COMMODORE SEMICONDUCTOR GROUP
a division of Commodore Business Machines, Inc.
950 Rittenhouse Rd., Norristown, PA 19403 • 215/666-7950 • TWX 510-660-4168 • FAX 215/647-0791

CMOS

65CE02 MICROPROCESSOR

DESCRIPTION

The Commodore 65CE02 is an enhanced version of the popular 8-bit 6502, designed with entirely new internal architecture and manufactured in 2-micron, double-level-metal, CMOS technology for high speed and low power consumption. The 65CE02 is code (*) and pin compatible with existing 6502/65C02's.

The instruction set has been streamlined, removing most 'dead' cycles which occurred due to page boundaries and micro-code pipelines, allowing existing code to run up to 25% faster. Additional instructions and addressing modes allow even greater program efficiency. Add to this operational speeds of up to 10MHz (100ns instruction cycles) and the 65CE02 is capable of a 350% decrease in program execution time compared to a standard 4MHz 6502.

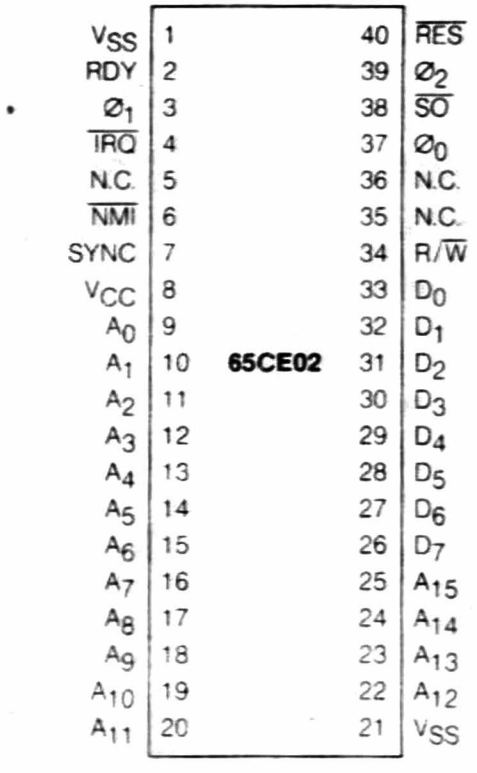
The 65CE02 provides the system designer with a high performance, low power microprocessor, while retaining downward compatibility with the existing family of microprocessor support devices.

(*) Application programs should be analyzed to see if they contain timing loops or previously undefined opcodes.

FEATURES

- CMOS technology for low power consumption
- 0 - 10 MHz operating speeds
- Single +5V supply required
- Eight-bit parallel processing
- Code compatible with existing 6502/65C02
- Execution times independent of page boundary crossings
- Executes existing 6502/65C02 code in up to 25% fewer cycles
- 16-bit stack pointer with two modes of operation
  - full 16-bit pointer
  - page programmable 8-bit pointer
- Base page register allows relocation of 'zero' page
- Three index registers: X, Y, Z
- Maskable & non-maskable interrupt capability
- 64K byte memory address space
- Direct memory access (DMA) capability
- 'Ready' input
- On-chip clock generates ϕ_1/ϕ_2
- Enhanced instruction set
  - 24 new instructions, for a total of 92
  - 46 new op codes, for a total of 256
  - Three new addressing modes
- Cross-assembler and low cost hardware emulator available

FIGURE 1
PIN CONFIGURATION



PRELIMINARY

SUMMARY OF 65CE02 ENHANCEMENTS

The 65CE02, upon reset, configures itself like any present CMOS 6502 processor, with the exception that many instructions require fewer cycles. This results in programs that execute in less time than older versions, even at the same clock frequency.

The stack pointer has been expanded to 16 bits, but can be used in two different ways. It can be used as a full 16-bit (word) stack pointer, or as an 8-bit (byte) pointer whose stack page is programmable. On reset, the byte mode is selected with page 01 set as the stack page, maintaining 6502 and 65C02 compatibility.

The zero page is also programmable via a new register, the "B" or "Base Page" register. On reset, this register is cleared, thus giving a true "zero" page. The user can then re-define any page in memory as the "zero" page.

A third index register, "Z", has been added to increase flexibility in data manipulation. This register is cleared on reset, providing STZ instruction compatibility with the 65C02.

All branching instructions have been expanded to include a 'word relative' addressing mode which allows branching anywhere within the 64K memory space. A new word relative branch to subroutine aids the programmer in creating re-locatable code modules, resulting in increased software flexibility.

Also included is an addressing mode which facilitates parameter passing to subroutines. Parameters and/or pointers to data arrays can be passed to a subroutine via the stack, and a special return instruction will 'fix' the stack pointer when the subroutine is finished.

The BIT (IMMEDIATE) test will set the N and V flags with valid states, which was not the case with earlier 65C02s. The BCD arithmetic instructions modify the N, Z, V, and C flags correctly, as was not the case in the 6502.

The following is a list of opcodes that have been added to the 210 previously defined MOS, Rockwell, and GTE opcodes.

1. Branches and Jumps

BPL	label	word-relative
BMI	label	word-relative
BRU	label	word-relative (BRA)
BVC	label	word-relative
BVS	label	word-relative
BCC	label	word-relative
BCS	label	word-relative
BNE	label	word-relative
BEQ	label	word-relative
BSR	label	Branch to SubRoutine (word relative)
JSR	(ABS)	Jump to SubRoutine absolute indirect
JSR	(ABS, X)	Jump to SubRoutine absolute indirect, X
RTN	#	ReTurN from subroutine and adjust stack pointer.

2. Arithmetic Operations

NEG	A	NEGate (or 2's complement) accumulator.
ASR	A	Arithmetic Shift Right, accumulator or memory
ASR	BP	
ASR	BP, X	
INW	BP	INcrement Word
DEW	BP	DEcrement Word
INZ		INcrement Z register
DEZ		DEcrement Z register
ASW	ABS	Arithmetic Shift left Word
ROW	ABS	ROtate left Word
ORA	(BP), Z	formerly (ZP) non-indexed that
AND	(BP), Z	are now indexed by Z register
EOR	(BP), Z	
ADC	(BP), Z	
CMP	(BP), Z	
SBC	(BP), Z	
CPZ	IMM	ComPare Z register with memory immediate,
CPZ	BP	base page, and
CPZ	ABS	absolute.

