CpE 213
Digital Systems Design

Lecture 5

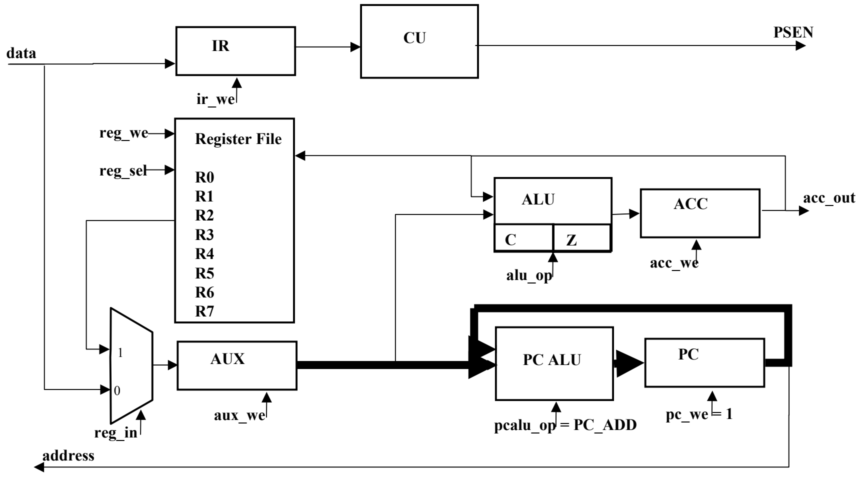
Tuesday 9/9/2003

Before the next lecture

- Review WIMP handout
- Download assignment 2 (due this Friday)
- Read sections 1.11 and 2.1 to 2.4.3 of your textbook.
- Fill in and return index cards.

Execute cycle:

SJMP loop

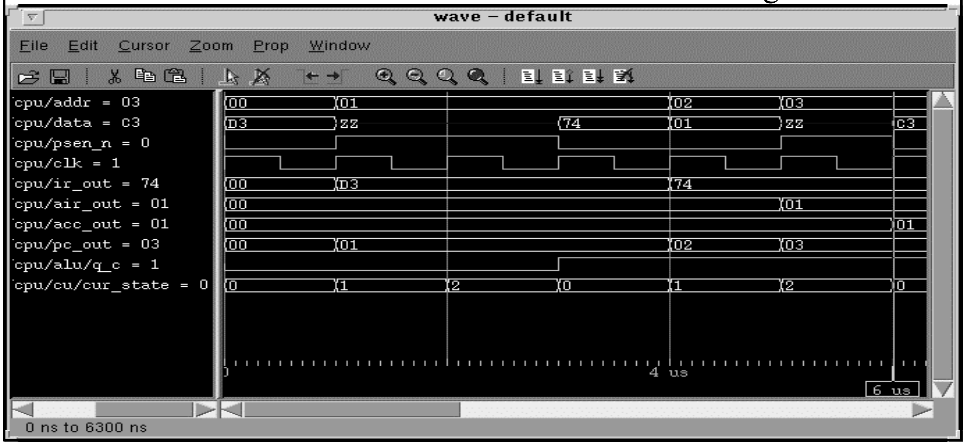


Wimp51 timing

- Registers get loaded by rising edge at end of cycle
- Each instruction lasts three clock cycles
- Each instruction cycle consists of a Fetch, Decode, and Execute clock cycle
- Classic 8051 is similar but more options so more complex (12 or more clock cycles per instruction)

Wimp51 timing diagram

- Two complete instruction cycles are shown
- First is a SETB C (D3h) at location 0000
- PC increments at end of Fetch cycle, C set at end of Exec cycle
- What is second instruction? Value of the data bus during fetch #2?

