

Registers

AX

Accumulator; used to store some calculation results

BX

Base; index register for MOVE

CX

Counter; count for string operations & shifts

DX

Datas; port address for IN and OUT

SP

Points to top of stack

BP

Points to base of stack frame

SI

Points to a source in stream operations

DI

Points to a destination in stream operations

CS

Points to base of segment containing code

DS

Points to base of segment containing datas

SS

Points to base of segment containing the stack

ES

Points to base of an additionnal segment

IP

Points to the next instruction to be run

Loops

LOOP Decrements CX and jumps to label if CX <> 0

LOOPE Decrements CX and jumps to label if CX <> 0 and ZF = 1

LOOPNE Decrements CX and jumps to label if CX <> 0 and ZF = 0

JCXZ Jumps to label if CX = 0

Remember: Loops uses CX and ZF registers

Instructions

Instructi on	Usage	Example
--------------	-------	---------

MOV	Assigns value to register	MOV AX, 15h
-----	---------------------------	-------------

ADD	Adds value to register	ADD AX, BX
-----	------------------------	------------

SUB	Subtracts value to register	SUB AX, 1
-----	-----------------------------	-----------

AND	Executes binary AND operation	AND AL, 1101111b
-----	-------------------------------	------------------

OR	Executes binary OR operation	OR AL, BX
----	------------------------------	-----------

NOT	Executes binary NOT operation	NOT AL
-----	-------------------------------	--------

XOR	Executes binary XOR operation	XOR AL, 01010101b
-----	-------------------------------	-------------------

SHL/SAL /SHR/SA R	Shifts to the left or to the right. When arithmetic, the sign bit doesn't shift. Explused bit is stocked in CF	SHL AL, 1 / SAL AL, 2
-------------------	--	-----------------------

ROL/RC L/ROR/R- CR	Rotates to the left or to the right, using or not CF	ROL AL, 1 / RCR AL, 2
--------------------	--	-----------------------

INC / DEC	Increments or decrements a value	INC AX / DEC BX
-----------	----------------------------------	-----------------

ADC / SBB	Does an addition or subtraction of >16bits numbers, by storing restraint in CF	ADC AX, CX
-----------	--	------------

STC / CLS / CMC	Sets CF to 1, 0 or inverts it	STC
-----------------	-------------------------------	-----

STD / CLD	Sets DF to 1 or 0	STD
-----------	-------------------	-----

STI / CLI	Sets IF to 1 or 0	CLI
-----------	-------------------	-----

Instructions (cont)

MUL / IMUL / DIV / IDIV	Multiplies or divides two numbers and stores it to AX (+DX for most significant bit)	MUL AL, 0001Ah MUL 00002h
-------------------------	--	------------------------------

JMP	Jumps to a label	JMP calc
-----	------------------	----------

CMP	Performs a comparison	CMP AL, 01234b
-----	-----------------------	----------------

PUSH / PUSHA / PUSHF	Pushes a data to SS: [SP] / all registers / FLAGS	PUSH 10h
----------------------	---	----------

POP / POPA / POPF	Restores fatas form SS: [SP] / all registers (except SP) / FLAGS	POP AX
-------------------	--	--------

Conditional jumps

Instruct ion	Description	Condition
JA	Jump if Above	CF = 0 and ZF = 0

JAE	Jump if Above or Equal	CF = 0
-----	------------------------	--------

JB	Jump if Below	CF = 1
----	---------------	--------

JBE	Jump if Below or Equal	CF = 1 or ZF = 1
-----	------------------------	------------------

JC	Jump if Carry	CF = 1
----	---------------	--------

JE	Jump if Equal	ZF = 1
----	---------------	--------

JG	Jump if Greater	ZF = 0 and SF = OF
----	-----------------	--------------------

JGE	Jump if Greater or Equal	SF = OF
-----	--------------------------	---------

JL	Jump if Less	SF <> OF
----	--------------	----------

JLE	Jump if Less or Equal	ZF = 1 or SF <> OF
-----	-----------------------	--------------------

JO	Jump if Overflow	OF = 1
----	------------------	--------

JP	Jump if Parity	PF = 1
----	----------------	--------

JPE	Jump if Parity Even	PF = 1
-----	---------------------	--------

JS	Jump if Sign	SF = 1
----	--------------	--------

JZ	Jump if Zero	ZF = 1
----	--------------	--------



Colors			
Decimal	Hexa	Binary	Color
0	00h	0000b	Black
1	01h	0001b	Blue
2	02h	0010b	Green
3	03h	0011b	Colbalt blue
4	04h	0100b	Red
5	05h	0101b	Violet
6	06h	0110b	Brown
7	07h	0111b	Light grey
8	08h	1000b	Dark grey
9	09h	1001b	Light blue
10	0Ah	1010b	Light green
11	0Bh	1011b	Light cobalt
12	0Ch	1100b	Light red
13	0Dh	1101b	Light violet
14	0Eh	1110b	Yellow
15	0Fh	1111b	White



By **Mikael Peigney** (Mika56)
cheatography.com/mika56/
gamersflag.com/

Published 13th February, 2013.
Last updated 5th June, 2014.
Page 2 of 2.

Sponsored by **CrosswordCheats.com**
Learn to solve cryptic crosswords!
<http://crosswordcheats.com>