3. Loads, Stores, Pushes, Pulls and Transfers

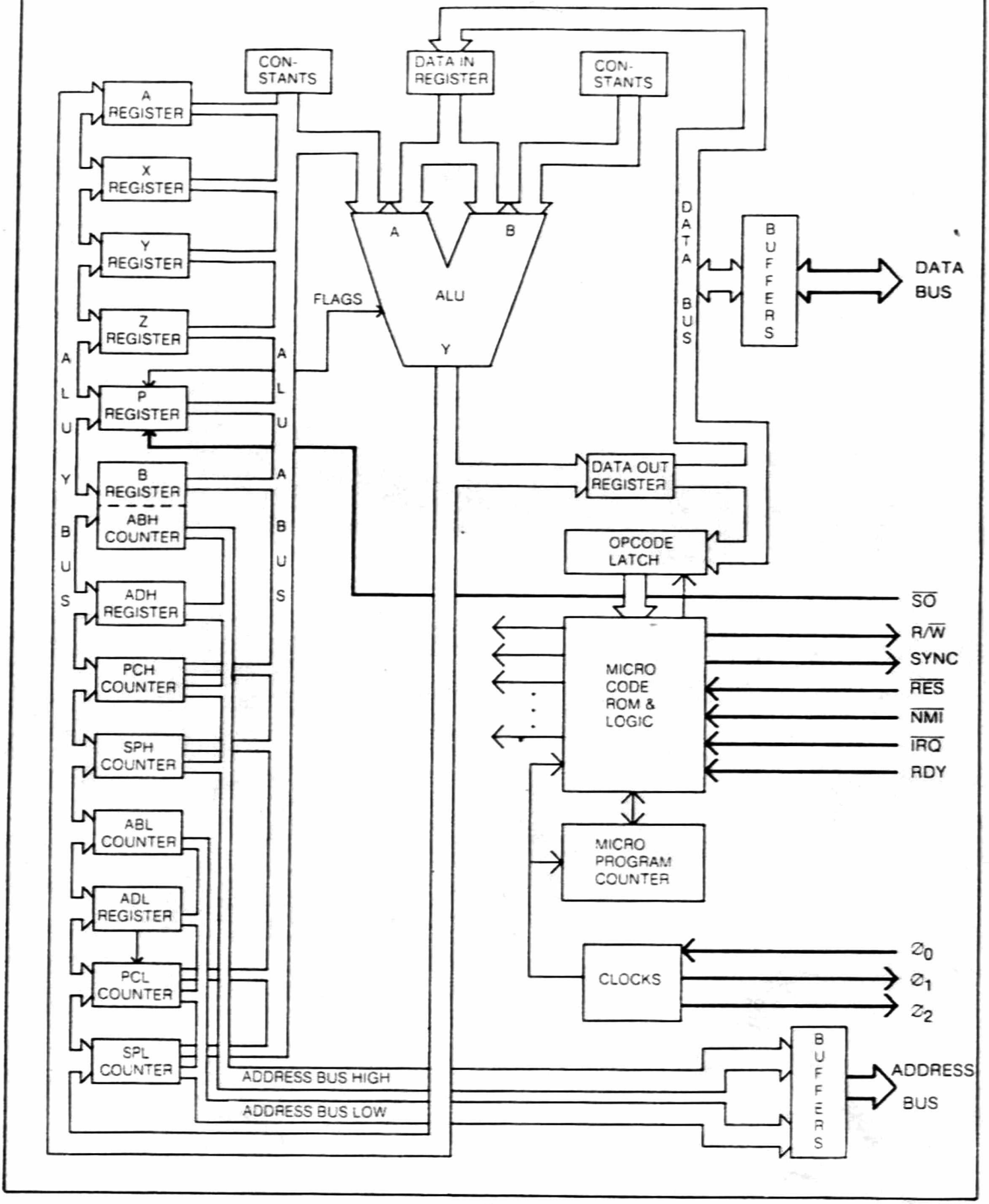
LDA	(BP), Z	formerly (ZP)
LDZ	IMM	LoaD Z register immediate,
LDZ	ABS	absolute
LDZ	ABS, X	absolute, X.
LDA	(d, SP), Y	LoaD Accum via stack vector indexed by Y
STA	(d, SP), Y	and Store
STX	ABS, Y	STore X Absolute, Y
STY	ABS, X	STore Y Absolute, X
STZ	BP	STore Z register (formerly store zero)
STZ	ABS	
STZ	BP, X	
STZ	ABS, X	
STA	(BP), Z	formerly (ZP)
CLE		CLear stack Extend disable bit
SEE		SEt stack Extend disable bit
PHW	IMM	PusH Data Immediate (Word)
PHW	ABS	PusH Data Absolute (Word)
PHZ		PusH Z register onto stack
PLZ		PulL Z register from stack
TAZ		Transfer Accumulator to Z register
TZA		Transfer Z register to Accumulator
TAB		Transfer Accumulator to Base page register
TBA		Transfer Base page register to Accumulator
TSY		Transfer Stack pointer high byte to Y register
TYS		Transfer Y register to Stack pointer high byte

4. Special Instructions

AUG	AUGment Instruction (4-byte NOP, reserved for future expansion)

FIGURE 2

65CE02 FUNCTIONAL BLOCK DIAGRAM



FUNCTIONAL DESCRIPTION

Figure 2 shows the block diagram of the 65CE02 CPU's internal architecture. This diagram supports the following description of the device's major elements.

Clock

The clock circuitry accepts the external $\varnothing_0$ clock rate and from it generates all required internal clock control signals. It provides the external $\varnothing_1$ and $\varnothing_2$ signals from which all inputs and outputs are referenced.

Micro Program Counter, Micro Code ROM and Logic PLA

This block controls and implements the opcode sequences.

Arithmetic and Logic Unit (ALU)

All arithmetic and logic operations are executed in the ALU. The ALU has no internal memory and all operational outputs are directed to the internal register/counter specified by the opcode.

Accumulator

The accumulator is a general purpose 8-bit computational register used for arithmetic and Boolean functions. It can not be used for indexing.

Index Registers (X, Y, Z)

There are three 8-bit index registers which may be incremented, decremented, compared or used to provide an index value to generate an effective address. The newly added Z register is cleared upon RESET allowing code compatibility with the 65C02.

When executing an instruction which specifies indexed addressing, the CPU fetches the OP code and the base address, and modifies the address by adding the index register value to it prior to performing the desired operation. Pre- or post-indexing of indirect addresses is possible (see addressing modes).

Base Page Register (B)

The Base Page register contains the value of the high order address byte used in the base page (formerly 'zero page') addressing modes. This register is programmable, via the TAB instruction, allowing any page in memory to function as the base page. On reset, the B register is cleared, initially providing a true "zero page".

Processor Status Register (P)

The 8-bit status register contains the values of the 8 status flags. Some flags are controlled by the program, while others are controlled by both the program and the ALU. One additional flag is present which enables extended (16-bit) stack pointer operation. The flags can be tested by a number of conditional branch instructions.