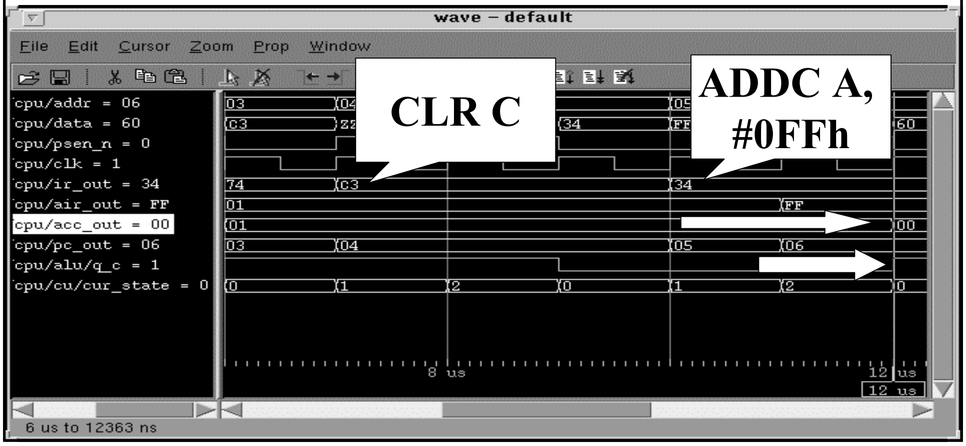


```
cpu/addr = 03
cpu/data = C3
cpu/psen_n = 0
cpu/clk = 1
cpu/ir_out = 74
cpu/air_out = 01
cpu/acc_out = 01
cpu/pc_out = 03
cpu/alu/q_c = 1
cpu/cu/cur_state = 0
```

0 ns to 6300 ns

Wimp51 timing diagram 2

- Two more instructions. What are they?
- Note ACC goes from 1 to 0 at end of #2
- C is set at end of instruction #2



```
cpu/addr = 06
cpu/data = 60
cpu/psen_n = 0
cpu/clk = 1
cpu/ir_out = 34
cpu/air_out = FF
cpu/acc_out = 00
cpu/pc_out = 06
cpu/alu/q_c = 1
cpu/cu/cur_state = 0
```

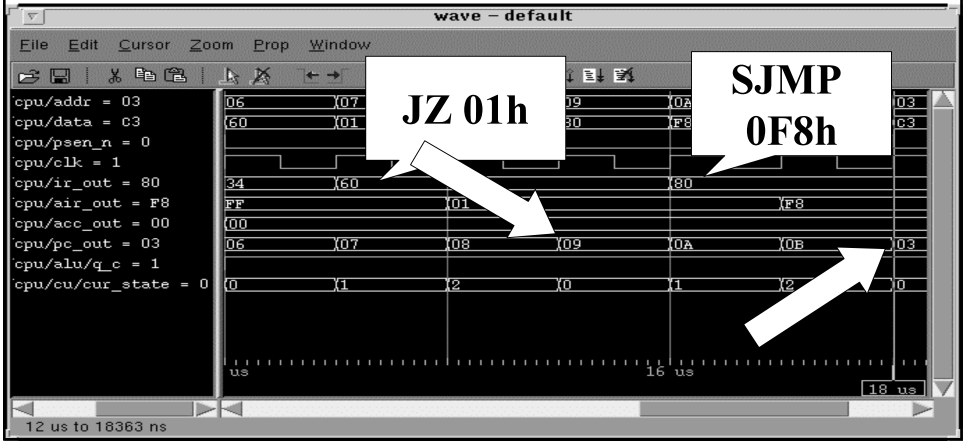
6 us to 12363 ns

CLR C

ADDC A, #0FFh

Wimp51 timing diagram 3

- Two more instructions. What are they?
- First instruction increments PC once more
- Instruction #2 increments PC then changes it to 3h



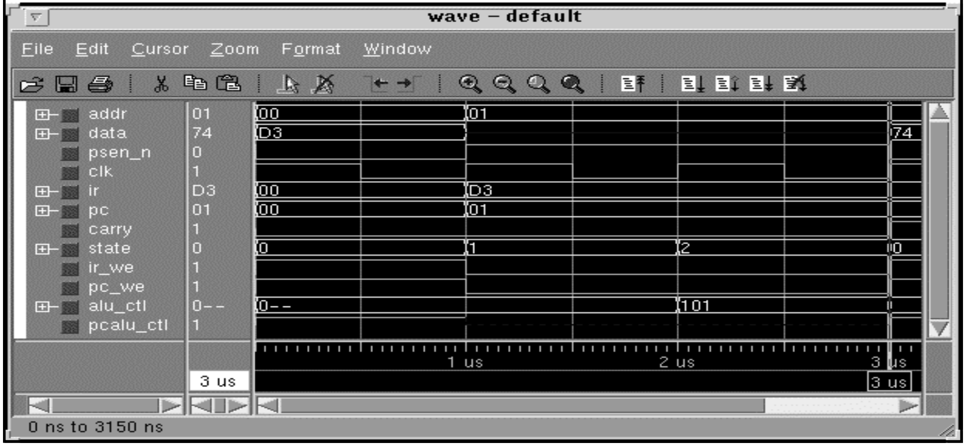
JZ 01h

SJMP 0F8h

```
cpu/addr = 03
cpu/data = C3
cpu/psen_n = 0
cpu/clk = 1
cpu/ir_out = 80
cpu/air_out = F8
cpu/acc_out = 00
cpu/pc_out = 03
cpu/alu/q_c = 1
cpu/cu/cur_state = 0
```

