commodore
semiconductor group

NMOS

MPS
6510
MICROPROCESSOR
WITH I/O

6510 MICROPROCESSOR WITH I/O

DESCRIPTION

The 6510 is a low-cost microcomputer system capable of solving a broad range of small-systems and peripheral-control problems at minimum cost to the user.

An 8-bit Bi-Directional I/O Port is located on-chip with the Output Register at Address 0 0 0 0 and the Data-Direction Register at Address 0 0 0 1. The I/O Port is bit-by-bit programmable.

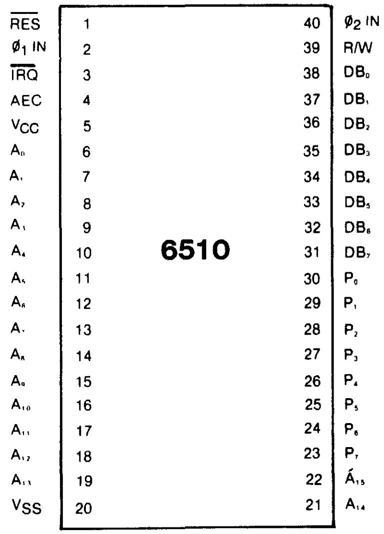
The Three-State sixteen-bit Address Bus allows Direct Memory Accessing (DMA) and multi-processor systems sharing a common memory.

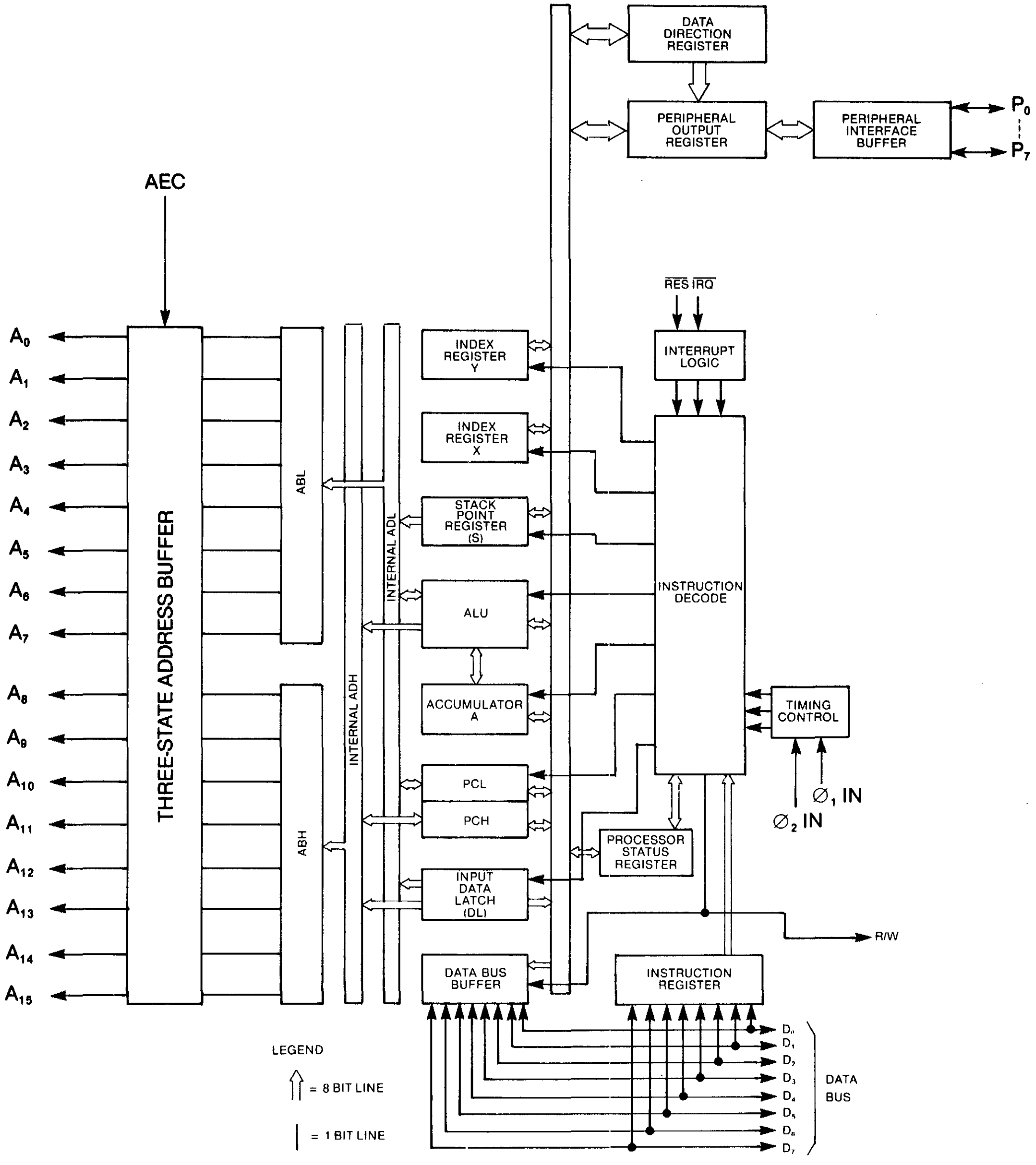
The internal processor architecture is identical to the MOS Technology 6502 to provide software compatibility.