Address Counters (ABL and ABH) and Address Registers (ADL and ADH)

These registers are used to provide the 16-bits of addressing information for memory and I/O exchanges. A unique design feature allows ADL and ADH to store indirect address vectors while ABL and ABH function as counters, thus relieving the ALU from internal address fetches and increasing throughput.

Stack Pointer Counter (SPH and SPL)

The stack pointer is a 16 bit register that can operate in two modes. It can be programmed to be either an 8-bit page programmable pointer, or a full 16-bit pointer. The processor status E bit selects the 8-bit mode when set, and selects the 16-bit mode when reset.

Upon reset, the 65CE02 will be configured in the 8-bit page-programmable mode, with the stack page set to 01. This maintains compatibility with earlier 6502 products. The programmer can quickly change the default stack page by loading the Y register with the desired page and transferring its contents to the stack pointer high byte, using the TYS opcode. The 8-bit stack pointer can be set by loading the X register with the desired value, and transferring its contents to the stack pointer low byte, using the TXS opcode.

To select the 16-bit stack pointer mode, the user must execute a CLE (for CLear Extend disable) opcode. Setting the 16-bit pointer is done by loading the X and Y registers with the desired stack pointer low and high bytes, respectively, and then transferring their contents to the stack pointer using TXS and TYS. To return to 8-bit page mode, simply execute a SEE (SEt Extend disable) opcode.

CAUTION

When using interrupts, and BOTH stack pointer bytes are to be changed, do NOT put any code between the TXS and TYS opcodes. Taking this precaution will prevent any interrupts from occurring between the setting of the two stack pointer bytes, thus preventing the writing of stack data to an incorrect area.

Program Counter (PCL and PCH)

This 16-bit up counter determines the area of memory from which program information will be fetched. The user can modify the contents with jumps, branches, subroutine calls, or returns. It is set initially, and by interrupts, from vectors at memory addresses FFFA through FFFF (hex). See $\overline{IRQ}$, $\overline{NMI}$ and $\overline{RESET}$ below.

SIGNAL DESCRIPTIONS

Clock Signals

The 65CE02 requires an external, TTL-level $\varnothing_0$ clock. Two full level clocks ($\varnothing_1$ and $\varnothing_2$) are generated by the 65CE02. $\varnothing_2$ in phase with $\varnothing_0$, and $\varnothing_1$ 180 degrees out of phase with $\varnothing_0$. The input clock may be stopped in either phase to place the CPU into standby mode.

For non-critical timing applications, a simple RC or crystal network may be connected between $\varnothing_0$ (in) and $\varnothing_1$ (out).

Address Bus

A_0-A_{15} forms a 16-bit address bus for memory and I/O exchanges on the data bus. The output of each address line is TTL compatible, capable of driving two standard TTL loads and 55pF.

Data Bus

D_0-D_7 form an 8-bit bidirectional data bus for data exchanges to and from the 65CE02 and peripheral devices. The output buffers are capable of driving two standard TTL loads and 55pF. This bus is tristated during 'read' operations, during $\varnothing_2$ low time, and throughout any cycle where RDY is pulled 'low' prior to $\varnothing_2$ rising.

Interrupt Request ($\overline{IRQ}$)

This active-low input requests that an interrupt sequence begin within the microprocessor. If the interrupt mask flag (I) in the status register is zero, an interrupt sequence will begin with the first SYNC after a multiple-cycle opcode. The program counter and processor status register are then stored on the stack and the interrupt mask flag is set so that no further $\overline{IRQ}$'s may occur. The program counter low byte (PCL) is then loaded from address FFFE, and the high byte (PCH) from FFFF. Program execution will continue from the vector located at these addresses.

Non-Maskable Interrupt ($\overline{NMI}$)

The $\overline{NMI}$ input cannot be masked by the processor status register I flag and will cause an interrupt after receiving a high to low transition. This interrupt sequence will begin with the first SYNC after a multiple-cycle opcode. The two program counter bytes, PCH and PCL, and the processor status register P, are pushed onto the stack. The program counter bytes PCL and PCH are then loaded from memory addresses FFFA and FFFB, respectively. Program execution will then continue from the vector located at these addresses.

INTERRUPT NOTES

1. Since $\overline{NMI}$ is non-maskable, another $\overline{NMI}$ can occur before the first is finished. Care should be taken to avoid this.
2. The RDY signal must be 'high' to insure $\overline{IRQ}$ is recognized. The $\overline{NMI}$ input is edge activated and the 65CE02 will remember an $\overline{NMI}$ event, even if it is prevented from acting upon it by deasserting RDY.
3. A 3K ohm external pull-up resistor should be used for proper wire-OR operation.

Ready (RDY)

This input signal allows the user to single-cycle the microprocessor on all cycles including write cycles. A negative transition during or coincident with $\varnothing_1$ will halt the microprocessor with the output address lines reflecting the current address. This condition will remain throughout subsequent $\varnothing_2$ cycles in which the RDY signal is held 'low'. The RDY feature allows microprocessor interfacing with low speed memory as well as Direct Memory Access (DMA).

Read/Write (R/$\overline{W}$)

The R/$\overline{W}$ output signal remains in the 'high' state while the 65CE02 is reading data from memory or peripherals on the 8-bit data bus. When 'low', valid output data is available on the data bus.

Set Overflow ($\overline{SO}$)

A negative transition on this input sets the overflow bit (V) in the processor status register. The signal is sampled on the rising edge of $\varnothing_2$.

Reset ($\overline{RES}$)

The $\overline{RES}$ input instantly resets the 65CE02. $\overline{RES}$ should be held 'low' for at least 2 clock cycles after V_{DD} reaches operating voltage during power-up. Likewise, after the system has been operating, a low on this line will cease microprocessing activity. A positive transition on this pin begins an initialization sequence lasting six clock cycles.

The stack pointer is set to "byte" mode and the stack page is set to page 01. The B and Z registers are cleared and the processor status bits E and I will be set. The program counter is loaded with the restart vector from locations FFFC (low byte) and FFFD (high byte). This is the startup location for program control. During normal operation, $\overline{RES}$ should be held 'high'.

Synchronize (SYNC)

The SYNC output signal identifies those cycles in which the microprocessor is fetching OP CODE data. The SYNC line goes 'high' during $\varnothing_1$ of an OP CODE fetch and stays 'high' for the remainder of that cycle. If the RDY line is pulled 'low' during the $\varnothing_1$ pulse in which SYNC went 'high', the processor will stop and remain in its current state until the RDY line goes 'high'. In this manner, the SYNC signal can be used to control RDY for single instruction execution.

ADDRESSING MODES

The 65CE02 has 18 addressing modes (3 more than the 65C02, and 5 more than the 6502). The bracketed expression following the title of the mode is used later to identify the addressing mode in the Instruction Set Op Code Matrix and Summary tables.

