LJ Technical Systems

6502 Instruction Set

Reference Manual

MP622/A

Issue Number: MP622/A
Written by: LJ Technical Publications Dept.

LJ Technical Systems Ltd.
Francis Way
Bowthorpe Industrial Estate
Norwich. NR5 9JA. England
Telephone: (01603) 748001
Fax: (01603) 746340

LJ Technical Systems Inc.
85, Corporate Drive, Holtsville,
11742-2007, New York, USA.
Telephone: (631) 758 1616
Fax: (631) 758 1788

6502 Instruction Set

This manual provides a summary of the 6502 Instruction Set. Throughout the summary, the following notations apply:

A	Accumulator
X,Y	Index Registers
M	Memory
P	Processor Status Register
S	Stack Pointer
✓	Change
-	No Change
+	Add
∧	Logical AND
—	Subtract
∀	Logical Exclusive Or
↑	Transfer from Stack
↓	Transfer to Stack
→	Transfer to
←	Transfer from
∨	Logical OR
PC	Program Counter
PCH	Program Counter High
PCL	Program Counter Low
OPER	Operand
#	Immediate Addressing Mode

ADD Memory to Accumulator with Carry

ADC **Add memory to accumulator with carry** **ADC**

Operation: A + M + C → A, C

N Z C I D V
✓ ✓ ✓ - - ✓

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Immediate	ADC #Oper	69	2	2
Zero Page	ADC Oper	65	2	3
Zero Page, X	ADC Oper, X	75	2	4
Absolute	ADC Oper	6D	3	4
Absolute, X	ADC Oper, X	7D	3	4*
Absolute, Y	ADC Oper, Y	79	3	4*
(Indirect, X)	ADC (Oper, X)	61	2	6
(Indirect), Y	ADC (Oper), Y	71	2	5*

* Add 1 if page boundary is crossed.

"AND" Memory With Accumulator

AND **"AND" memory with accumulator** **AND**

Logical AND to the accumulator

Operation: A /\ M → A

N Z C I D V
✓ ✓ - - - -

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Immediate	AND #Oper	29	2	2
Zero Page	AND Oper	25	2	3
Zero Page, X	AND Oper, X	35	2	4
Absolute	AND Oper	2D	3	4
Absolute, X	AND Oper, X	3D	3	4*
Absolute, Y	AND Oper, Y	39	3	4*
(Indirect, X)	AND (Oper, X)	21	2	6
(Indirect), Y	AND (Oper), Y	31	2	5*

* Add 1 if page boundary is crossed.

ASL Shift Left One Bit (Memory or Accumulator)

ASL **ASL Shift Left One Bit (Memory or Accumulator)** **ASL**

Operation: C ← 7 6 5 4 3 2 1 ∅ ←Ø

N Z C I D V
✓ ✓ ✓ - - -

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Accumulator	ASL A	0A	1	2
Zero Page	ASL Oper	06	2	5
Zero Page, X	ASL Oper, X	16	2	6
Absolute	ASL Oper	0E	3	6
Absolute, X	ASL Oper, X	1E	3	7

BCC Branch on Carry Clear

BCC **BCC Branch on Carry Clear** **BCC**

Operation: Branch on C = 0

N Z C I D V
- - - - - -

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Relative	BCC Oper	90	2	2*

* Add 1 if branch occurs to same page.
* Add 2 if branch occurs to different page.

BCS Branch on Carry Set

BCS **BCS Branch on Carry Set** **BCS**

Operation: Branch on C = 1

N Z C I D V
- - - - - -

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Relative	BCS Oper	B0	2	2*

* Add 1 if branch occurs to same page
* Add 2 if branch occurs to different page.

BEQ Branch on Result Zero

BEQ **BEQ Branch on Result Zero** **BEQ**

Operation: Branch on Z = 1

N	Z	C	I	D	V
-	-	-	-	-	-

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Relative	BEQ Oper	F0	2	2*

* Add 1 if branch occurs to same page
* Add 2 if branch occurs to different page

BIT Test Bits in Memory with Accumulator

BIT **BIT Test Bits in Memory with Accumulator** **BIT**

Operation: A /\ M, M7 → N, M6 → V

Bits 6 and 7 are transferred to the status register.
If the result of A /\ M is zero then Z = 1, otherwise Z = 0

N	Z	C	I	D	V
M7	✓	-	-	-	M6

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Zero Page Absolute	BIT Oper BIT Oper	24 2C	2 3	3 4

BMI Branch on Result Minus

BMI **BMI Branch on result minus** **BMI**

Operation: Branch on N = 1

N	Z	C	I	D	V
-	-	-	-	-	-

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Relative	BMI Oper	30	2	2*

* Add 1 if branch occurs to same page
* Add 2 if branch occurs to different page.