FEATURES OF THE 6510 . . .

- 8-Bit Bi-Directional I/O Port
- Single +5 volt supply
- N channel, silicon gate, depletion load technology
- Eight bit parallel processing
- 56 Instructions
- Decimal and binary arithmetic
- Thirteen addressing modes
- True indexing capability
- Programmable stack pointer
- Variable length stack
- Interrupt capability
- 8 Bit Bi-Directional Data Bus
- Addressable memory range of up to 65K bytes
- Direct memory access capability
- Bus compatible with M6800
- Pipeline architecture
- 1 MHz and 2MHz operation
- Use with any type or speed memory

PIN CONFIGURATION





6510 BLOCK DIAGRAM

6510 CHARACTERISTICS

MAXIMUM RATINGS

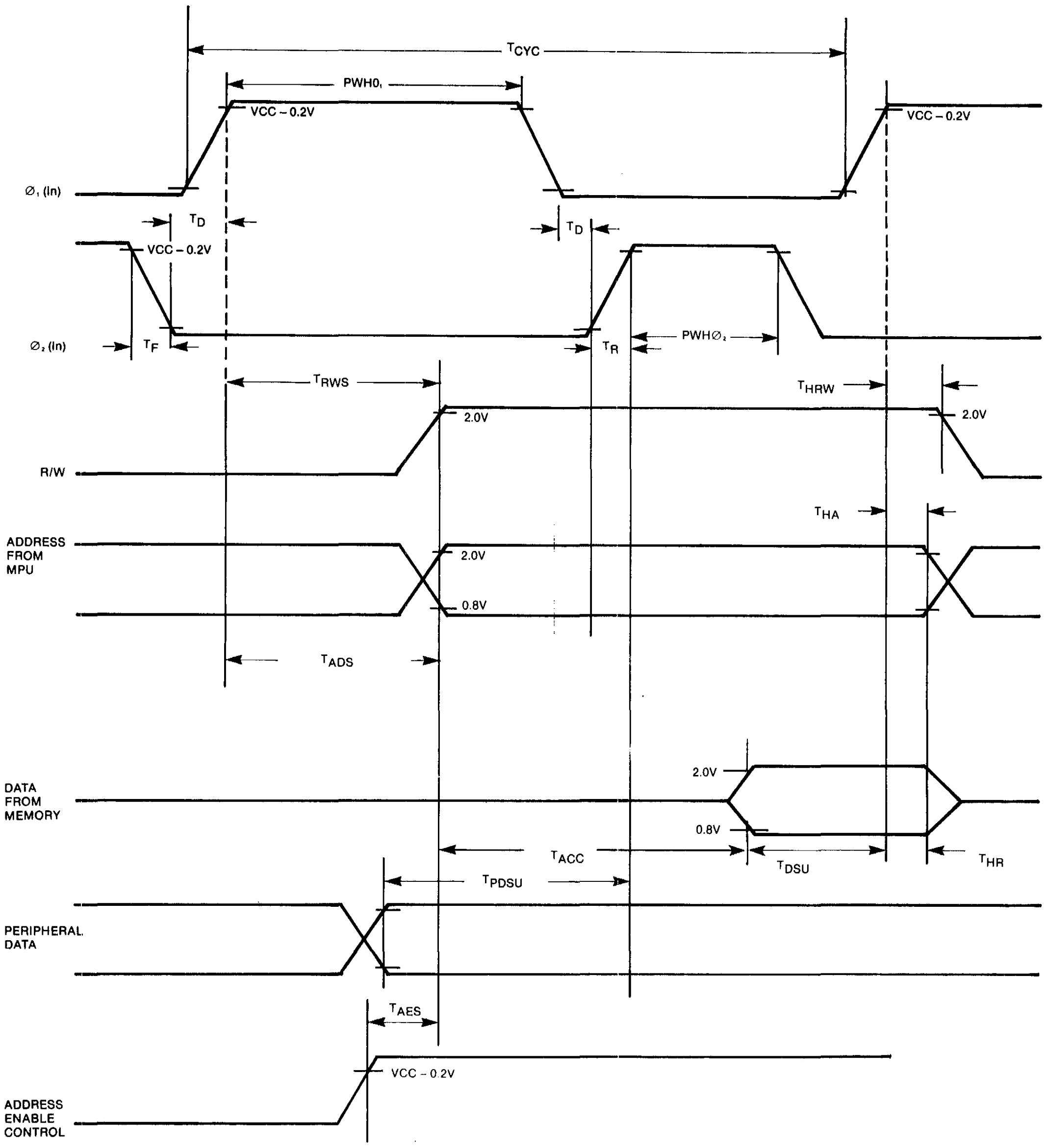
RATING	SYMBOL	VALUE	UNIT
SUPPLY VOLTAGE	Vcc	−0.3 to +7.0	Vdc
INPUT VOLTAGE	Vin	−0.3 to +7.0	Vdc
OPERATING TEMPERATURE	T_A	0 to +70	°C
STORAGE TEMPERATURE	T_{STG}	−55 to +150	°C