Immediate [IMM]

In immediate addressing the second byte of the instruction contains the operand, with no further memory addressing required. (PHW -PusH Word-requires a third byte).

Absolute [ABS]

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 64K bytes of addressable memory.

Base Page [BP]

The base page mode allows for shorter code and execution time by using the contents of the base page register as the high order address byte. This differs from the previous Zero Page addressing mode in that the 'zero page' may now be relocated, thus making memory mapping more flexible and coding more efficient.

Accumulator [ACCUM]

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

Implied [IMPL]
In the implied addressing mode, the address containing the operand is implicitly stated in the operation code of the instruction.

Base Page Indexed [BP, X or BP, Y] (note 1)
This form of base page addressing calculates the effective low order address byte by adding the second instruction byte to the contents of the indexing register (X or Y). The high order byte is specified by the contents of the base page register. Additionally, due to the nature of "base page" addressing, no carry is added to the high order byte and page boundary crossing does not occur.

Absolute Indexed [ABS, X or ABS, Y]
This mode forms the effective address by adding the contents of X (or Y) to the address contained in the second and third bytes of the instruction. The index register contains the index or count value and the instruction specifies the base address.

Indexed Indirect [(BP, X)] (note 1)
In indexed indirect addressing, 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 in Base Page whose contents are the low order eight bits of the effective address. The next memory location in Base Page contains the high order eight bits of the effective address.

Indirect Indexed [(BP), Y or (BP), Z] (notes 1, 2)
In indirect indexed addressing, the second byte of the instruction points to a memory location in Base Page. The contents of this location and the Y (or Z) index register are added, 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 Base Page memory location, providing the high order eight bits of the effective address.

Stack Vector Indirect Indexed [(D, SP), Y] (note 3)
This new mode is similar to indirect indexed addressing. The Stack replaces the Base Page and the second instruction byte specifies the displacement from the current stack pointer location rather than the location within Base Page.
The contents of this displaced stack location are 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 (D+1) stack location, the result being the high order eight bits of the effective address.

Relative - Byte [REL]
Relative addressing is used only with branch instructions and establishes the destination for the branch. The second byte of the instruction is the "offset", whose range is -128 to 127 bytes, and is added to the address of the instruction following the branch.

Relative - Word [WREL] (note 3)
Similar to above but uses the second and third bytes as the 'offset'. The range of the offset is -32768 to +32767 bytes relative to the address of the third instruction byte (not the next instruction).

Absolute Indirect [(ABS)]
In the ABS mode the second and third bytes of the instruction respectively contain the lower and upper eight bits of a memory location. The content of this memory location is the low order byte of the effective address. The next memory location contains the high order byte of the effective address.

Base Page Relative [BP REL] (notes 1, 4)
This mode is used only with the bit-test branch instructions (BBS and BCC). The second byte of the instruction specifies the low order byte of Base Page memory to be tested. The third byte of the instruction is the offset, whose range is -128 to 127 bytes, referenced to the location of the next instruction.

Indexed Absolute Indirect [(ABS,X)] (note 4)
In this addressing mode the contents of the second and third instruction bytes are added to the X register. The sixteen bit result is a memory address containing the effective low order address byte. The next memory location contains the high order byte of the effective address.

---

note 1 - referenced to base page, not necessarily zero page.
2 - (BP), Z is a new addressing mode allowing indexing from the new Z register; it can also be used as the 65C02 uses (IND) by setting base page = zero page, and the contents of Z to $00 (default condition after RESET).
3 - new addressing mode not available on 65C02 or 6502.
4 - not available on 6502.

