

## 6502 Processor Instruction Set

### Load and Store Group

Mnemônico	Operação	Tipos de Instruções	Código Binário	Nro. Bytes	Flags Afetados
<b>LDA</b>	Load Accumulator	LDA \$aaaa	\$AD	3	N,Z
		LDA \$aa	\$A5	2	
		LDA #\$aa	\$A9	2	
		LDA \$aaaa,X	\$BD	3	
		LDA \$aaaa,Y	\$B9	3	
		LDA \$aa,X	\$B5	2	
		LDA (\$aa,X)	\$A1	2	
		LDA (\$aa),Y	\$B1	2	
<b>LDX</b>	Load X Register	LDX \$aaaa	\$AE	3	N,Z
		LDX \$aa	\$A6	2	
		LDX #\$aa	\$A2	2	
		LDX \$aaaa,Y	\$BE	3	
		LDX \$aa,Y	\$B6	2	
<b>LDY</b>	Load Y Register	LDY \$aaaa	\$AC	3	N,Z
		LDY \$aa	\$A4	2	
		LDY #\$aa	\$A0	2	
		LDY \$aaaa,X	\$BC	3	
		LDY \$aa,X	\$B4	2	
<b>STA</b>	Store Accumulator	STA \$aaaa	\$8D	3	none
		STA \$aa	\$85	2	
		STA \$aaaa,X	\$9D	3	
		STA \$aaaa,Y	\$99	3	
		STA \$aa,X	\$95	2	
		STA (\$aa,X)	\$81	2	
		STA (\$aa),Y	\$91	2	
<b>STX</b>	Store X Register	STX \$aaaa	\$8E	3	none
		STX \$aa	\$86	2	
		STX \$aa,Y	\$96	2	
<b>STY</b>	Store Y Register	STY \$aaaa	\$8C	3	none
		STY \$aa	\$84	2	
		STY \$aa,X	\$94	2	

### Arithmetic Group

<b>ADC</b>	Add with Carry	ADC \$aaaa	\$6D	3	N,V,Z,C
		ADC \$aa	\$65	2	
		ADC #\$aa	\$69	2	
		ADC \$aaaa,X	\$7D	3	
		ADC \$aaaa,Y	\$79	3	
		ADC \$aa,X	\$75	2	
		ADC (\$aa,X)	\$61	2	
		ADC (\$aa),Y	\$71	2	
<b>SBC</b>	Subtract with Carry	SBC \$aaaa	\$ED	3	N,V,Z,C
		SBC \$aa	\$E5	2	
		SBC #\$aa	\$E9	2	
		SBC \$aaaa,X	\$FD	3	
		SBC \$aaaa,Y	\$F9	3	
		SBC \$aa,X	\$F5	2	
		SBC (\$aa,X)	\$E1	2	
		SBC (\$aa),Y	\$F1	2	

### Increment and Decrement Group

<b>INC</b>	Increment a memory location	INC \$aaaa	\$EE	3	N,Z
		INC \$aa	\$E6	2	
		INC \$aaaa,X	\$FE	3	
		INC \$aa,X	\$F6	2	
<b>INX</b>	Increment the X register	INX	\$E8	1	N,Z
<b>INY</b>	Increment the Y register	INY	\$C8	1	N,Z
<b>DEC</b>	Decrement a memory location	DEC \$aaaa	\$CE	3	N,Z
		DEC \$aa	\$C6	2	
		DEC \$aaaa,X	\$DE	3	
		DEC \$aa,X	\$D6	2	
<b>DEX</b>	Decrement the X register	DEX	\$CA	1	N,Z
<b>DEY</b>	Decrement the Y register	DEY	\$88	1	N,Z

### Register Transfer Group

<b>TAX</b>	Transfer accumulator to X	TAX	\$AA	1	N,Z
<b>TAY</b>	Transfer accumulator to Y	TAY	\$A8	1	N,Z
<b>TXA</b>	Transfer X to accumulator	TXA	\$8A	1	N,Z
<b>TYA</b>	Transfer Y to accumulator	TYA	\$98	1	N,Z

