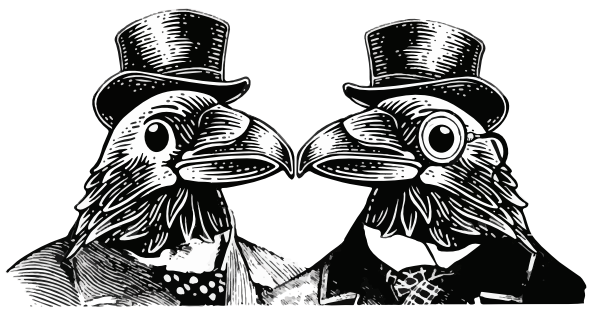
6502 & 65816 Instructions

Name and Description	Addressing Modes	Op-Codes	Status NVMXDIZC
PER Push effective program counter relative address	**PER Rel**	**62 --**	**--------**
PHA Push accumulator onto stack	PHA	48	--------
PHB Push data bank register onto stack	**PHB**	**48**	**--------**
PHD Push direct page register onto stack	**PHD**	**0B**	**--------**
PHK Push pgm bank register onto stack	**PHK**	**4B**	**--------**
PHP Push processor status on stack	PHP	08	--------
PHX Push index X onto stack	• PHX	DA	--------
PHY Push index Y onto stack	• PHY	5A	--------
PLA Pull accumulator from stack	PLA	68	N-----Z-
PLB Pull data bank register from stack	**PLB**	**AB**	**N-----Z-**
PLD Pull direct page register from stack	**PLD**	**2B**	**N-----Z-**
PLP Pull processor status register from stack	PLP	28	NVMXDIZC
PLX Pull index X from stack	• PLX	FA	N-----Z-
PLY Pull index Y from stack	• PLY	7A	N-----Z-
REP Reset processor status register bits	**REP #Imm**	**C2 -**	**NVMXDIZC**
ROL Rotate left one bit (Memory or Accumulator)	ROL ZP	26 -	N-----ZC
	ROL A	2A	N-----ZC
	ROL Addr	2E --	N-----ZC
	ROL ZP,X	36 -	N-----ZC
	ROL Addr,X	3E --	N-----ZC
ROR Rotate right one bit (Memory or Accumulator)	ROR ZP	66 -	N-----ZC
	ROR A	6A	N-----ZC
	ROR Addr	6E --	N-----ZC
	ROR ZP,X	66 -	N-----ZC
	ROR Addr,X	6E --	N-----ZC
RTI Return from interrupt	RTI	40	NVMXDIZC
RTL Return from subroutine long	**RTL**	**6B**	**--------**
RTS Return from subroutine	RTS	60	--------
SBC Subtract memory from accumulator with borrow	SBC (ZP,X)	E1 -	NV----ZC
	SBC SR, S	**E3 -**	**NV----ZC**
	SBC ZP	E5 -	NV----ZC
	SBC [ZP]	**E7 -**	**NV----ZC**
	SBC #Imm	E9 -	NV----ZC
	SBC Addr	ED --	NV----ZC
	SBC LongAddr	**EF ---**	**NV----ZC**
	SBC (ZP),Y	F1 --	NV----ZC
	• SBC (ZP)	F2 -	NV----ZC
	SBC (SR, S), Y	**F3 -**	**NV----ZC**
	SBC ZP,X	F5 -	NV----ZC
	SBC [ZP],Y	**F7 -**	**NV----ZC**
	SBC Addr,Y	F9 --	NV----ZC
	SBC Addr,X	FD --	NV----ZC
	SBC LongAddr,X	**FF ---**	**NV----ZC**
SEC Set carry flag	SEC	38	-------C
SED Set decimal flag	SED	F8	----D---
SEI Set interrupt disable flag	SEI	78	-----I--
SEP Set processor status bits	SEP	E2	NVMXDIZC
STA Store accumulator to memory	STA (ZP,X)	81 -	--------
	STA SR, S	**83 -**	**--------**
	STA ZP	85 -	--------
	STA [ZP]	**87 -**	**--------**
	STA Addr	8D --	--------
	STA LongAddr	**8F ---**	**--------**
	STA (ZP),Y	91 --	--------
	• STA (ZP)	92 -	--------
	STA (SR, S), Y	**93 -**	**--------**
	STA ZP,X	95 -	--------
	STA [ZP],Y	**97 -**	**--------**
	STA Addr,Y	99 --	--------
	STA Addr,X	9D --	--------
	STA LongAddr,X	**9F ---**	**--------**
STP Stop processor	**STP**	**DB**	**--------**
STX Store index X to memory	STX ZP	86 -	--------
	STX Addr	8E --	--------
	STX ZP,Y	96 -	--------
STY Store index Y to memory	STY ZP	84 -	--------
	STY Addr	8C --	--------
	STY ZP,X	94 -	--------
STZ Store zero to memory	• STZ ZP	64 -	--------
	• STZ ZP,X	74 -	--------
	• STZ Addr	9C --	--------
	• STZ Addr,X	9E --	--------
TAX Transfer accumulator to index X	TAX	AA	N-----Z-
TAY Transfer accumulator to index Y	TAY	A8	N-----Z-
TCD Transfer 16-bit accumulator to direct page reg	**TCD**	**5B**	**N-----Z-**
TCS Transfer 16-bit accumulator to stack pointer	**TCS**	**1B**	**--------**
TDC Transfer direct page reg to 16-bit accumulator	**TDC**	**7B**	**N-----Z-**
TRB Test and reset bits against accumulator	• TRB ZP	14 -	------Z-
	• TRB Addr	1C --	------Z-
TSB Test and set bits against accumulator	• TSB ZP	04 -	------Z-
	• TSB Addr	0C --	------Z-
TSC Transfer stack pointer to 16-bit accumulator	**TSC**	**3B**	**N-----Z-**
TSX Transfer stack pointer to index X	TSX	BA	N-----Z-
TXA Transfer index X to accumulator	TXA	8A	N-----Z-
TXS Transfer index X to stack pointer	TXS	9A	--------
TXY Transfer index X to index Y	**TXY**	**9B**	**N-----Z-**
TYA Transfer index Y to accumulator	TXA	98	N-----Z-
TYX Transfer index Y to index X	TYX	BB	N-----Z-
WDM Reserved for future expansion	**WDM**	**42**	**--------**
XBA Exchange B and A 8-bit accumulators	**XBA**	**EB**	**N-----Z-**
XCE Exchange carry and emulation flags	**XCE**	**FB**	**--MX---CE**



Crow™ Cousins micro software

Entries with bullets (•) are present in the 65C02 and the 65816

BOLDED entries are 65816-specific and not applicable to previous 6502 processors

The dashes following opcodes indicate the total length of the instruction. e.g. 2 dashes = a 3 byte instruction