This device contains input protection against damage due to high static voltages or electric fields; however, precautions should be taken to avoid application of voltages higher than the maximum rating.

ELECTRICAL CHARACTERISTICS (Vcc = 5.0V ± 5%, Vss = 0, T_A = 0° to + 70°C)

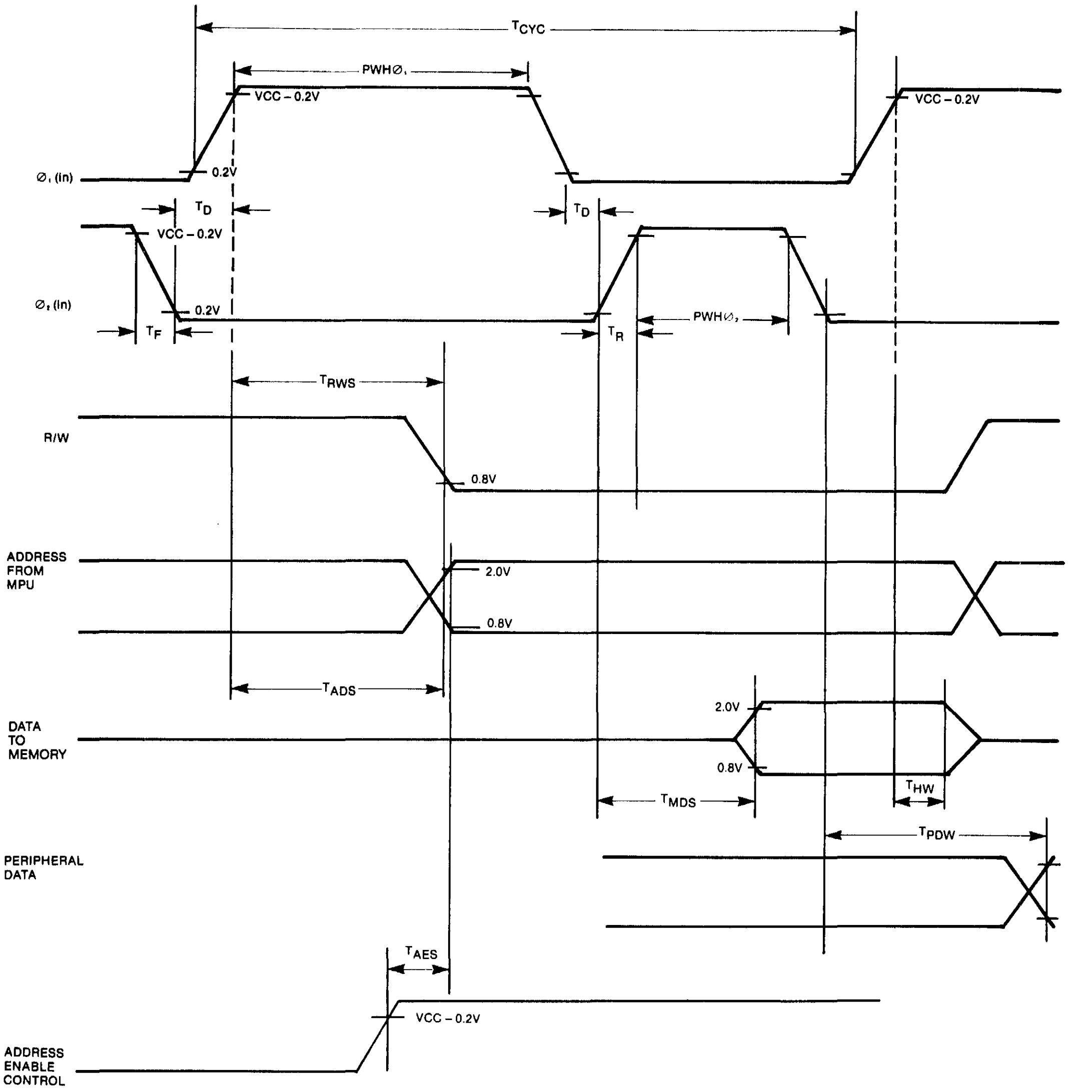
CHARACTERISTIC	SYMBOL	MIN.	TYP.	MAX.	UNIT
Input High Voltage $\varnothing_1$, $\varnothing_2$(in)	VIH	Vcc - 0.2	—	Vcc + 1.0V	Vdc
Input High Voltage $\overline{RES}$, P_0-P_7,$\overline{IRQ}$, Data		Vss + 2.0	—	—	Vdc
Input Low Voltage $\varnothing_1$, $\varnothing_2$(in)	VIL	Vss - 0.3	—	Vss + 0.2	Vdc
$\overline{RES}$, P_0-P_7 $\overline{IRQ}$, Data		—	—	Vss + 0.8	Vdc
Input Leakage Current (V_{in} = 0 to 5.25V, Vcc = 5.25V) Logic	Iin	—	—	2.5	μA
$\varnothing_1$, $\varnothing_2$(in)		—	—	100	μA
Three State (Off State) Input Current (V_{in} = 0.4 to 2.4V, Vcc = 5.25V) Data Lines	ITSI	—	—	10	μA
Output High Voltage (I_{OH} = -100μAdc, Vcc = 4.75V) Data, AO-A15, R/W, P_0-P_7	VOH	Vss + 2.4	—	—	Vdc