BNE Branch on Result Not Zero

BNE **BNE Branch on result not zero** **BNE**

Operation: Branch on Z = 0

N	Z	C	I	D	V
-	-	-	-	-	-

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Relative	BNE Oper	D0	2	2*

* Add 1 if branch occurs to same page
* Add 2 if branch occurs to different page.

BPL Branch on Result Plus

BPL **BPL Branch on Result Plus** **BPL**

Operation: Branch on N = 0

N	Z	C	I	D	V
-	-	-	-	-	-

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Relative	BPL Oper	10	2	2*

* Add 1 if branch occurs to same page
* Add 2 if branch occurs to different page

BRK Force Break

BRK **BRK Force Break** **BRK**

Operation: Forced Interrupt PC + 2 ↓ P ↓

N	Z	C	I	D	V
-	-	-	1	-	-

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Implied	BRK	00	1	7

* A BRK command cannot be masked by setting I.

BVC Branch on Overflow Clear

BVC **BVC Branch on Overflow Clear** **BVC**

Operation: Branch on V = 0

N	Z	C	I	D	V
-	-	-	-	-	-

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Relative	BVC Oper	50	2	2*

* Add 1 if branch occurs to same page
* Add 2 if branch occurs to different page.

BVS Branch on Overflow Set

BVS **Branch on Overflow Set** **BVS**

Operation: Branch on V = 1

N	Z	C	I	D	V
-	-	-	-	-	-

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Relative	BVS Oper	70	2	2*

* Add 1 if branch occurs to same page
* Add 2 if branch occurs to different page.

CLC Clear Carry Flag

CLC **CLC Clear Carry Flag** **CLC**

Operation: $\varnothing \rightarrow C$

N	Z	C	I	D	V
-	-	$\varnothing$	-	-	-

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Implied	CLC	18	1	2

CLD Clear Decimal Mode

CLD **CLD Clear Decimal Mode** **CLD**

Operation: ∅ → D

N	Z	C	I	D	V
-	-	-	-	∅	-

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Implied	CLD	D8	1	2

CLI Clear Interrupt Disable Bit

CLI **CLI Clear Interrupt Disable Bit** **CLI**

Operation: ∅ → 1

N	Z	C	I	D	V
-	-	-	∅	-	-

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Implied	CLI	58	1	2

CLV Clear Overflow Flag

CLV **CLV Clear Overflow Flag** **CLV**

Operation: ∅ → V

N	Z	C	I	D	V
-	-	-	-	-	∅

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Implied	CLV	B8	1	2

CMP Compare Memory and Accumulator

CMP **CMP Compare Memory and Accumulator** **CMP**

Operation: A - M

N Z C I D V
✓✓ ✓ - - -

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Immediate	CMP #Oper	C9	2	2
Zero Page	CMP Oper	C5	2	3
Zero Page, X	CMP Oper, X	D5	2	4
Absolute	CMP Oper	CD	3	4
Absolute, X	CMP Oper, X	DD	3	4*
Absolute, Y	CMP Oper, Y	D9	3	4*
(Indirect, X)	CMP (Oper, X)	C1	2	6
(Indirect), Y	CMP (Oper), Y	D1	2	5*

* Add 1 if page boundary is crossed.

CPX Compare Memory and Index X

CPX **Compare Memory and Index X** **CPX**

Operation: X - M

N Z C I D V
✓✓ ✓ - - -

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Immediate	CPX #Oper	E0	2	2
Zero Page	CPX Oper	E4	2	3
Absolute	CPX Oper	EC	3	4

CPY Compare Memory and Index Y

CPY **CPY Compare Memory and Index Y** **CPY**

Operation: Y - M

N Z C I D V
✓✓ ✓ - - -

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Immediate	CPY #Oper	C0	2	2
Zero Page	CPY Oper	C4	2	3
Absolute	CPY Oper	CC	3	4