Wimp51 timing diagram 4

- Examine the control signals for SETB C
- WE signals let registers be updated on rising clock edge
- Dashed line represents "don't care" signals



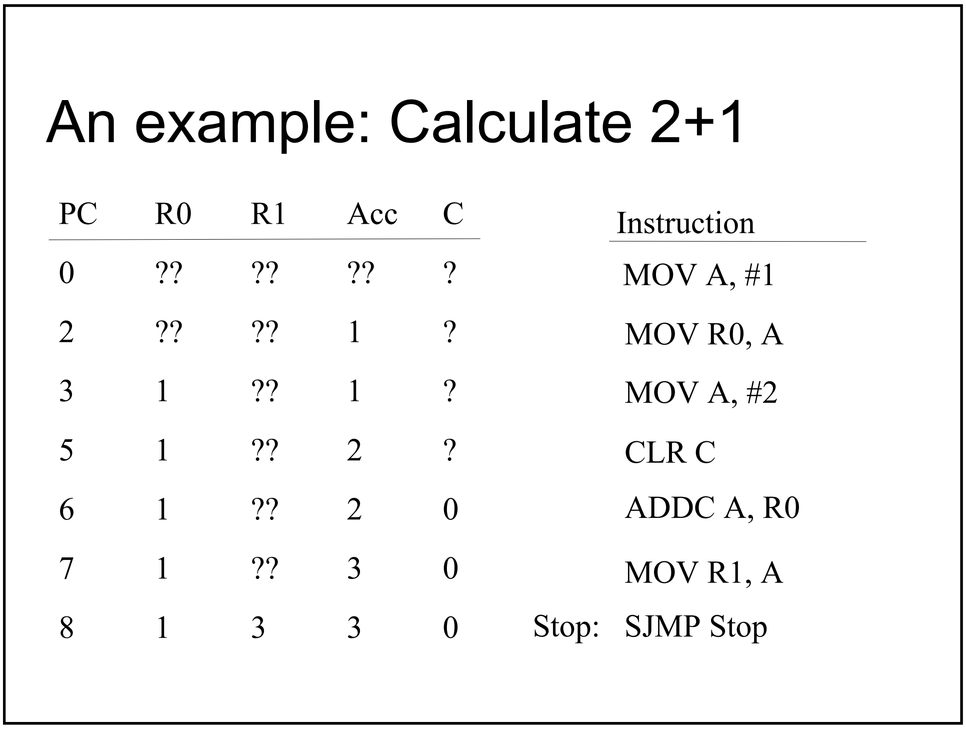
addr	01
data	74
psen_n	0
clk	1
ir	D3
pc	01
carry	1
state	0
ir_we	1
pc_we	1
alu_ctl	0--
pcalu_ctl	1

Wimp51 timing

- Timing diagrams generated by a simulator
- Simulator ‘executes’ a model of Wimp51
- Model is written in VHDL
- VHDL can be used to synthesize Wimp51 hardware
- Simulation models let us ‘try out’ a design before committing to hardware
- We can simulate hardware, software, or both
- Used for rapid prototyping and to get it right the first time

An example: Calculate 2+1

```
Code            Addr          Instr
74  01          0000h         MOV A, #01h
F8              0002h         MOV R0, A
74  02          0003h         MOV A, #02h
C3              0005h         CLR C
38              0006h         ADDC A, R0
F9              0007h         MOV R1, A
80  FE          0008h  Stop:  SJMP Stop
```



An example: Calculate 2+1

PC	R0	R1	Acc	C		Instruction
0	??	??	??	?		MOV A, #1
2	??	??	1	?		MOV R0, A
3	1	??	1	?		MOV A, #2
5	1	??	2	?		CLR C
6	1	??	2	0		ADDC A, R0
7	1	??	3	0		MOV R1, A
8	1	3	3	0	Stop:	SJMP Stop