Out Low Voltage (I_{OL} = 1.6mAdc, Vcc = 4.75V) Data, A0-A15, R/W, P_0-P_7	VOL	—	—	Vss + 0.4	Vdc
Power Supply Current	ICC	—	125		mA
Capacitance V_{in} = O, T_A = 25°C, f = 1MHz)	C				pF
Logic, P_0-P_7	C_{in}	—	—	10	
Data		—	—	15	
AO-A15, R/W	C_{out}	—	—	12	
$\varnothing_1$	$C_{\varnothing_1}$	—	30	50	
$\varnothing_2$	$C_{\varnothing_2}$	—	50	80	

CLOCK TIMING



TIMING FOR READING DATA FROM
MEMORY OR PERIPHERALS

CLOCK TIMING



TIMING FOR WRITING DATA TO
MEMORY OR PERIPHERALS

AC CHARACTERISTICS

1 MHz TIMING **2 MHz TIMING**

ELECTRICAL CHARACTERISTICS (VCC = 5V ± 5%, VSS = 0V, $T_A = 0° - 70°C$)
Minimum Clock Frequency = 50 KHz

CLOCK TIMING

CHARACTERISTIC	SYMBOL	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	UNITS
Cycle Time	T_{CYC}	1000	—	—	500	—	—	ns
Clock Pulse Width ∅1 (Measured at VCC − 0.2V) ∅2	PWH∅1 PWH∅2	430 470	— —	— —	215 235	— —	— —	ns ns
Fall Time, Rise Time (Measured from 0.2V to VCC − 0.2V)	T_F, T_R	—	—	25	—	—	15	ns
Delay Time between Clocks (Measured at 0.2V)	T_D	0	—	—	0	—	—	ns

READ/WRITE TIMING (LOAD = 1TTL)