DEC Decrement Memory by One

DEC **DEC Decrement Memory by One** **DEC**

Operation: M - 1 → M

N	Z	C	I	D	V
✓	✓	-	-	-	-

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Zero Page	DEC Oper	C6	2	5
Zero Page, X	DEC Oper, X	D6	2	6
Absolute	DEC Oper	CE	3	6
Absolute, X	DEC Oper, X	DE	3	7

DEX Decrement Index X by One

DEX **DEX Decrement Index X by One** **DEX**

Operation: X - 1 → X

N	Z	C	I	D	V
✓	✓	-	-	-	-

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Implied	DEX	CA	1	2

DEY Decrement Index Y by One

DEY **DEY Decrement Index Y by One** **DEY**

Operation: Y - 1 → Y

N	Z	C	I	D	V
✓	✓	-	-	-	-

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Implied	DEY	88	1	2

EOR "Exclusive-Or" Memory with Accumulator

EOR **" Exclusive - Or" Memory with Accumulator** **EOR**

Operation: $A \forall M \rightarrow A$

N Z C I D V
✓ ✓ - - - -

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Immediate	EOR #Oper	49	2	2
Zero Page	EOR Oper	45	2	3
Zero Page, X	EOR Oper, X	55	2	4
Absolute	EOR Oper	4D	3	4
Absolute, X	EOR Oper, X	5D	3	4*
Absolute, Y	EOR Oper, Y	59	3	4*
(Indirect, X)	EOR (Oper, X)	41	2	6
(Indirect), Y	EOR (Oper), Y	51	2	5*

* Add 1 if page boundary is crossed.

INC Increment Memory by One

INC **INC Increment Memory by One** **INC**

Operation: $M + 1 \rightarrow M$

N Z C I D V
✓ ✓ - - - -

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Zero Page	INC Oper	E6	2	5
Zero Page, X	INC Oper, X	F6	2	6
Absolute	INC Oper	EE	3	6
Absolute, X	INC Oper, X	FE	3	7

INX Increment Index X by One

INX **INX Increment Index X by One** **INX**

Operation: $X + 1 \rightarrow X$

N Z C I D V
✓ ✓ - - - -

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Implied	INX	E8	1	2

INY Increment Index Y by One

INY **INY Increment Index Y by One** **INY**

Operation: Y + 1 → Y

N	Z	C	I	D	V
✓	✓	-	-	-	-

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Implied	INY	C8	1	2

JMP Jump to New Location

JMP **JMP Jump to New Location** **JMP**

Operation: (PC + 1) → PCL
(PC + 2) → PCH

N	Z	C	I	D	V
-	-	-	-	-	-

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Absolute Indirect	JMP Oper JMP (Oper)	4C 6C	3 3	3 5

JSR Jump to New Location Saving Return Address

JSR **Jump to New Location Saving Return Address** **JSR**

Operation: PC + 2 ↓ , (PC + 1) → PCL
(PC + 2) → PCH

N	Z	C	I	D	V
-	-	-	-	-	-

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Absolute	JSR Oper	20	3	6

LDA Load Accumulator with Memory

LDA **Load Accumulator with Memory** **LDA**

Operation: M → A

N Z C I D V
✓ ✓ - - - -

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Immediate	LDA #Oper	A9	2	2
Zero Page	LDA Oper	A5	2	3
Zero Page, X	LDA Oper, X	B5	2	4
Absolute	LDA Oper	AD	3	4
Absolute, X	LDA Oper, X	BD	3	4*
Absolute, Y	LDA Oper, Y	B9	3	4*
(Indirect, X)	LDA (Oper, X)	A1	2	6
(Indirect), Y	LDA (Oper), Y	B1	2	5*

* Add 1 if page boundary is crossed.

LDX Load Index X With Memory

LDX **LDX Load Index X with Memory** **LDX**

Operation: M → X

N Z C I D V
✓ ✓ - - - -

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Immediate	LDX #Oper	A2	2	2
Zero Page	LDX Oper	A6	2	3
Zero Page, Y	LDX Oper, Y	B6	2	4
Absolute	LDX Oper	AE	3	4
Absolute, Y	LDX Oper, Y	BE	3	4*

* Add 1 if page boundary is crossed.

LDY Load Index Y With Memory

LDY **LDY Load Index Y with Memory** **LDY**

Operation: M → Y

N Z C I D V
✓ ✓ - - - -

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Immediate	LDY #Oper	A0	2	2
Zero Page	LDY Oper	A4	2	3
Zero Page, X	LDY Oper, X	B4	2	4
Absolute	LDY Oper	AC	3	4
Absolute, X	LDY Oper, X	BC	3	4*

* Add 1 if page boundary is crossed.

