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | PC | AR | DR | IR | SC |
| Initial | 7FF | - | - | - | - |
| T0 | 7FF | 7FF | - | - | 1 |
| T1 | 800 | 7FF | - | EA9F | 2 |
| T2 | 800 | A9F | - | EA9F | 3 |
| T3 | 800 | C35 | - | EA9F | 4 |
| T4 | 800 | C35 | FFFF | EA9F | 5 |
| T5 | 800 | C35 | 0000 | EA9F | 6 |
| T6 | 801 | C35 | 0000 | EA9F | 0 |

5-12>

a> The instruction is 932E

b> This instruction is [ADD I 32E], which will add the contents of the memory word whose address is stored in memory location 32E( indirect addressing). So AC will have the value 7EC3+ 8B9F = 0A62