CHARACTERISTIC	SYMBOL	MIN.	TYP.	MAX.	MIN.	TYP.	MAX.	UNITS
Read/Write Setup Time from 6508	T_{RWS}	—	100	300	—	100	150	ns
Address Setup Time from 6508	T_{ADS}	—	100	300	—	100	150	ns
Memory Read Access Time	T_{ACC}	—	—	575	—	—	300	ns
Data Stability Time Period	T_{DSU}	100	—	—	50			ns
Data Hold Time-Read	T_{HR}		—	—				ns
Data Hold Time-Write	T_{HW}	10	30	—	10	30		ns
Data Setup Time from 6510	T_{MDS}	—	150	200	—	75	100	ns
Address Hold Time	T_{HA}	10	30	—	10	30		ns
R/W Hold Time	T_{HRW}	10	30	—	10	30		ns
Delay Time, Address valid to ∅2 positive transition	T_{AEW}	180	—	—				ns
Delay Time, ∅2 negative transition to Peripheral Data valid	T_{PDW}	—	—	1			0.5	µs
Peripheral Data Setup Time	T_{PDSU}	300	—	—				ns
Address Enable Setup Time	T_{AES}			60			60	ns

SIGNAL DESCRIPTION

Clocks ($\varnothing_1$, $\varnothing_2$)

The 6510 requires a two phase non-overlapping clock that runs at the Vcc voltage level.

Address Bus (A_0-A_{15})

The three state outputs are TTL compatible, capable of driving one standard TTL load and 130 pf.

Data Bus (D_0-D_7)

Eight pins are used for the data bus. This is a Bi-Directional bus, transferring data to and from the device and peripherals. The outputs are tri-state buffers capable of driving one standard TTL load and 130 pf.

Reset

This input is used to reset or start the microprocessor from a power down condition. During the time that this line is held low, writing to or from the microprocessor is inhibited. When a positive edge is detected on the input, the microprocessor will immediately begin the reset sequence.

After a system initialization time of six clock cycles, the mask interrupt flag will be set and the microprocessor will load the program counter from the memory vector locations FFFC and FFFD. This is the start location for program control.

After Vcc reaches 4.75 volts in a power up routine, reset must be held low for at least two clock cycles. At this time the R/W signal will become valid.

When the reset signal goes high following these two clock cycles, the microprocessor will proceed with the normal reset procedure detailed above.

Interrupt Request ($\overline{\text{IRQ}}$)

This TTL level input requests that an interrupt sequence begin within the microprocessor. The microprocessor will complete the current instruction being executed before recognizing the request. At that time, the interrupt mask bit in the Status Code Register will be examined. If the interrupt mask flag is not set, the microprocessor will begin an interrupt sequence. The Program Counter and Processor Status Register are stored in the stack. The microprocessor will then set the interrupt mask flag high so that no further interrupts may occur. At the end of this cycle, the program counter low will be loaded from address FFFE, and program counter high from location FFFF, therefore transferring program control to the memory vector located at these addresses.

Address Enable Control (AEC)

The Address Bus is valid only when the Address Enable Control line is high. When low, the Address Bus is in a high-impedance state. This feature allows easy DMA and multiprocessor systems.

I/O Port (P_0-P_7)

Eight pins are used for the peripheral port, which can transfer data to or from peripheral devices. The Output Register is located in RAM at Address 0001, and the Data Direction Register is at Address 0000. The outputs are capable at driving one standard TTL load and 130 pf.

Read/Write (R/W)

This signal is generated by the microprocessor to control the direction of data transfers on the Data Bus. This line is high except when the microprocessor is writing to memory or a peripheral device.

ADDRESSING MODES

ACCUMULATOR ADDRESSING—This form of addressing is represented with a one byte instruction, implying an operation on the accumulator.

IMMEDIATE ADDRESSING—In immediate addressing, the operand is contained in the second byte of the instruction, with no further memory addressing required.

ABSOLUTE ADDRESSING—In absolute addressing, the second byte of the instruction specifies the eight low order bits of the effective address while the third byte specifies the eight high order bits. Thus, the absolute addressing mode allows access to the entire 65K bytes of addressable memory.

ZERO PAGE ADDRESSING—The zero page instructions allow for shorter code and execution times by only fetching the second byte of the instruction and assuming a zero high address byte. Careful use of the zero page can result in significant increase in code efficiency.

INDEXED ZERO PAGE ADDRESSING—(X, Y indexing)—This form of addressing is used in conjunction with the index register and is referred to as "Zero Page, X" or "Zero Page, Y." The effective address is calculated by adding the second byte to the contents of the index register. Since this is a form of "Zero Page" addressing, the content of the second byte references a location in page zero. Additionally, due to the "Zero Page" addressing nature of this mode, no carry is added to the high order 8 bits of memory and crossing of page boundaries does not occur.