LSR Shift Right One Bit (Memory or Accumulator)

LSR **LSR Shift Right One Bit (memory or accumulator)** **LSR**

Operation: ∅→ 7 6 5 4 3 2 1 ∅ → C

N Z C I D V
∅ ✓ ✓ - - -

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Accumulator	LSR A	4A	1	2
Zero Page	LSR Oper	46	2	5
Zero Page, X	LSR Oper, X	56	2	6
Absolute	LSR Oper	4E	3	6
Absolute, X	LSR Oper, X	5E	3	7

NOP No Operation

NOP **NOP No Operation** **NOP**

Operation: No Operation (2 cycles)

N Z C I D V
- - - - - -

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Implied	NOP	EA	1	2

ORA "OR" Memory with Accumulator

ORA **ORA "OR" Memory with Accumulator** **ORA**

Operation: A ∨ M → A

N Z C I D V
✓ ✓ - - - -

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Immediate	ORA #Oper	09	2	2
Zero Page	ORA Oper	05	2	3
Zero Page, X	ORA Oper, X	15	2	4
Absolute	ORA Oper	0D	3	4
Absolute, X	ORA Oper, X	1D	3	4*
Absolute, Y	ORA Oper, Y	19	3	4*
(Indirect, X)	ORA (Oper, X)	01	2	6
(Indirect), Y	ORA (Oper), Y	11	2	5*

* Add 1 if page boundary is crossed.

PHA Push Accumulator on Stack

PHA **PHA Push Accumulator on Stack** **PHA**

Operation: A ↓

N Z C I D V
- - - - - -

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Implied	PHA	48	1	3

PHP Push Processor Status on Stack

PHP **PHP Push Processor Status on Stack** **PHP**

Operation: P ↓

N Z C I D V
- - - - - -

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Implied	PHP	08	1	3

PLA Pull Accumulator from Stack

PLA **PLA Pull Accumulator from Stack** **PLA**

Operation: A ↑

N Z C I D V
✓ ✓ - - - -

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Implied	PLA	68	1	4

PLP Pull Processor Status from Stack

PLP **PLP Pull Processor Status from Stack** **PLP**

Operation: P ↑

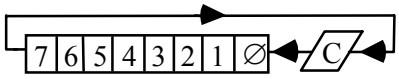
N Z C I D V
From Stack

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Implied	PLP	28	1	4

ROL Rotate One Bit Left (Memory or Accumulator)

ROL **ROL Rotate One Bit Left (memory or accumulator)** **ROL**

Operation:



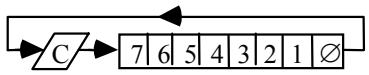
M or A

N Z C I D V
✓ ✓ ✓ - - -

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Accumulator	ROL A	2A	1	2
Zero Page	ROL Oper	26	2	5
Zero Page, X	ROL Oper, X	36	2	6
Absolute	ROL Oper	2E	3	6
Absolute, X	ROL Oper, X	3E	3	7

ROR Rotate One Bit Right (Memory or Accumulator)

ROR **ROR Rotate One Bit Right (memory or accumulator)** **ROR**

Operation: 

M or A

N Z C I D V
✓ ✓ ✓ - - -

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Accumulator	ROR A	6A	1	2
Zero Page	ROR Oper	66	2	5
Zero Page, X	ROR Oper, X	76	2	6
Absolute	ROR Oper	6E	3	6
Absolute, X	ROR Oper, X	7E	3	7

RTI Return from Interrupt

RTI **RTI Return from Interrupt** **RTI**

Operation: P ↑ PC ↑

N Z C I D V
From Stack

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Implied	RTI	40	1	6

RTS Return from Sub-Routine

RTS **RTS Return from Sub-Routine** **RTS**

Operation: PC ↑ , PC + 1 → PC

N Z C I D V
- - - - - -

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Implied	RTS	60	1	6

SBC Subtract Memory from Accumulator with Borrow

SBC **SBC Subtract Memory from Accumulator with Borrow** **SBC**

Operation: A - M - C → A
Note: C = Borrow

N	Z	C	I	D	V
✓	✓	✓	-	-	-

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Immediate	SBC #Oper	E9	2	2
Zero Page	SBC Oper	E5	2	3
Zero Page, X	SBC Oper, X	F5	2	4
Absolute	SBC Oper	ED	3	4
Absolute, X	SBC Oper, X	FD	3	4*
Absolute, Y	SBC Oper, Y	F9	3	4*
(Indirect, X)	SBC (Oper, X)	E1	2	6
(Indirect), Y	SBC (Oper), Y	F1	2	5*

* Add 1 if page boundary is crossed.

