OPERATION CODE TABLE

LSD / MSD	0	1	2	3	4	5	6	7
0	BRK	ORA-IND. X				ORA-Z PAGE	ASL-Z PAGE	
1	BPL	ORA-IND. Y				ORA-Z PAGE, X	ASL-Z PAGE, X	
2	JSR	AND-IND. X			BIT-Z. Page	AND-Z PAGE	ROL-Z PAGE	
3	BMI	AND-IND. Y				AND-Z PAGE X	ROL-Z PAGE X	
4	RTI	EOR-IND. X				EOR-Z PAGE	LSR-Z PAGE	
5	BVC	EOR-IND. Y				EOR-Z PAGE, X	LSR-Z PAGE, X	
6	RTS	ADC-IND. X				ADC-Z PAGE	ROR-Z PAGE	
7	BVS	ADC-IND. Y				ADC-Z PAGE, X		
8		STA-IND. X			STY-Z. Page	STA-Z PAGE	STX-Z PAGE	
9	BCC	STA-IND. Y			STY-Z. Page, X	STA-Z PAGE, X	STX-Z PAGE, Y	
A	LDY-IMM	LDA-IND. X	LDX-IMM		LDY-Z. Page	LDA-Z PAGE	LDX-Z PAGE	
B	BCS	LDA-IND. Y			LDY-Z. Page, X	LDA-Z PAGE, X	LDX-Z PAGE, Y	
C	CPY-IMM	CMP-IND. X			CPY-Z. Page	CMP-Z PAGE	DEC-Z PAGE	
D	BNE	CMP-IND. Y				CMP-Z PAGE, X	DEC-Z PAGE, X	
E	CPX-IMM	SBC-IND. X			CPX-Z. Page	SBC-Z PAGE	INC-Z PAGE	
F	BEQ	SBC-IND. Y				SBC-Z PAGE, X	INC-Z PAGE, X	

8	9	A	B	C	D	E	F	LSD / MSD
PHP	ORA-IMM	ASL-A			ORA-ABS	ASL-ABS		0
CLC	ORA-ABS, Y				ORA-ABS, X	ASL-ABS, X		1
PLP	AND-IMM	ROL-A		BIT-ABS	AND-ABS	ROL-ABS		2
SEC	AND-ABS, Y				AND-ABS, X	ROL-ABS, X		3
PHA	EOR-IMM	LSR-A		JUMP-ABS	EOR-ABS	LSR-ABS		4
CLI	EOR-ABS, Y				EOR-ABS, X	LSR-ABS, X		5
PLA	ADC-IMM	ROR-A		JMP-IND	ADC-ABS	ROR-ABS		6
SEI	ADC-ABS, Y				ADC-ABS, X			7
DEY		TXA		STY-ABS	STA-ABS	STX-ABS		8
TYA	STA-ABS, Y	TXS			STA-ABS, X			9
TAY	LDA-IMM	TAX		LDY-ABS	LDA-ABS	LDX-ABS		A
CLV	LDA-ABS, Y	TSX		LDY-ABS, X	LDA-ABS, X	LDX-ABS, Y		B
INY	CMP-IMM	DEX		CPY-ABS	CMP-ABS	DEC-ABS		C
CLD	CMP-ABS, Y				CMP-ABS, X	DEC-ABS, X		D
INX	SBC-IMM	NOP		CPX-ABS	SBC-ABS	INC-ABS		E
SED	SBC-ABS, Y				SBC-ABS, X	INC-ABS, X		F

ADDRESSING MODES

IMM — IMMEDIATE ADDRESSING — The operand is contained in the second byte of the instruction.

ABS — ABSOLUTE ADDRESSING — The second byte of the instruction contains the 8 low order bits of the effective address (EA). The third byte contains the 8 high order bits of the effective address.

Z PAGE — ZERO PAGE ADDRESSING — Second byte contains the 8 low order bits of the effective address. The 8 high order bits are zero.

A — ACCUMULATOR — One byte instruction operating on the accumulator.