INDEXED ABSOLUTE ADDRESSING—(X, Y indexing)—This form of addressing is used in conjunction with X and Y index register and is referred to as "Absolute, X," and "Absolute, Y." The effective address is formed by adding the contents of X and Y to the address contained in the second and third bytes of the instruction. This mode allows the index register to contain the index or count value and the instruction to contain the base address. This type of indexing allows any location referencing and the index to modify multiple fields resulting in reduced coding and execution time.

IMPLIED ADDRESSING—In the implied addressing mode, the address containing the operand is implicitly stated in the operation code of the instruction.

RELATIVE ADDRESSING—Relative addressing is used only with branch instructions and establishes a destination for the conditional branch.

The second byte of the instruction becomes the operand which is an "Offset" added to the contents of the lower eight bits of the program counter when the counter is set at the next instruction. The range of the offset is −128 to +127 bytes from the next instruction.

INDEXED INDIRECT ADDRESSING—In indexed indirect addressing (referred to as [Indirect, X]), the second byte of the instruction is added to the contents of the X index register, discarding the carry. The result of this addition points to a memory location on page zero whose contents is the low order eight bits of the effective address. The next memory location in page zero contains the high order eight bits of the effective address. Both memory locations specifying the high and low order bytes of the effective address must be in page zero.

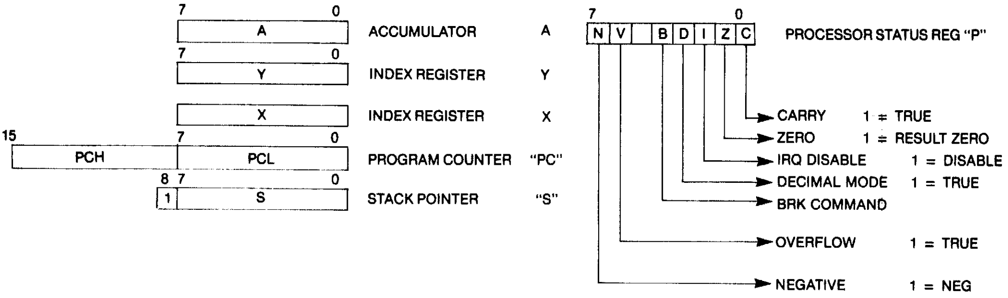
INDIRECT INDEXED ADDRESSING—In indirect indexed addressing (referred to as [Indirect, Y]), the second byte of the instruction points to a memory location in page zero. The contents of this memory location is added to the contents of the Y index register, the result being the low order eight bits of the effective address. The carry from this addition is added to the contents of the next page zero memory location, the result being the high order eight bits of the effective address.

ABSOLUTE INDIRECT—The second byte of the instruction contains the low order eight bits of a memory location. The high order eight bits of that memory location is contained in the third byte of the instruction. The contents of the fully specified memory location is the low order byte of the effective address. The next memory location contains the high order byte of the effective address which is loaded into the sixteen bits of the program counter.

INSTRUCTION SET—ALPHABETIC SEQUENCE

ADC	Add Memory to Accumulator with Carry
AND	"AND" Memory with Accumulator
ASL	Shift left One Bit (Memory or 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 One Bit Right (Memory or Accumulator)
NOP	No Operation
ORA	"OR" Memory with Accumulator
PHA	Push Accumulator on Stack
PHP	Push Processor Status on 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 Register
TYA	Transfer Index Y to Accumulator