SEC Set Carry Flag

SEC **SEC Set Carry Flag** **SEC**

Operation: 1 → C

N	Z	C	I	D	V
-	-	1	-	-	-

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Implied	SEC	38	1	2

SED Set Decimal Mode

SED **SED Set Decimal Mode** **SED**

Operation: 1 → D

N	Z	C	I	D	V
-	-	-	-	1	-

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Implied	SED	F8	1	2

SEI Set Interrupt Disable Status

SEI **SEI Set Interrupt Disable Status** **SEI**

Operation: $1 \rightarrow I$

N	Z	C	I	D	V
-	-	-	1	-	-

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Implied	SEI	78	1	2

STA Store Accumulator in Memory

STA **STA Store Accumulator in Memory** **STA**

Operation: $A \rightarrow M$

N	Z	C	I	D	V
-	-	-	-	-	-

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Zero Page	STA Oper	85	2	3
Zero Page, X	STA Oper, X	95	2	4
Absolute	STA Oper	8D	3	4
Absolute, X	STA Oper, X	9D	3	5
Absolute, Y	STA Oper, Y	99	3	5
(Indirect, X)	STA (Oper, X)	81	2	6
(Indirect), Y	STA (Oper), Y	91	2	6

STX Store Index X in Memory

STX **STX Store Index X in Memory** **STX**

Operation: $X \rightarrow M$

N	Z	C	I	D	V
-	-	-	-	-	-

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Zero Page	STX Oper	86	2	3
Zero Page, Y	STX Oper, Y	96	2	4
Absolute	STX Oper	8E	3	4

STY Store Index Y in Memory

STY **STY Store Index Y in Memory** **STY**

Operation: $Y \rightarrow M$

N	Z	C	I	D	V
-	-	-	-	-	-

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Zero Page	STY Oper	84	2	3
Zero Page, X	STY Oper, X	94	2	4
Absolute	STY Oper	8C	3	4

TAX Transfer Accumulator to Index X

TAX **TAX Transfer Accumulator to Index X** **TAX**

Operation: $A \rightarrow X$

N	Z	C	I	D	V
✓	✓	-	-	-	-

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Implied	TAX	AA	1	2

TAY Transfer Accumulator to Index Y

TAY **TAY Transfer Accumulator to Index Y** **TAY**

Operation: $A \rightarrow Y$

N	Z	C	I	D	V
✓	✓	-	-	-	-

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Implied	TAY	A8	1	2

TSX Transfer Stack Pointer to Index X

TSX **TSX Transfer Stack Pointer to Index X** **TSX**

Operation: S → X

N	Z	C	I	D	V
✓	✓	-	-	-	-

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Implied	TSX	BA	1	2

TXA Transfer Index X to Accumulator

TXA **TXA Transfer Index X to Accumulator** **TXA**

Operation: X → A

N	Z	C	I	D	V
✓	✓	-	-	-	-

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Implied	TXA	8A	1	2

TXS Transfer Index X to Stack Pointer

TXS **TXS Transfer Index X to Stack Pointer** **TXS**

Operation: X → S

N	Z	C	I	D	V
-	-	-	-	-	-

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Implied	TXS	9A	1	2

TYA Transfer Index Y to Accumulator

TYA **TYA Transfer Index Y to Accumulator** **TYA**

Operation: $Y \rightarrow A$

N	Z	C	I	D	V
✓	✓	-	-	-	-

Addressing Mode	Assembly Language Form	Op Code	No. of Bytes	No. of Cycles
Implied	TYA	98	1	2