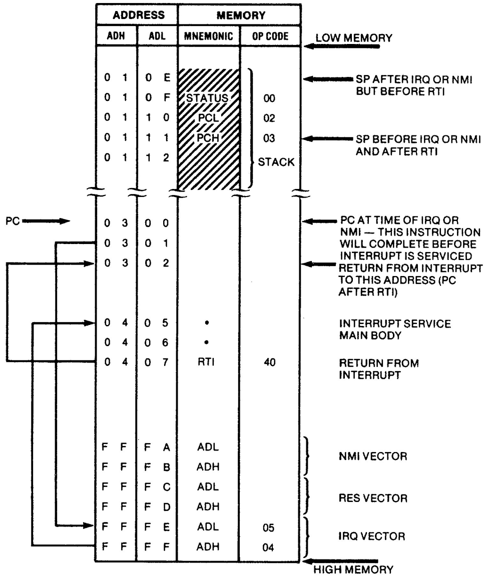
Z PAGE, X - Z PAGE, Y — ZERO PAGE INDEXED — The second byte of the instruction is added to the index (carry is dropped) to form the low order byte of the EA. The high order byte of the EA is zeros.

ABS, X - ABS, Y — ABSOLUTE INDEXED — The effective address is formed by adding the index to the second and third byte of the instruction.

(IND, X) — INDEXED INDIRECT — The second byte of the instruction is added to the X index, discarding the carry. The result points to a location on page zero which contains the 8 low order bits of the EA. The next byte contains the 8 high order bits.

(IND), Y — INDIRECT INDEXED — The second byte of the instruction points to a location in page zero. The contents of this memory location is added to the Y index, the result being the low order eight bits of the EA. The carry from this operation is added to the contents of the next page zero location, the result being the 8 high order bits of the EA.

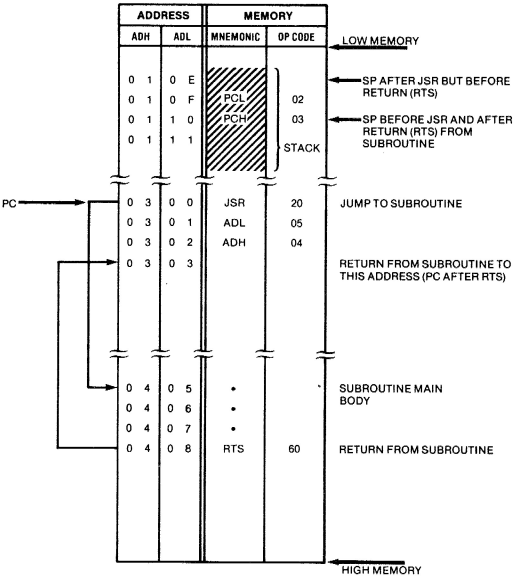
FIG. 1 IRQ, NMI, RTI, BRK OPERATION



ADDRESS		MEMORY	
ADH	ADL	MNEMONIC	OP CODE
0 1	0 E		
0 1	0 F	STATUS	00
0 1	1 0	PCL	02
0 1	1 1	PCH	03
0 1	1 2		STACK
0 3	0 0		
0 3	0 1		
0 3	0 2		
0 4	0 5	•	
0 4	0 6	•	
0 4	0 7	RTI	40
F F	F A	ADL	
F F	F B	ADH	
F F	F C	ADL	
F F	F D	ADH	
F F	F E	ADL	05
F F	F F	ADH	04

LOW MEMORY
SP AFTER IRQ OR NMI BUT BEFORE RTI
SP BEFORE IRQ OR NMI AND AFTER RTI
PC
PC AT TIME OF IRQ OR NMI — THIS INSTRUCTION WILL COMPLETE BEFORE INTERRUPT IS SERVICED
RETURN FROM INTERRUPT TO THIS ADDRESS (PC AFTER RTI)
INTERRUPT SERVICE MAIN BODY
RETURN FROM INTERRUPT
NMI VECTOR
RES VECTOR
IRQ VECTOR
HIGH MEMORY

FIG. 2 JSR, RTS OPERATION



ADDRESS		MEMORY	
ADH	ADL	MNEMONIC	OP CODE
0 1	0 E		
0 1	0 F	PCL	02
0 1	1 0	PCH	03
0 1	1 1		STACK
0 3	0 0	JSR	20
0 3	0 1	ADL	05
0 3	0 2	ADH	04
0 3	0 3		
0 4	0 5	•	
0 4	0 6	•	
0 4	0 7	•	
0 4	0 8	RTS	60

LOW MEMORY
SP AFTER JSR BUT BEFORE RETURN (RTS)
SP BEFORE JSR AND AFTER RETURN (RTS) FROM SUBROUTINE
PC
JUMP TO SUBROUTINE
RETURN FROM SUBROUTINE TO THIS ADDRESS (PC AFTER RTS)
SUBROUTINE MAIN BODY
RETURN FROM SUBROUTINE
HIGH MEMORY