PROGRAMMING MODEL



INSTRUCTION SET — OP CODES, Execution Time, Memory Requirements

INSTRUCTIONS		IMMEDIATE			ABSOLUTE			ZERO PAGE			ACCUM			IMPLIED			(IND, X)			(IND), Y			Z PAGE, X			ABS, X			ABS, Y			RELATIVE			INDIRECT			Z PAGE, Y			CONDITION CODES					
MNEMONIC	OPERATION	OP	N	#	OP	N	#	OP	N	#	OP	N	#	OP	N	#	OP	N	#	OP	N	#	OP	N	#	OP	N	#	OP	N	#	OP	N	#	OP	N	#	OP	N	#	N	Z	C	I	D	V
ADC	A+M+C→A (1)	69	2	2	6D	4	3	65	3	2							61	6	2	71	5	2	75	4	2	7D	4	3	79	4	3										✓	✓	✓	.	.	✓
AND	AΛM→A (1)	29	2	2	2D	4	3	25	3	2							21	6	2	31	5	2	35	4	2	3D	4	3	39	4	3										✓	✓	.	.	.	.
ASL	C ← [7 0] ← 0				0E	6	3	06	5	2	0A	2	1										16	6	2	1E	7	3													✓	✓	✓	.	.	.
BCC	BRANCH ON C=0 (2)																															90	2	2							.	.	.	.	.	.
BCS	BRANCH ON C=1 (2)																															B0	2	2							.	.	.	.	.	.
BEQ	BRANCH ON Z=1 (2)																															F0	2	2							.	.	.	.	.	.
BIT	A M				2C	4	3	24	3	2																															M_7	✓	.	.	.	M_6
BMI	BRANCH ON N=1 (2)																															30	2	2							.	.	.	.	.	.
BNE	BRANCH ON Z=0 (2)																															D0	2	2							.	.	.	.	.	.
BPL	BRANCH ON N=0 (2)																															10	2	2							.	.	.	.	.	.
BRK	(See Fig. 1)													00	7	1																									.	.	.	1	.	.
BVC	BRANCH ON V=0 (2)																															50	2	2							.	.	.	.	.	.
BVS	BRANCH ON V=1 (2)																															70	2	2							.	.	.	.	.	.
CLC	0→C													18	2	1																									.	.	0	.	.	.
CLD	0→D													D8	2	1																									.	.	.	.	0	.
CLI	0→I													58	2	1																									.	.	.	0	.	.
CLV	0→V													B8	2	1																									.	.	.	.	.	0
CMP	A−M (1)	C9	2	2	CD	4	3	C5	3	2							C1	6	2	D1	5	2	D5	4	2	DD	4	3	D9	4	3										✓	✓	✓	.	.	.
CPX	X−M	E0	2	2	EC	4	3	E4	3	2																															✓	✓	✓	.	.	.
CPY	Y−M	C0	2	2	CC	4	3	C4	3	2																															✓	✓	✓	.	.	.
DEC	M−1→M				CE	6	3	C6	5	2													D6	6	2	DE	7	3													✓	✓	.	.	.	.
DEX	X−1→X													CA	2	1																									✓	✓	.	.	.	.
DEY	Y−1→Y													88	2	1																									✓	✓	.	.	.	.
EOR	A⊻M→A (1)	49	2	2	4D	4	3	45	3	2							41	6	2	51	5	2	55	4	2	5D	4	3	59	4	3										✓	✓	.	.	.	.
INC	M+1→M				EE	6	3	E6	5	2													F6	6	2	FE	7	3													✓	✓	.	.	.	.
INX	X+1→X													E8	2	1																									✓	✓	.	.	.	.
INY	Y+1→Y													C8	2	1																									✓	✓	.	.	.	.
JMP	JUMP TO NEW LOC				4C	3	3																												6C	5	3				.	.	.	.	.	.
JSR	(See Fig. 2) JUMP SUB				20	6	3																																		.	.	.	.	.	.
LDA	M→A (1)	A9	2	2	AD	4	3	A5	3	2							A1	6	2	B1	5	2	B5	4	2	BD	4	3	B9	4	3										✓	✓	.	.	.	.
LDX	M→X (1)	A2	2	2	AE	4	3	A6	3	2																			BE	4	3							B6	4	2	✓	✓	.	.	.	.
LDY	M→Y (1)	A0	2	2	AC	4	3	A4	3	2													B4	4	2	BC	4	3													✓	✓	.	.	.	.
LSR	0 → [7 0] → C				4E	6	3	46	5	2	4A	2	1										56	6	2	5E	7	3													0	✓	✓	.	.	.
NOP	NO OPERATION													EA	2	1																									.	.	.	.	.	.
ORA	AVM A	09	2	2	0D	4	3	05	3	2							01	6	2	11	5	2	15	4	2	1D	4	3	19	4	3										✓	✓	.	.	.	.
PHA	A→M_S S−1→S													48	3	1																									.	.	.	.	.	.
PHP	P→M_S S−1→S													08	3	1																									.	.	.	.	.	.
PLA	S+1→S M_S→A													68	4	1																									✓	✓	.	.	.	.
PLP	S+1→S M_S→P													28	4	1																									(RESTORED)					
ROL	[7 0] ← [C]				2E	6	3	26	5	2	2A	2	1										36	6	2	3E	7	3													✓	✓	✓	.	.	.
ROR	[C] → [7 0]				6E	6	3	66	5	2	6A	2	1										76	6	2	7E	7	3													✓	✓	✓	.	.	.
RTI	(See Fig. 1) RTRN INT													40	6	1																									(RESTORED)					
RTS	(See Fig. 2) RTRN SUB													60	6	1																									.	.	.	.	.	.
SBC	A−M−C→A (1)	E9	2	2	ED	4	3	E5	3	2							E1	6	2	F1	5	2	F5	4	2	FD	4	3	F9	4	3										✓	✓	(3)	.	.	✓
SEC	1→C													38	2	1																									.	.	1	.	.	.
SED	1→D													F8	2	1																									.	.	.	.	1	.
SEI	1→I													78	2	1																									.	.	.	1	.	.
STA	A→M				8D	4	3	85	3	2							81	6	2	91	6	2	95	4	2	9D	5	3	99	5	3										.	.	.	.	.	.
STX	X→M				8E	4	3	86	3	2																												96	4	2	.	.	.	.	.	.
STY	Y→M				8C	4	3	84	3	2													94	4	2																.	.	.	.	.	.
TAX	A→X													AA	2	1																									✓	✓	.	.	.	.
TAY	A→Y													A8	2	1																									✓	✓	.	.	.	.
TSX	S→X													BA	2	1																									✓	✓	.	.	.	.
TXA	X→A													8A	2	1																									✓	✓	.	.	.	.
TXS	X→S													9A	2	1																									.	.	.	.	.	.
TYA	Y→A													98	2	1																									✓	✓	.	.	.	.

