

# 6502 INSTRUCTIONS

Modes		BEQ NUBDIZC		JSR NUBDIZC		SBC NUBDIZC	
<b>Modes</b>		Branch on Equal		Jump to SubRoutine		Subtract with Carry	
Ac Accumulator		Lb 2+		Ab 6		Im 2 Z 3 Zx 4 Ab 4 Ax 4+ Ay 4+ Ix 6 Iy 5+	
Lb Label LABEL		<b>BRK NUBDIZC</b>		<b>LDA NUBDIZC</b>		<b>STA NUBDIZC</b>	
Im Immediate #s12		BReak		LoAd Accumulator		STore Accumulator	
Z Zero Page s12		Im 7		Im 2 Z 3 Zx 4 Ab 4 Ax 4+ Ay 4+ Ix 6 Iy 5+		Z 3 Zx 4 Ab 4 Ax 5 Ay 5 Ix 6 Iy 6	
Zx Zero Page,X s12,X		<b>CMP NUBDIZC</b>		<b>LDX NUBDIZC</b>		<b>TXS NUBDIZC</b>	
Zy Zero Page,Y s12,Y		CoMPare accumulator		LoAd X register		TranSfer X to Stack	
Ab Absolute s1234		Im 2 Z 3 Zx 4 Ab 4 Ax 4+ Ay 4+ Ix 6 Iy 5+		Im 2 Z 3 Zy 4 Ab 4 Ay 4+		Im 2	
Ax Absolute,X s1234,X		<b>CPX NUBDIZC</b>		<b>LDY NUBDIZC</b>		<b>INC NUBDIZC</b>	
Ay Absolute,Y s1234,Y		ComPare X register		LoAd Y register		INCrement memory	
In Indirect (s1234)		Im 2 Z 3 Ab 4		Im 2 Z 3 Zx 4 Ab 4 Ax 4+		Z 5 Zx 6 Ab 6 Ax 7	
Ix Indirect,X (s12,X)		<b>CPY NUBDIZC</b>		<b>LSR NUBDIZC</b>		<b>TAY NUBDIZC</b>	
Iy Indirect,Y (s12,Y)		ComPare Y register		Logical Shift Right		TranSfer A to Y	
		Im 2 Z 3 Ab 4		A 2 Z 5 Zx 6 Ab 6 Ax 7		Im 2	
<b>ADC NUBDIZC</b>		<b>DEC NUBDIZC</b>		<b>NOP NUBDIZC</b>		<b>TYA NUBDIZC</b>	
ADD with Carry		DECrement memory		No OPeration		TranSfer Y to A	
Im 2 Z 3 Zx 4 Ab 4 Ax 4+ Ay 4+ Ix 6 Iy 5+		Z 5 Zx 6 Ab 6 Ax 7		Im 2		Im 2	
<b>AND NUBDIZC</b>		<b>EOR NUBDIZC</b>		<b>ORA NUBDIZC</b>		<b>DEV NUBDIZC</b>	
bitwise AND with accumulator		bitwise Exclusive OR		bitwise OR with Accumulator		DEcrement Y	
Im 2 Z 3 Zx 4 Ab 4 Ax 4+ Ay 4+ Ix 6 Iy 5+		Im 2 Z 3 Zx 4 Ab 4 Ax 4+ Ay 4+ Ix 6 Iy 5+		Im 2 Z 3 Zx 4 Ab 4 Ax 4+ Ay 4+ Ix 6 Iy 5+		Im 2	
<b>ASL NUBDIZC</b>		<b>CLC NUBDIZC</b>		<b>TAX NUBDIZC</b>		<b>INX NUBDIZC</b>	
Arithmetic Shift Left		CLear Carry		TranSfer A to X		INCrement Y	
A 2 Z 5 Zx 6 Ab 6 Ax 7		Im 2		Im 2		Im 2	
<b>BIT NUBDIZC</b>		<b>SEC NUBDIZC</b>		<b>TXA NUBDIZC</b>		<b>TSX NUBDIZC</b>	
test BITS		SEt Carry		TranSfer X to A		TranSfer Stack to X	
Z 3 Ab 4		Im 2		Im 2		Im 2	
<b>BPL NUBDIZC</b>		<b>CLI NUBDIZC</b>		<b>DEX NUBDIZC</b>		<b>PHA NUBDIZC</b>	
Branch on PLus		CLear Interrupt		DEcrement X		Push Accumulator	
Lb 2+		Im 2		Im 2		Im 3	
<b>BMI NUBDIZC</b>		<b>SEI NUBDIZC</b>		<b>INX NUBDIZC</b>		<b>PLA NUBDIZC</b>	
Branch on MInus		SEt Interrupt		INCrement X		Pull Accumulator	
Lb 2+		Im 2		Im 2		Im 4	
<b>BUC NUBDIZC</b>		<b>CLV NUBDIZC</b>		<b>ROL NUBDIZC</b>		<b>PHP NUBDIZC</b>	
Branch on oVerflow		CLear oVerflow		ROtate Left		Push Processor status	
Clear		Im 2		A 2 Z 5 Zx 6 Ab 6 Ax 7		Im 3	
Lb 2+		<b>GLD NUBDIZC</b>		<b>ROR NUBDIZC</b>		<b>PLP NUBDIZC</b>	
<b>BUS NUBDIZC</b>		CLear Decimal		ROtate Right		Pull Processor status	
Branch on oVerflow		Im 2		A 2 Z 5 Zx 6 Ab 6 Ax 7		Im 4	
Set		<b>SED NUBDIZC</b>		<b>RTI NUBDIZC</b>		<b>STX NUBDIZC</b>	
Lb 2+		SEt Decimal		ReTurn from Interrupt		STore X register	
<b>BCC NUBDIZC</b>		Im 2		Im 6		Z 3 Zy 4 Ab 4	
Branch on Carry		<b>JMP NUBDIZC</b>		<b>RTS NUBDIZC</b>		<b>STY NUBDIZC</b>	
Clear		JuMP		ReTurn from Subroutine		STore Y register	
Lb 2+		Ab 3 In 5		Im 6		Z 3 Zx 4 Ab 4	
<b>BCS NUBDIZC</b>							
Branch on Carry Set							
Lb 2+							
<b>BNE NUBDIZC</b>							
Branch on Not Equal							
Lb 2+							