IB Computer Science

Unit 2

Introduction to Programming and Personal Computing: P88

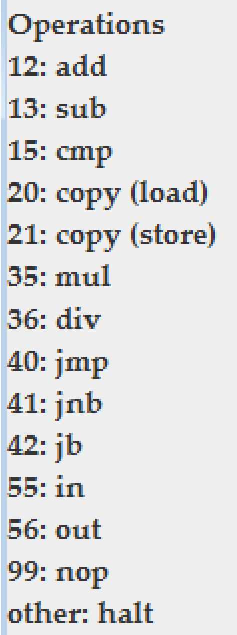
Student_______________

Date: _________________

The P88 Instruction Set

The operation codes (op codes) are listed below on the left.

When you write a program that includes the words that represent the op code you are writing in low-level assembly language. You will then need to translate each assembly language statement into the machine code that can be entered into the P88 simulator. Examples of using assembly language and coding in machine language are provided below.



Operations
12: add
13: sub
15: cmp
20: copy (load)
21: copy (store)
35: mul
36: div
40: jmp
41: jnb
42: jb
55: in
56: out
99: nop
other: halt

The valid instructions for the P88 Simulator are shown on the right.

Remember that most of the action takes place in the AX register.

Getting input from the user:

in prompts the user for a numeric value
and puts it into the AX register;
code this in machine language as
55999 (op code 55, 999 as placeholder)

Sending output to the user

out take the value in the AX register and
copies it to the output box;
code this in machine language as
56999 (op code 56, 999 as paceholder)

Copy a value from a memory location to the AX register

copy (load) 40 takes the value that is stored in
location 40 and copies it to the AX register.
This is the "load" function.
Code this as 20040

Copy the value in the AX register into a memory location

copy (store) 50 copies the value in the AX register
and stores it in memory location 50.
Code this "store" function as 21050

IB Computer Science

Unit 2

Introduction to Programming and Personal Computing: P88

Student_______________

Date: _________________

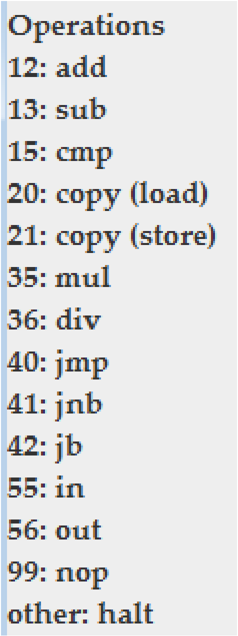
Arithmetic Operations

All of these arithmetic operations are performed on the AX register and a value that is stored in a memory location. The results are stored in the AX register.

Suppose the memory locations 50-59 contained the following values:

12	15	30	40	100	200	3	400	500	600
50	51	52	53	54	55	56	57	58	59

Let the current content of the AX register contain 27

	Pseudo Code	Assembly Code	Machine Code	Results
Operations 12: add 13: sub 15: cmp 20: copy (load) 21: copy (store) 35: mul 36: div 40: jmp 41: jnb 42: jb 55: in 56: out 99: nop other: halt	Addition AX = AX + 15 The value 15 has been stored in memory location 51	Add 51	12051	The AX register contains 27+15=42
	Subtraction AX = AX – 12 The value 12 has been stored in memory location 50	Sub 50	13050	The AX register contains 27-12 = 15
	Multiplication AX = AX * 100 The value 100 has been stored in memory location 54	Mul 54	35054	The AX register contains 27*100 = 2700
	Division AX = AX / 3 The value 3 is stored in memory location 56	Div 56	36056	The AX register contains 27 / 3 = 9

IB Computer Science
Unit 2

Introduction to Programming and Personal Computing: P88

Student_______________

Date: ________________

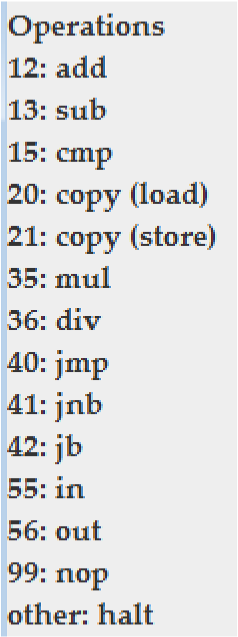
Comparison and Branching Operations

12	15	30	40	100	200	3	400	500	600
50	51	52	53	54	55	56	57	58	59

Let the current content of the AX register contain 10

Suppose that you have a program that is stored in memory locations 00 through 29, and whenever you have some sort of error conditions you want your program to output the contents of the AX register and then stop (halt) the program.

								56999	0
20	21	22	23	24	25	26	27	28	29

	Pseudo Code	Assembly Code	Machine Code	Results
 Operations 12: add 13: sub 15: cmp 20: copy (load) 21: copy (store) 35: mul 36: div 40: jmp 41: jnb 42: jb 55: in 56: out 99: nop other: halt	Output AX Quit	Out halt	56999 0	Displays the value of the AX register in the output box (10).
	GOTO 29	Jmp 29	40029	Goes to instruction 29 – which will ouput the AX register and halt the program

IB Computer Science
Unit 2

Introduction to Programming and Personal Computing: P88

Student_______________

Date: ________________

Comparison and Branching Operations (continued)

12	15	30	40	100	200	3	400	0	600
50	51	52	53	54	55	56	57	58	59

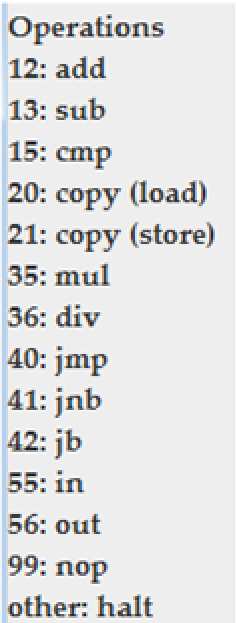
Let the current content of the AX register contain 10

The comparison operation : cmp

The cmp operation will compare the contents of the AX register with the value stored at a specified memory location. **The contents of the AX register are not changed by the cmp operation: the result of the comparison is stored in the CF register.**

```
Assembly Instruction        Operation Performed
CMP  AX,<memory>            If AX<(memory) then
                                  CF = B
                             else
                                 CF = NB
```

After doing a CMP operation you can use JB and JNB to branch to a block of code to process the results of the comparison.

	Pseudo Code	Assembly Code	Machine Code	Results
 Operations 12: add 13: sub 15: cmp 20: copy (load) 21: copy (store) 35: mul 36: div 40: jmp 41: jnb 42: jb 55: in 56: out 99: nop other: halt	IF CF = B Goto 23	Jb 23	42023	IF the CF register is B then the next instruction will be from memory location 23
	IF CF = NB Goto 31	jnb 31	41031	If the CF register is NB then the next instruction will be from memory location 31

The last operation to discuss is the **no-op** operation, code **99.** This instruction does not do anything, but is useful when you don't want to do anything.