### Logical Group

<b>AND</b>	Logical AND	AND \$aaaa	\$2D	3	N,Z
		AND \$aa	\$25	2	
		AND # \$aa	\$29	2	
		AND \$aaaa,X	\$3D	3	
		AND \$aaaa,Y	\$39	3	
		AND \$aa,X	\$35	2	
		AND (\$aa,X)	\$21	2	
		AND (\$aa),Y	\$31	2	
		<b>EOR</b>	Exclusive OR	EOR \$aaaa	
EOR \$aa	\$45			2	
EOR # \$aa	\$49			2	
EOR \$aaaa,X	\$5D			3	
EOR \$aaaa,Y	\$59			3	
EOR \$aa,X	\$55			2	
EOR (\$aa,X)	\$41			2	
EOR (\$aa),Y	\$51			2	
<b>ORA</b>	Logical Inclusive OR	ORA \$aaaa	\$0D	3	N,Z
		ORA \$aa	\$05	2	
		ORA # \$aa	\$09	2	
		ORA \$aaaa,X	\$1D	3	
		ORA \$aaaa,Y	\$19	3	
		ORA \$aa,X	\$15	2	
		ORA (\$aa,X)	\$01	2	
		ORA (\$aa),Y	\$11	2	

### Compare and Bit Test Group

<b>CMP</b>	Compare accumulator	CMP \$aaaa	\$CD	3	N,Z,C
		CMP \$aa	\$C5	2	
		CMP #\$aa	\$C9	2	
		CMP \$aaaa,X	\$DD	3	
		CMP \$aaaa,Y	\$D9	3	
		CMP \$aa,X	\$D5	2	
		CMP (\$aa,X)	\$C1	2	
		CMP (\$aa),Y	\$D1	2	
<b>CPX</b>	Compare X register	CPX \$aaaa	\$EC	3	N,Z,C
		CPX \$aa	\$E4	2	
		CPX #\$aa	\$E0	2	
<b>CPY</b>	Compare Y register	CPY \$aaaa	\$CC	3	N,Z,C
		CPY \$aa	\$C4	2	
		CPY #\$aa	\$C0	2	
<b>BIT</b>	Bit Test	BIT \$aaaa	\$2C	3	N,V,Z
		BIT \$aa	\$24	2	

### Shift and Rotate Group

<b>ASL</b>	Arithmetic Shift Left	ASL \$aaaa	\$0E	3	N,Z,C
		ASL \$aa	\$06	2	
		ASL A	\$0A	1	
		ASL \$aaaa,X	\$1E	3	
		ASL \$aa,X	\$16	2	
<b>LSR</b>	Logical Shift Right	LSR \$aaaa	\$4E	3	N,Z,C
		LSR \$aa	\$46	2	
		LSR A	\$4A	1	
		LSR \$aaaa,X	\$5E	3	
		LSR \$aa,X	\$56	2	
<b>ROL</b>	Rotate Left	ROL \$aaaa	\$2E	3	N,Z,C
		ROL \$aa	\$26	2	
		ROL A	\$2A	1	
		ROL \$aaaa,X	\$3E	3	
		ROL \$aa,X	\$36	2	
<b>ROR</b>	Rotate Right	ROR \$aaaa	\$6E	3	N,Z,C
		ROR \$aa	\$66	2	
		ROR A	\$6A	1	
		ROR \$aaaa,X	\$7E	3	
		ROR \$aa,X	\$76	2	

### Jump and Branch Group

<b>JMP</b>	Jump to another location	JMP \$aaaa	\$4C	3	none
		JMP (\$aaaa)	\$6C	3	
<b>BCC</b>	Branch if carry flag clear	BCC aa	\$90	2	none
<b>BCS</b>	Branch if carry flag set	BCS aa	\$B0	2	none
<b>BEQ</b>	Branch if zero flag set	BEQ aa	\$F0	2	none
<b>BMI</b>	Branch if negative flag set	BMI aa	\$30	2	none
<b>BNE</b>	Branch if zero flag clear	BNE aa	\$D0	2	none
<b>BPL</b>	Branch if negative flag clear	BPL aa	\$10	2	none
<b>BVC</b>	Branch if overflow flag clear	BVC aa	\$50	2	none
<b>BVS</b>	Branch if overflow flag set	BVS aa	\$70	2	none