Summary of Addressing Modes

	Accumulator	Immediate	Zero Page	Zero Page, X	Zero Page, Y	Absolute	Absolute, X	Absolute, Y	Implied	Relative	(Indirect, X)	(Indirect), Y	Absolute Indirect
ADC		69	65	75		6D	7D	79			61	71	
AND		29	25	35		2D	3D	39			21	31	
ASL	0A		06	16		0E	1E						
BCC										90			
BCS										B0			
BEQ										F0			
BIT			24			2C							
BMI										30			
BNE										D0			
BPL										10			
BRK									00				
BVC										50			
BVS										70			
CLC									18				
CLD									D8				
CLI									58				
CLV									B8				
CMP		C9	C5	D5		CD	DD	D9			C1	D1	
CPX		E0	E4			EC							
CPY		C0	C4			CC							
DEC			C6	D6		CE	DE						
DEX									CA				
DEY									88				
EOR		49	45	55		4D	5D	59			41	51	
INC			E6	F6		EE	FE						
INX									E8				
INY									C8				
JMP						4C							6C

	Accumulator	Immediate	Zero Page	Zero Page, X	Zero Page, Y	Absolute	Absolute, X	Absolute, Y	Implied	Relative	(Indirect, X)	(Indirect), Y	Absolute Indirect
JSR						20							
LDA		A9	A5	B5		AD	BD	B9			A1	B1	
LDX		A2	A6		B6	AE		BE					
LDY		A0	A4	B4		AC	BC						
LSR	4A		46	56		4E	5E						
NOP									EA				
ORA		09	05	15		0D	1D	19			01	11	
PHA									48				
PHP									08				
PLA									68				
PLP									28				
ROL	2A		26	36		2E	3E						
ROR	6A		66	76		6E	7E						
RTI									40				
RTS									60				
SBC		E9	E5	F5		ED	FD	F9			E1	F1	
SEC									38				
SED									F8				
SEI									78				
STA			85	95		8D	9D	99			81	91	
STX			86		96	8E							
STY			84	94		8C							
TAX									AA				
TAY									A8				
TYA									98				
TSX									BA				
TXA									8A				
TXS									9A				

Numerical Listing

00	BRK	30	BMI
01	ORA - (Indirect, X)	31	AND - (Indirect,X)
02	Future Expansion	32	Future Expan sion
03	Future Expansion	33	Future Expansion
04	Future Expansion	34	Future Expansion
05	ORA - Zero Page	35	AND - Zero Page,X
06	ASL - Zero Page	36	ROL - Zero Page,X
07	Future Expansion	37	Future Expansion
08	PHP	38	SEC
09	ORA - Immediate	39	AND - Absolute,Y
0A	ASL - Accumulator	3A	Future Expansion
0B	Future Expansion	3B	Future Expansion
0C	Future Expansion	3C	Future Expansion
0D	ORA - Absolute	3D	AND - Absolute,X
0E	ASL - Absolute	3E	ROL - Absolute,X
0F	Future Expansion	3F	Future Expansion
10	BPL	40	RTI
11	ORA - (Indirect),Y	41	EOR - (Indirect,X)
12	Future Expansion	42	Future Expansion
13	Future Expansion	43	Future Expansion
14	Future Expansion	44	Future Expansion
15	ORA - Zero Page,X	45	EOR - Zero Page
16	ASL - Zero Page,X	46	LSR - Zero Page
17	Future Expansion	47	Future Expansion
18	CLC	48	PHA
19	ORA - Absolute,Y	49	EOR - Immediate
1A	Future Expansion	4A	LSR - Accumulator
1B	Future Expansion	4B	Future Expansion
1C	Future Expansion	4C	JMP - Absolute
1D	ORA - Absolute,X	4D	EOR - Absolute
1E	ASL - Absolute,X	4E	LSR - Absolute
1F	Future Expansion	4F	Future Expansion
20	JSR	50	BVC
21	AND - (Indirect,X)	51	EOR - (Indirect),Y
22	Future Expansion	52	Future Expansion
23	Future Expansion	53	Future Expansion
24	BIT - Zero Page	54	Future Expansion
25	AND - Zero Page	55	EOR - Zero Page,X
26	ROL - Zero Page	56	LSR - Zero Page,X
27	Future Expansion	57	Future Expansion
28	PLP	58	CLI
29	AND - Immediate	59	EOR - Absolute,Y
2A	ROL - Accumulator	5A	Future Expansion
2B	Future Expansion	5B	Future Expansion
2C	BIT - Absolute	5C	Future Expansion
2D	AND - Absolute	5D	EOR - Absolute,X
2E	ROL - Absolute	5E	LSR - Absolute,X
2F	Future Expansion	5F	Future Expansion