65CE02 OPCODE MATRIX

LSD

MSD	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	MSD
0	BRK IMPLIED 2, 7	ORA (BP, X) 2, 5	CLE IMPLIED 1, 2	SEE IMPLIED 1, 2	TSB BP 2, 4	ORA BP 2, 3	ASL BP 2, 4	RMB0 BP 2, 4	PHP IMPLIED 1, 3	ORA IMM 2, 2	ASL ACCUM 1, 1	TSY IMPLIED 1, 1	TSB ABS 3, 5	ORA ABS 3, 4	ASL ABS 3, 5	BBR0 BP 3, 4	0
1	BPL REL 2, 2	ORA (BP), Y 2, 5	ORA (BP), Z 2, 5	BPL W REL 3, 3	TRB BP 2, 4	ORA BP, X 2, 3	ASL BP, X 2, 4	RMB1 BP 2, 4	CLC IMPLIED 1, 1	ORA ABS, Y 3, 4	INC ACCUM 1, 1	INZ IMPLIED 1, 1	TRB ABS 3, 5	ORA ABS, X 3, 4	ASL ABS, X 3, 5	BBR1 BP 3, 4	1
2	JSR ABS 3, 5	AND (BP, X) 2, 5	JSR (ABS) 3, 7	JSR (ABS, X) 3, 7	BIT BP 2, 4	AND BP 2, 3	ROL BP 2, 4	RMB2 BP 2, 4	PLP IMPLIED 1, 3	AND IMM 2, 2	ROL ACCUM 1, 1	TYS IMPLIED 1, 1	BIT ABS 3, 5	AND ABS 3, 4	ROL ABS 3, 5	BBR2 BP 3, 4	2
3	BMI REL 2, 2	AND (BP), Y 2, 5	AND (BP), Z 2, 5	BMI W REL 3, 3	BIT BP, X 2, 4	AND BP, X 2, 3	ROL BP, X 2, 4	RMB3 BP 2, 4	SEC IMPLIED 1, 1	AND ABS, Y 3, 4	DEC ACCUM 1, 1	DEZ IMPLIED 1, 1	BIT ABS, X 3, 5	AND ABS, X 3, 4	ROL ABS, X 3, 5	BBR3 BP 3, 4	3
4	RTI IMPLIED 1, 5	EOR (BP, X) 2, 5	NEG ACCUM 1, 2	ASR ACCUM 1, 2	ASR BP 2, 4	EOR BP 2, 3	LSR BP 2, 4	RMB4 BP 2, 4	PHA IMPLIED 1, 3	EOR IMM 2, 2	LSR ACCUM 1, 1	TAZ IMPLIED 1, 1	JMP ABS 3, 3	EOR ABS 3, 4	LSR ABS 3, 5	BBR4 BP 3, 4	4
5	BVC REL 2, 2	EOR (BP), Y 2, 5	EOR (BP), Z 2, 5	BVC W REL 3, 3	ASR BP, X 2, 4	EOR BP, X 2, 3	LSR BP, X 2, 4	RMB5 BP 2, 4	CLI IMPLIED 1, 1	EOR ABS, Y 3, 4	PHY IMPLIED 1, 3	TAB IMPLIED 1, 1	AUG IMPLIED 4, 4	EOR ABS, X 3, 4	LSR ABS, X 3, 5	BBR5 BP 3, 4	5
6	RTS IMPLIED 1, 4	ADC (BP, X) 2, 5	RTN IMPLIED 2, 7	BSR W REL. 3, 5	STZ BP 2, 3	ADC BP 2, 3	ROR BP 2, 4	RMB6 BP 2, 4	PLA IMPLIED 1, 3	ADC IMM 2, 2	ROR ACCUM 1, 1	TZA IMPLIED 1, 1	JMP (ABS) 3, 5	ADC ABS 3, 4	ROR ABS 3, 5	BBR6 BP 3, 4	6
7	BVS REL 2, 2	ADC (BP), Y 2, 5	ADC (BP), Z 2, 5	BVS W REL 3, 3	STZ BP, X 2, 3	ADC BP, X 2, 3	ROR BP, X 2, 4	RMB7 BP 2, 4	SEI IMPLIED 1, 2	ADC ABS, Y 3, 4	PLY IMPLIED 1, 3	TBA IMPLIED 1, 1	JMP (ABS, X) 3, 5	ADC ABS, X 3, 4	ROR ABS, X 3, 5	BBR7 BP 3, 4	7
8	BRU REL 2, 2	STA (BP, X) 2, 5	STA (d,SP), Y 2, 6	BRU W REL 3, 3	STY BP 2, 3	STA BP 2, 3	STX BP 2, 3	SMB0 BP 2, 4	DEY IMPLIED 1, 1	BIT IMM 2, 2	TXA IMPLIED 1, 1	STY ABS, X 3, 4	STY ABS 3, 4	STA ABS 3, 4	STX ABS 3, 4	BBS0 BP 3, 4	8
9	BCC REL 2, 2	STA (BP), Y 2, 5	STA (BP), Z 2, 5	BCC W REL 3, 3	STY BP, X 2, 3	STA BP, X 2, 3	STX BP, Y 2, 3	SMB1 BP 2, 4	TYA IMPLIED 1, 1	STA ABS, Y 3, 4	TXS IMPLIED 1, 1	STX ABS, Y 3, 4	STZ ABS 3, 4	STA ABS, X 3, 4	STZ ABS, X 3, 4	BBS1 BP 3, 4	9
A	LDY IMM 2, 2	LDA (BP, X) 2, 5	LDX IMM 2, 2	LDZ IMM 2, 2	LDY BP 2, 3	LDA BP 2, 3	LDX BP 2, 3	SMB2 BP 2, 4	TAY IMPLIED 1, 1	LDA IMM 2, 2	TAX IMPLIED 1, 1	LDZ ABS 3, 4	LDY ABS 3, 4	LDA ABS 3, 4	LDX ABS 3, 4	BBS2 BP 3, 4	A
B	BLS REL 2, 2	LDA (BP), Y 2, 5	LDA (BP), Z 2, 5	BCS W REL 3, 3	LDY BP, X 2, 3	LDA BP, X 2, 3	LDX BP, Y 2, 3	SMB3 BP 2, 4	CLV IMPLIED 1, 1	LDA ABS, Y 3, 4	TSX IMPLIED 1, 1	LDZ ABS, X 3, 4	LDY ABS, X 3, 4	LDA ABS, X 3, 4	LDX ABS, Y 3, 4	BBS3 BP 3, 4	B
C	CPY IMM 2, 2	CMP (BP, X) 2, 5	CPZ IMM 2, 2	DEW BP 2, 6	CPY BP 2, 3	CMP BP 2, 3	DEC BP 2, 4	SMB4 BP 2, 4	INY IMPLIED 1, 1	CMP IMM 2, 2	DEX IMPLIED 1, 1	ASW ABS 3, 7	CPY ABS 3, 4	CMP ABS 3, 4	DEC ABS 3, 5	BBS4 BP 3, 4	C
D	BNE REL 2, 2	CMP (BP), Y 2, 5	CMP (BP), Z 2, 5	BNE W REL 3, 3	CPZ BP 2, 3	CMP BP, X 2, 3	DEC BP, X 2, 4	SMB5 BP 2, 4	CLD IMPLIED 1, 1	CMP ABS, Y 3, 4	PHX IMPLIED 1, 3	PHZ IMPLIED 1, 3	CPZ ABS 3, 4	CMP ABS, X 3, 4	DEC ABS, X 3, 5	BBS5 BP 3, 4	D
E	CPX IMM 2, 2	SBC (BP, X) 2, 5	LDA (d,SP), Y 2, 6	INW BP 2, 6	CPX BP 2, 3	SBC BP 2, 3	INC BP 2, 4	SMB6 BP 2, 4	INX IMPLIED 1, 1	SBC IMM 2, 2	NOP IMPLIED 1, 1	ROW ABS 2, 6	CPX ABS 3, 4	SBC ABS 3, 4	INC ABS 3, 5	BBS6 BP 3, 4	E
F	BEQ REL 2, 2	SBC (BP), Y 2, 5	SBC (BP), Z 2, 5	BEQ W REL 3, 3	PHW IMM W 3, 5	SBC BP, X 2, 3	INC BP, X 2, 4	SMB7 BP 2, 4	SED IMPLIED 1, 1	SBC ABS, Y 3, 4	PLX IMPLIED 1, 3	PLZ IMPLIED 1, 3	PHW ABS W 3, 7	SBC ABS, X 3, 4	INC ABS, X 3, 5	BBS7 BP 3, 4	F
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	

LSD

NEW OPCODE

0

0 | BRK IMPLIED 2, 7

- OPCODE
- ADDRESSING MODE
- Instruction Bytes, Machine Cycles

Note that the number of machine cycles for every instruction remains fixed, regardless of decimal mode and page boundaries.

DC CHARACTERISTICS

Absolute Maximum Ratings

Stresses above those listed may cause permanent damage to the circuit. Functional operation of the device at these or any conditions other than those indicated in the Operating Conditions of this specification is not implied. Exposure to the maximum ratings for extended periods may adversely affect device reliability.