(1) ADD 1 TO N IF PAGE BOUNDARY IS CROSSED
(2) ADD 1 TO N IF BRANCH OCCURS TO SAME PAGE
ADD 2 TO N IF BRANCH OCCURS TO DIFFERENT PAGE
(3) CARRY NOT = BORROW
(4) IF IN DECIMAL MODE Z FLAG IS INVALID
ACCUMULATOR MUST BE CHECKED FOR ZERO RESULT

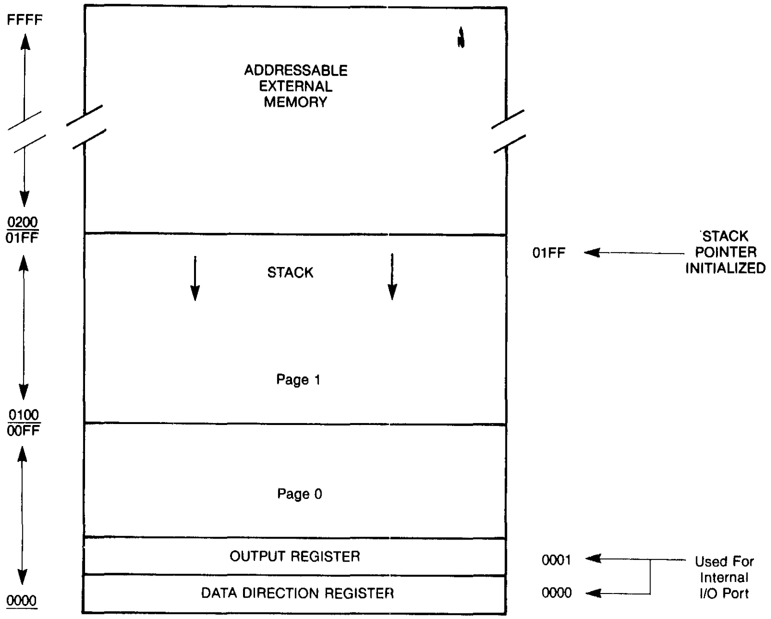
X INDEX, X
Y INDEX, Y
A ACCUMULATOR
M MEMORY PER EFFECTIVE ADDRESS
M_S MEMORY PER STACK POINTER

\+ ADD
− SUBTRACT
Λ AND
V OR

⊻ EXCLUSIVE OR
✓ MODIFIED
· NOT MODIFIED
M_7 MEMORY BIT 7
M_6 MEMORY BIT 6

N NO CYCLES
\# NO BYTES

Note: Commodore Semiconductor Group cannot assume liability for the use of undefined OP Codes



6510 MEMORY MAP

APPLICATIONS NOTES

Locating the Output Register at the internal I/O Port in Page Zero enhances the powerful Zero Page Addressing instructions of the 6510.

By assigning the I/O Pins as inputs (using the Data Direction Register) the user has the ability to change the contents of address 0001 (the Output Register) using peripheral devices. The ability to change these contents using peripheral inputs, together with Zero Page Indirect Addressing instructions, allows novel and versatile programming techniques not possible earlier.