60	RTS	90	BCC
61	ADC - (Indirect,X)	91	STA - (Indirect),Y
62	Future Expansion	92	Future Expansion
63	Future Expansion	93	Future Expansion
64	Future Expansion	94	STY - Zero Page,X
65	ADC - Zero Page	95	STA - Zero Page,X
66	ROR - Zero Page	96	STX - Zero Page,Y
67	Future Expansion	97	Future Expansion
68	PLA	98	TYA
69	ADC - Immediate	99	STA - Absolute,Y
6A	ROR - Accumulator	9A	TXS
6B	Future Expansion	9B	Future Expansion
6C	JMP - Indirect	9C	Future Expansion
6D	ADC - Absolute	9D	STA - Absolute,X
6E	ROR - Absolute	9E	Future Expansion
6F	Future Expansion	9F	Future Expansion
70	BVS	A0	LDY - Immediate
71	ADC - (Indirect),Y	A1	LDA - (Indirect,X)
72	Future Expansion	A2	LDX - Immediate
73	Future Expansion	A3	Future Expansion
74	Future Expansion	A4	LDY - Zero Page
75	ADC - Zero Page,X	A5	LDA - Zero Page
76	ROR - Zero Page,X	A6	LDX - Zero Page
77	Future Expansion	A7	Future Expansion
78	SEI	A8	TAY
79	ADC - Absolute,Y	A9	LDA - Immediate
7A	Future Expansion	AA	TAX
7B	Future Expansion	AB	Future Expansion
7C	Future Expansion	AC	LDY - Absolute
7D	ADC - Absolute,X	AD	LDA - Absolute
7E	ROR - Absolute,X	AE	LDX - Absolute
7F	Future Expansion	AF	Future Expansion
80	Future Expansion	B0	BCS
81	STA - (Indirect,X)	B1	LDA - (Indirect),Y
82	Future Expansion	B2	Future Expansion
83	Future Expansion	B3	Future Expansion
84	STY - Zero Page	B4	LDY - Zero Page,X
85	STA - Zero Page	B5	LDA - Zero Page,X
86	STX - Zero Page	B6	LDX - Zero Page,Y
87	Future Expansion	B7	Future Expansion
88	DEY	B8	CLV
89	Future Expansion	B9	LDA - Absolute,Y
8A	TXA	BA	TSX
8B	Future Expansion	BB	Future Expansion
8C	STY - Absolute	BC	LDY - Absolute,X
8D	STA - Absolute	BD	LDA - Absolute,X
8E	STX - Absolute	BE	LDX - Absolute,Y
8F	Future Expansion	BF	Future Expansion

C0	CPY - Immediate
C1	CMP - (Indirect,X)
C2	Future Expansion
C3	Future Expansion
C4	CPY - Zero Page
C5	CMP - Zero Page
C6	DEC - Zero Page
C7	Future Expansion
C8	INY
C9	CMP - Immediate
CA	DEX
CB	Future Expansion
CC	CPY - Absolute
CD	CMP - Absolute
CE	DEC - Absolute
CF	Future Expansion
D0	BNE
D1	CMP - (Indirect),Y
D2	Future Expansion
D3	Future Expansion
D4	Future Expansion
D5	CMP - Zero Page,X
D6	DEC - Zero Page,X
D7	Future Expansion
D8	CLD
D9	CMP - Absolute,Y
DA	Future Expansion
DB	Future Expansion
DC	Future Expansion
DD	CMP - Absolute,X
DE	DEC - Absolute,X
DF	Future Expansion
E0	CPX - Immediate
E1	SBC - (Indirect,X)
E2	Future Expansion
E3	Future Expansion
E4	CPX - Zero Page
E5	SBC - Zero Page
E6	INC - Zero Page
E7	Future Expansion
E8	INX
E9	SBC - Immediate
EA	NOP
EB	Future Expansion
EC	CPX - Absolute
ED	SBC - Absolute
EE	INC - Absolute
EF	Future Expansion
F0	BEQ
F1	SBC - (Indirect),Y
F2	Future Expansion
F3	Future Expansion
F4	Future Expansion
F5	SBC - Zero Page,X
F6	INC - Zero Page,X
F7	Future Expansion
F8	SED
F9	SBC - Absolute,Y
FA	Future Expansion
FB	Future Expansion
FC	Future Expansion
FD	SBC - Absolute,X
FE	INC - Absolute,Y
FF	Future Expansion