Characteristic	Min	Max	Units
Ambient temperature under bias	-25	+125	deg C
Storage temperature	-65	+125	deg C
Applied supply voltage	-0.5	+7.0	volts
Applied output voltage	-0.5	+5.5	volts
Applied input voltage	-2.0	+7.0	volts

Operating Conditions

All electrical characteristics are specified over the entire range of the operating conditions unless otherwise noted. All voltages are referenced to V_{SS} = 0.0 V

Condition	Min	Max	Units
Supply voltage (V_{DD})	4.5	5.5	volts
Free air temperature	0	70	deg C

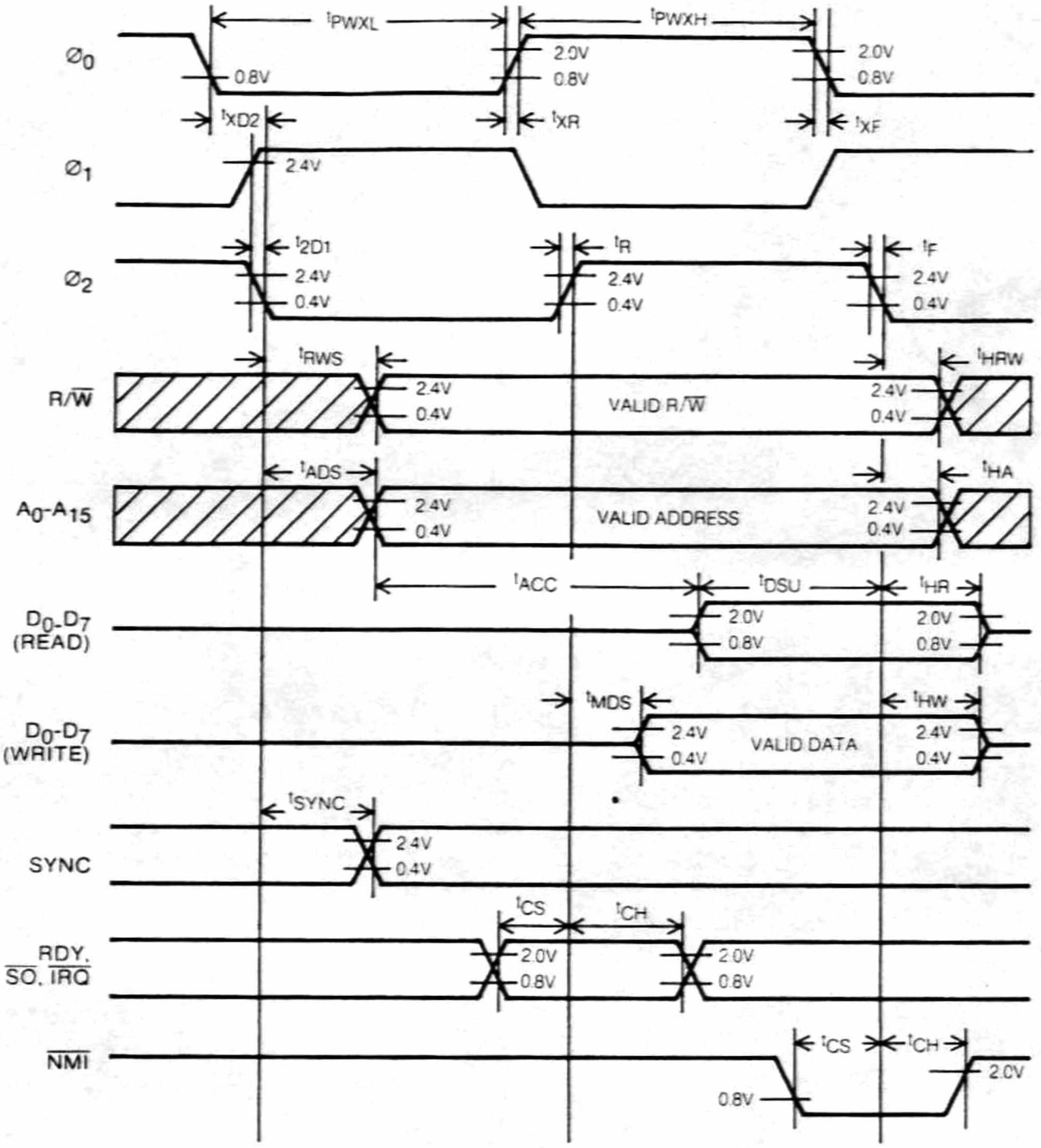
Interface Characteristics

Characteristic	Symbol	Min	Max	Units	Conditions
Input High Level	V_{IH}	2.0	V_{DD}	Volts	
Input Low Level	V_{IL}	- 0.5	0.8	Volts	
Input Leakage	I_{IN}	-10	10	µAmps	$0.0V<V_{IN}<V_{DD}$
Output High Level	V_{OH}	2.4	-	Volts	Iload = -200µA
Output Low Level	V_{OL}	-	0.4	Volts	Iload = 3.2mA
Supply Current	I_{DD}	-	10	µAmps	standby
		-	3.5	mA/MHz	active
Input Capacitance	C_{IN}	-	10	pF	V_{IN}=0V, f=4MHz
Output Capacitance	C_{OUT}	-	10	pF	

65CE02 TIMING SPECIFICATION

Parameter	Symbol	2 MHz		4 MHz		6 MHz		8 MHz		10 MHz	
		MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX
Cycle time	t_{CYC}	500		250		167		125		100	
Ext Clk Width Low	t_{PWXL}	230		115		75		55		45	
Ext Clk Width High	t_{PWXH}	230		115		75		55		45	
Ext low to $\emptyset_2$ low	t_{XD2}	5	40	5	30	5	30	5	30	5	25
$\emptyset_2$ low to $\emptyset_1$ high	t_{2D1}	- 20	20	- 15	15	- 10	10	- 10	10	- 10	10
$\emptyset_2$ Clk Rise, Fall	t_R, t_F		20		15		10		10		10
Ext Signal Rise, Fall	t_{XR}, t_{XF}		20		15		10		10		10
Read/Write Setup	t_{RWS}		50		40		35		30	5	25
Read/Write Hold	t_{HRW}	15		15		10		10		5	15
Address Setup	t_{ADS}		50		40		35		30	5	25
Address Hold	t_{HA}	15		15		10		10		5	15
Read Access	t_{ACC}	390		170		102		70		55	
Read Data Setup	t_{DSU}	60		40		30		25		20	
Read Data Hold	t_{HR}	10		10		10		10		10	
Write Data Delay	t_{MDS}		100		50		40		35	20	30
Write Data Hold	t_{HW}	15		15		10		10		5	15
SYNC Delay	t_{SYNC}		50		40		35		30	5	25
RDY, SO, IRQ, NMI Setup	t_{CS}	50		30		30		25		20	
RDY, SO, IRQ, NMI Hold	t_{CH}	30		25		25		20		20	

