MOS 6502 / MOS 6510

Deep Dive Instruction Set

LDA Load Accumulator with Memory

M -> A

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

addressing	assembler	opc	bytes	cycls
immediate	LDA #oper	A9	2	2
zeropage	LDA oper	A5	2	3
zeropage,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 to cycles if page boundary is crossed

** add 1 to cycles if branch occurs on same page
   add 2 to cycles if branch occurs to different page

Legend to Flags: + ... modified
                  - .... not modified
                  1 .... set
                  0 .... cleared
                 M6 .... memory bit 6
                 M7 .... memory bit 7

ADC Add Memory to Accumulator with Carry

A + M + C -> A, C

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

addressing	assembler	opc	bytes	cycls
immediate	ADC #oper	69	2	2
zeropage	ADC oper	65	2	3
zeropage,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 to cycles if page boundary is crossed

** add 1 to cycles if branch occurs on same page
   add 2 to cycles if branch occurs to different page

Legend to Flags: + ... modified
                  - .... not modified
                  1 .... set
                  0 .... cleared
                 M6 .... memory bit 6
                 M7 .... memory bit 7

AND AND Memory with Accumulator

A AND M -> A

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

addressing	assembler	opc	bytes	cycls
immediate	AND #oper	29	2	2
zeropage	AND oper	25	2	3
zeropage,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 to cycles if page boundary is crossed

** add 1 to cycles if branch occurs on same page
   add 2 to cycles if branch occurs to different page

Legend to Flags: + ... modified
- .... not modified
1 .... set
0 .... cleared
M6 .... memory bit 6
M7 .... memory bit 7

A	.	B	=	C
1		1	=	1
1		0	=	0
0		1	=	0
0		0	=	0

```
 10010011
 00100101
 --------
 00000001
```

ASL Shift Left One Bit (Memory or Accumulator)

C <- ←76543210→ <- 0

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

10010010

addressing	assembler	opc	bytes	cycls
accumulator	ASL A	0A	1	2
zeropage	ASL oper	06	2	5
zeropage,X	ASL oper,X	16	2	6
absolute	ASL oper	0E	3	6
absolute,X	ASL oper,X	1E	3	7

* add 1 to cycles if page boundary is crossed

** add 1 to cycles if branch occurs on same page
   add 2 to cycles if branch occurs to different page

Legend to Flags:
+ .... modified
- .... not modified
1 .... set
0 .... cleared
M6 .... memory bit 6
M7 .... memory bit 7

BCC Branch on Carry Clear

branch on C = 0

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

addressing	assembler	opc	bytes	cycls
relative	BCC oper	90	2	2**

BCS Branch on Carry Set

branch on C = 1

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

addressing	assembler	opc	bytes	cycls
relative	BCS oper	B0	2	2**

* add 1 to cycles if page boundary is crossed

** add 1 to cycles if branch occurs on same page
   add 2 to cycles if branch occurs to different page

Legend to Flags:
+ .... modified
- .... not modified
1 .... set
0 .... cleared
M6 .... memory bit 6
M7 .... memory bit 7

BEQ Branch on Result Zero

branch on Z = 1

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

addressing	assembler	opc	bytes	cycls
relative	BEQ oper	F0	2	2**

BMI Branch on Result Minus

branch on N = 1

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

addressing	assembler	opc	bytes	cycls
relative	BMI oper	30	2	2**

* add 1 to cycles if page boundary is crossed

** add 1 to cycles if branch occurs on same page
   add 2 to cycles if branch occurs to different page

Legend to Flags:
+ .... modified
- .... not modified
1 .... set
0 .... cleared
M6 .... memory bit 6
M7 .... memory bit 7

BIT Test Bits in Memory with Accumulator

bits 7 and 6 of operand are transferred to bit 7 and 6 of SR (N,V);
the zeroflag is set to the result of operand AND accumulator.

A AND M, M7 -> N, M6 -> V

N	Z	C	I	D	V
M7	+	-	-	-	M6

addressing	assembler	opc	bytes	cycls
zeropage	BIT oper	24	2	3
absolute	BIT oper	2C	3	4

11010010

* add 1 to cycles if page boundary is crossed

** add 1 to cycles if branch occurs on same page
   add 2 to cycles if branch occurs to different page

Legend to Flags:
+ .... modified
- .... not modified
1 .... set
0 .... cleared
M6 .... memory bit 6
M7 .... memory bit 7

BNE Branch on Result not Zero

branch on Z = 0

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

addressing	assembler	opc	bytes	cycls
relative	BNE oper	D0	2	2**

BPL Branch on Result Plus

branch on N = 0

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

addressing	assembler	opc	bytes	cycls
relative	BPL oper	10	2	2**

* add 1 to cycles if page boundary is crossed

** add 1 to cycles if branch occurs on same page
   add 2 to cycles if branch occurs to different page

Legend to Flags:
+ .... modified
- .... not modified
1 .... set
0 .... cleared
M6 .... memory bit 6
M7 .... memory bit 7

BVC Branch on Overflow Clear

branch on V = 0

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

addressing	assembler	opc	bytes	cycls
relative	BVC oper	50	2	2**

BVS Branch on Overflow Set

branch on V = 1

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

addressing	assembler	opc	bytes	cycls
relative	BVS oper	70	2	2**

* add 1 to cycles if page boundary is crossed

** add 1 to cycles if branch occurs on same page
   add 2 to cycles if branch occurs to different page

Legend to Flags:
+ .... modified
- .... not modified
1 .... set
0 .... cleared
M6 .... memory bit 6
M7 .... memory bit 7

BRK Force Break

interrupt,
push PC+2, push SR

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

addressing	assembler	opc	bytes	cycls
implied	BRK	00	1	7

CLC Clear Carry Flag

0 -> C

N	Z	C	I	D	V
-	-	0	-	-	-

addressing	assembler	opc	bytes	cycls
implied	CLC	18	1	2

* add 1 to cycles if page boundary is crossed

** add 1 to cycles if branch occurs on same page
   add 2 to cycles if branch occurs to different page

Legend to Flags: + ... modified
- .... not modified
1 .... set
0 .... cleared
M6 .... memory bit 6
M7 .... memory bit 7

CLD Clear Decimal Mode

0 -> D

N	Z	C	I	D	V
-	-	-	-	0	-

addressing	assembler	opc	bytes	cycls
implied	CLD	D8	1	2

CLI Clear Interrupt Disable Bit

0 -> I

N	Z	C	I	D	V
-	-	-	0	-	-

addressing	assembler	opc	bytes	cycls
implied	CLI	58	1	2

* add 1 to cycles if page boundary is crossed

** add 1 to cycles if branch occurs on same page
   add 2 to cycles if branch occurs to different page

Legend to Flags: + ... modified
- .... not modified
1 .... set
0 .... cleared
M6 .... memory bit 6
M7 .... memory bit 7

CLV Clear Overflow Flag

0 -> V

N	Z	C	I	D	V
-	-	-	-	-	0

addressing	assembler	opc	bytes	cycls
implied	CLV	B8	1	2

* add 1 to cycles if page boundary is crossed

** add 1 to cycles if branch occurs on same page
   add 2 to cycles if branch occurs to different page

Legend to Flags:
+ .... modified
- .... not modified
1 .... set
0 .... cleared
M6 .... memory bit 6
M7 .... memory bit 7