Alphabetical List

ADC	Add Memory to Accumulator with Carry
AND	'AND' Memory with Accumulator
ASL	Shift Left One Bit (Memory to Accumulator)
BCC	Branch on Carry Clear
BCS	Branch on Carry Set
BEQ	Branch on Result Zero
BIT	Test Bits in Memory with Accumulator
BMI	Branch on Result Minus
BNE	Branch on Result not Zero
BPL	Branch on Result Plus
BRK	Force Break
BVC	Branch on Overflow Clear
BVS	Branch on Overflow Set
CLC	Clear Carry Flag
CLD	Clear Decimal Mode
CLI	Clear Interrupt Disable Bit
CLV	Clear Overflow Flag
CMP	Compare Memory and Accumulator
CPX	Compare Memory and Index X
CPY	Compare Memory and Index Y
DEC	Decrement Memory by One
DEX	Decrement Index X by One
DEY	Decrement Index Y by One
EOR	'Exclusive Or' Memory with Accumulator
INC	Increment Memory by One
INX	Increment Index X by One
INY	Increment Index Y by One
JMP	Jump to New Location
JSR	Jump to New Location Saving Return Address
LDA	Load Accumulator with Memory
LDX	Load Index X with Memory
LDY	Load Index Y with Memory
LSR	Shift Right One Bit (Memory or Accumulator)
NOP	No Operation
ORA	'OR' Memory with Accumulator
PHA	Push Accumulator to Stack
PHP	Push Processor Status to Stack
PLA	Pull Accumulator from Stack
PLP	Pull Processor Status from Stack
ROL	Rotate One Bit Left (Memory or Accumulator)
ROR	Rotate One Bit Right (Memory or Accumulator)
RTI	Return from Interrupt

RTS	Return from Subroutine
SBC	Subtract Memory from Accumulator with Borrow
SEC	Set Carry Flag
SED	Set Decimal Mode
SEI	Set Interrupt Disable Status
STA	Store Accumulator in Memory
STX	Store Index X in Memory
STY	Store Index Y in Memory
TAX	Transfer Accumulator to Index X
TAY	Transfer Accumulator to Index Y
TSX	Transfer Stack Pointer to Index X
TXA	Transfer Index X to accumulator
TXS	Transfer Index X to Stack Pointer
TYA	Transfer Index Y to Accumulator

Instruction Addressing Modes and Related Execution Times (in clock cycles)													
	Accumulator	Immediate	Zero Page	Zero Page, X	Zero Page, Y	Absolute	Absolute, X	Absolute, Y	Implied	Relative	(Indirect, X)	(Indirect), Y	Absolute Indirect
ADC		2	3	4		4	4*	4*			6	5*	
AND		2	3	4		4	4*	4*		6	6	5*	
ASL	2		5	6		6	7						
BCC										2**			
BCS										2**			
BEQ										2**			
BIT			3			4							
BMI										2**			
BNE										2**			
BPL										2**			
BRK													
BVC										2**			
BVS										2**			
CLC									2				
CLD									2				
CLI									2				
CLV									2				
CMP		2	3	4		4	4*	4*			6	5*	
CPX		2	3			4							
CPY		2	3			4							
DEC			5	6		6	7						
DEX									2				
DEY									2				
EOR		2	3	4		4	4*	4*			6	5*	
INC			5	6		6	7						
INX									2				
INY									2				
JMP						3							5

Instruction Addressing Modes and Related Execution Times (in clock cycles)													
	Accumulator	Immediate	Zero Page	Zero Page, X	Zero Page, Y	Absolute	Absolute, X	Absolute, Y	Implied	Relative	(Indirect, X)	(Indirect), Y	Absolute Indirect
JSR						6							
LDA		2	3	4		4	4*	4*			6	5*	
LDX		2	3		4	4		4*					
LDY		2	3	4		4	4*						
LSR	2		5	6		6	7						
NOP									2				
ORA		2	3	4		4	4*	4*			6	5*	
PHA									3				
PHP									3				
PLA									4				
PLP									4				
ROL	2		5	6		6	7						
ROR	2		5	6		6	7						
RTI									6				
RTS									6				
SBC		2	3	4		4	4*	4*			6	5*	
SEC									2				
SED									2				
SEI									2				
STA			3	4		4	5	6			6	6	
STX			3		4	4							
STY			3	4		4							
TAX									2				
TAY									2				
TSX									2				
TXA									2				
TXS									2				
TYA									2				

* Add one cycle if indexing across page boundary.
** Add one cycle if branch is taken. Add one additional if branching operation crosses page boundary.