65CE02 TIMING DIAGRAM



65CE02 INSTRUCTION SET SUMMARY

INSTRUCTIONS		ADDRESSING MODES																							
Mnemonic	Operation	IMM			ABS			BP			ACCUM			IMPL			(BP, X)			(BP), Y			(BP), Z			(D, SP), Y		
		OP	N	#	OP	N	#	OP	N	#	OP	N	#	OP	N	#	OP	N	#	OP	N	#	OP	N	#	OP	N	#
ADC	A + M + C → A	69	2	2	6D	4	3	65	3	2							61	5	2	71	5	2	72	5	2			
AND	A .and. M → A	29	2	2	2D	4	3	25	3	2							21	5	2	31	5	2	32	5	2			
ASL	C ← /7...0/ ← 0				0E	5	3	06	4	2	0A	1	1															
ASR	M7 → /7...0/ → C							44	4	2	43	2	1															
ASW	C ← /15...0/ ← 0				CB	7	3																					
AUG	No Operation													5C	4	4												
BIT	A .and. M	89	2	2	2C	5	3	24	4	2																		
BBR	Branch on Bit Reset																											
BBS	Branch on Bit Set																											
BCC	Branch on C = 0																											
BCS	Branch on C = 1																											
BEQ	Branch on Z = 1																											
BMI	Branch on N = 1																											
BNE	Branch on Z = 0																											
BPL	Branch on N = 0																											
BRK	BReaK													00	7	2												
BRU (BRA)	BRanch Unconditional																											
BSR	Branch to SubRoutine																											
BVC	Branch on V = 0																											
BVS	Branch on V = 1																											
CLC	0 → C													18	1	1												
CLD	0 → D													D8	1	1												
CLE	0 → E													02	2	1												
CLI	0 → I													58	2	1												
CLV	0 → V													B8	1	1												
CMP	A - M	C9	2	2	CD	4	3	C5	3	2							C1	5	2	D1	5	2	D2	5	2			
CPX	X - M	E0	2	2	EC	4	3	E4	3	2																		
CPY	Y - M	C0	2	2	CC	4	3	C4	3	2																		
CPZ	Z - M	C2	2	2	DC	4	3	D4	3	2																		
DEC	M - 1 → M				CE	5	3	C6	4	2	3A	1	1															
DEW	Mw - 1 → Mw							C3	6	2																		
DEX	X - 1 → X													CA	1	1												
DEY	Y - 1 → Y													88	1	1												
DEZ	Z - 1 → Z													3B	1	1												
EOR	A .exclu or. M → A	49	2	2	4D	4	3	45	3	2							41	5	2	51	5	2	52	5	2			
INC	M + 1 → M				EE	5	3	E6	4	2	1A	1	1															
INW	Mw + 1 → Mw							E3	6	2																		
INX	X + 1 → X													E8	1	1												
INY	Y + 1 → Y													C8	1	1												
INZ	Z + 1 → Z													1B	1	1												
JMP	JuMP to new location				4C	3	3																					
JSR	Jump to SubRoutine				20	5	3																					
LDA	M → A	A9	2	2	AD	4	3	A5	3	2							A1	5	2	B1	5	2	B2	5	2	E2	6	2
LDX	M → X	A2	2	2	AE	4	3	A6	3	2																		
LDY	M → Y	A0	2	2	AC	4	3	A4	3	2																		
LDZ	M → Z	A3	2	2	AB	4	3																					
LSR	0 → /7...0/ → C				4E	5	3	46	4	2	4A	1	1															
NEG	1 - A → A										42	2	1															
NOP	No OPeration													EA	1	1												
ORA	A .or. M → A	09	2	2	0D	4	3	05	3	2							01	5	2	11	5	2	12	5	2			
PHA	A → Ms,S -1 → S													48	3	1												
PHP	P → Ms,S -1 → S													08	3	1												
PHW	Mw → Ms,S -2 → S	F4	5	3	FC	7	3																					
PHX	X → Ms,S -1 → S													DA	3	1												
PHY	Y → Ms,S -1 → S													5A	3	1												
PHZ	Z → Ms,S -1 → S													DB	3	1												
PLA	S +1 → S,Ms → A													68	3	1												
PLP	S +1 → S,Ms → P													28	3	1												
PLX	S +1 → S,Ms → X													FA	3	1												
PLY	S +1 → S,Ms → Y													7A	3	1												
PLZ	S +1 → S,Ms → Z													FB	3	1												
RMB	0 → Mb							(a)	4	2																		
ROL	C ← /7...0/ ← C				2E	5	3	26	4	2	2A	1	1															
ROR	C → /7...0/ → C				6E	5	3	66	4	2	6A	1	1															
ROW	C ← /15...0/ ← C				EB	6	2																					
RTI	ReTurn from Interrupt													40	5	1												
RTN	ReTurn from KerNal													62	7	2												
RTS	ReTurn from Subr													60	4	1												
SBC	A - M - 1 + C → A	E9	2	2	ED	3	2	E5	3	2							E1	5	2	F1	5	2	F2	5	2			
SEC	1 → C													38	1	1												
SED	1 → D													F8	1	1												
SEE	1 → E													03	2	1												
SEI	1 → I													78	2	1												
SMB	1 → Mb							(b)	4	2																		
STA	A → M				8D	4	3	85	3	2							81	5	2	91	5	2	92	5	2	82	6	2
STX	X → M				8E	4	3	86	3	2																		
STY	Y → M				8C	4	3	84	3	2																		
STZ	Z → M				9C	4	3	64	3	2																		
TAB	A → B													5B	1	1												
TAX	A → X													AA	1	1												
TAY	A → Y													A8	1	1												
TAZ	A → Z													4B	1	1												
TBA	B → A													7B	1	1												
TRB	A and M → M				1C	5	3	14	4	2																		
TSB	A or M → M				0C	5	3	04	4	2																		
TSX	S → X													BA	1	1												
TSY	S → Y													0B	1	1												
TXA	X → A													8A	1	1												
TXS	X → S													9A	1	1												
TYA	Y → A													98	1	1												
TYS	Y → S													2B	1	1												
TZA	Z → A													6B	1	1												

INST.	ADDRESSING MODES																					CONDITION CODES							
Mnemonic	BP, X			BP, Y			ABS, X			ABS, Y			REL (BYTE)			WREL (WORD)			(ABS)			BP REL			(ABS, X)			FLAGS							
	OP	N	#	OP	N	#	OP	N	#	OP	N	#	OP	N	#	OP	N	#	OP	N	#	OP	N	#	OP	N	#	N	V	E	B	D	I	Z	C
ADC	75	3	2				7D	4	3	79	4	3																N	V	-	-	-	-	Z	C
AND	35	3	2				3D	4	3	39	4	3																N	-	-	-	-	-	Z	
ASL	16	4	2				1E	5	3																			N	-	-	-	-	-	Z	C
ASR	54	4	2																									N	-	-	-	-	-	Z	C
ASW																												N	-	-	-	-	-	Z	C
AUG																												-	-	-	-	-	-	-	-
BIT	34	4	2				3C	5	3																			7	6	-	-	-	-	Z	-
BBR																						(c)	4	3				-	-	-	-	-	-	-	-
BBS																						(d)	4	3				-	-	-	-	-	-	-	-
BCC													90	2	2	93	3	3										-	-	-	-	-	-	-	-
BCS													B0	2	2	B3	3	3										-	-	-	-	-	-	-	-
BEQ													F0	2	2	F3	3	3										-	-	-	-	-	-	-	-
BMI													30	2	2	33	3	3										-	-	-	-	-	-	-	-
BNE													D0	2	2	D3	3	3										-	-	-	-	-	-	-	-
BPL													10	2	2	13	3	3										-	-	-	-	-	-	-	-
BRK																												-	-	-	-	-	-	-	-
BRU (BRA)													80	2	2	83	3	3										-	-	-	-	-	-	-	-
BSR																63	5	3										-	-	-	-	-	-	-	-
BVC													50	2	2	53	3	3										-	-	-	-	-	-	-	-
BVS													70	2	2	73	3	3										-	-	-	-	-	-	-	-
CLC																												-	-	-	-	-	-	-	R
CLD																												-	-	-	-	R	-	-	-
CLE																												-	-	R	-	-	-	-	-
CLI																												-	-	-	-	-	R	-	-
CLV																												-	R	-	-	-	-	-	-
CMP	D5	3	2				DD	4	3	D9	4	3																N	-	-	-	-	-	Z	C
CPX																												N	-	-	-	-	-	Z	C
CPY																												N	-	-	-	-	-	Z	C
CPZ																												N	-	-	-	-	-	Z	C
DEC	D6	4	2				DE	5	3																			N	-	-	-	-	-	Z	-
DEW																												N	-	-	-	-	-	Z	-
DEX																												N	-	-	-	-	-	Z	-
DEY																												N	-	-	-	-	-	Z	-
DEZ																												N	-	-	-	-	-	Z	-
EOR	55	3	2				5D	4	3	59	4	3																N	-	-	-	-	-	Z	-
INC	F6	4	2				FE	5	3																			N	-	-	-	-	-	Z	-
INW																												N	-	-	-	-	-	Z	-
INX																												N	-	-	-	-	-	Z	-
INY																												N	-	-	-	-	-	Z	-
INZ																												N	-	-	-	-	-	Z	-
JMP																			6C	5	3				7C	5	3	-	-	-	-	-	-	-	-
JSR																			22	7	3				23	7	3	-	-	-	-	-	-	-	-
LDA	B5	3	2				BD	4	3	B9	4	3																7	-	-	-	-	-	Z	-
LDX				B6	3	4				BE	4	3																7	-	-	-	-	-	Z	-
LDY	B4	3	2				BC	4	3																			7	-	-	-	-	-	Z	-
LDZ							BB	4	3																			7	-	-	-	-	-	Z	-
LSR	56	4	2				5E	5	3																			R	-	-	-	-	-	Z	C
NEG																												N	-	-	-	-	-	Z	-
NOP																												N	-	-	-	-	-	Z	-
ORA	15	3	2				1D	4	3	19	4	3																N	-	-	-	-	-	Z	-
PHA																												-	-	-	-	-	-	-	-
PHP																												-	-	-	-	-	-	-	-
PHW																												-	-	-	-	-	-	-	-
PHX																												-	-	-	-	-	-	-	-
PHY																												-	-	-	-	-	-	-	-
PHZ																												-	-	-	-	-	-	-	-
PLA																												N	-	-	-	-	-	Z	-
PLP																												7	6	-	-	3	2	1	0
PLX																												N	-	-	-	-	-	Z	-
PLY																												N	-	-	-	-	-	Z	-
PLZ																												N	-	-	-	-	-	Z	-
RMB																												-	-	-	-	-	-	-	-
ROL	36	4	2				3E	5	3																			N	-	-	-	-	-	Z	C
ROR	76	4	2				7E	5	3																			N	-	-	-	-	-	Z	C
ROW																												N	-	-	-	-	-	Z	C
RTI																												7	6	-	-	3	2	1	0
RTN																												-	-	-	-	-	-	-	-
RTS																												-	-	-	-	-	-	-	-
SBC	F5	3	2				FD	4	3	F9	4	3																N	V	-	-	-	-	Z	C
SEC																												-	-	-	-	-	-	-	S
SED																												-	-	-	-	S	-	-	-
SEE																												-	-	S	-	-	-	-	-
SEI																												-	-	-	-	-	S	-	-
SMB																												-	-	-	-	-	-	-	-
STA	95	3	2				9D	4	3	99	4	3																-	-	-	-	-	-	-	-
STX				96	3	2				9B	4	3																-	-	-	-	-	-	-	-
STY	94	3	2				8B	4	3																			-	-	-	-	-	-	-	-
STZ	74	3	2				9E	4	3																			-	-	-	-	-	-	-	-
TAB																												-	-	-	-	-	-	-	-
TAX																												N	-	-	-	-	-	Z	-
TAY																												N	-	-	-	-	-	Z	-
TAZ																												N	-	-	-	-	-	Z	-
TBA																												N	-	-	-	-	-	Z	-
TRB																												-	-	-	-	-	-	Z	-
TSB																												-	-	-	-	-	-	Z	-
TSX																												N	-	-	-	-	-	Z	-
TSY																												N	-	-	-	-	-	Z	-
TXA																												N	-	-	-	-	-	Z	-
TXS																												-	-	-	-	-	-	-	-
TYA																												N	-	-	-	-	-	Z	-
TYS																												-	-	-	-	-	-	-	-
TZA																												N	-	-	-	-	-	Z	-

LEGEND

A = Accumulator
X = X Index Register
Y = Y Index Register
Z = Z Index Register
P = Processor Status Register
B = Base Page Register
S = Stack Pointer
OP = Hexidecimal Opcode
N = Number of Cycles
# = Number of Bytes

M = Memory per Effective Address
Ms = Memory per Stack Pointer
Mw = Memory per Effective Address (Word)
Mb = Memory Bit
M7 = Memory Bit 7

	OPCODE
(a) = Reset M(0-7)	07-77
(b) = Set M(0-7)	87-F7
(c) = Branch on M(0-7) = 0	0F-7F
(d) = Branch on M(0-7) = 1	8F-FF

ORDERING INFORMATION

The CSG 65CE02 is available in a plastic 40-pin dual-in-line package with operating frequencies of 2, 4, 6, 8 or 10 MHz. These versions are coded into the part number as follows:

65CE02 -

- 2 - 2 MHz
- 4 - 4 MHz
- 6 - 6 MHz
- 8 - 8 MHz
- 10 - 10 MHz

For more information contact:

Commodore Semiconductor Group
950 Rittenhouse Road
Norristown, PA 19403
Tel.: 215/666-7950
TWX 510-660-4168
FAX 215/647-0791