W65C832 CPU Datasheet v2.0

Date: 9-6-10

Updated and .PDFed by <u>ReactiveMicro.com</u> to resolve several issues with the original datasheet.

Thanks to Tom and Constantine for sharing and helping the Community! Without them this datasheet and all related projects that come from it would not be possible. Thanks guys!

Original location and files: http://apple2.org.za/gswv/a2zine/Docs/CPU_65832

The 65C832 CPU Manual in GIF Graphic format Copyright © 1999 by Constantine Garland

This folder contains a series of 8 bit 256 color GIF graphic files that are the copyrighted material of Constantine Garland.

TOC1.GIF through TOC5.GIF consist of the manual information, specification and data sheet signed by Woz and the table of contents. M1.GIF through M51.GIF contains the entire context of the 65C832 CPU Preliminary Manual, information, specification and diagrams and data sheets.

They may be used without any restrictions for historical information, educational, research and graphic art needs by anybody, provided no fees or money are charged for such use.

Apple IIgs Forever! Constantine

Subject: Re: 65832 CPU??? Apple //f???

Date: 11 Apr 2002 04:56:36 GMT From: cturley2@aol.com (Cturley2) Newsgroups: comp.sys.apple2

Rubywand wrote asking:

<< That's a pretty rare manual. How did you obtain it?>>

;-) Legally -- by snail mail from a guy in Oklahoma that offered to send it to me for free. He told me he was a student in high school years ago and decided he wanted info on the then rumored NEW 65832 CPU for the IIgs.

He contacted WDC with his request convinced them he was a big time chip manufacture interested in buying it for commercial production and they sent him one.

I got wind of him from an offer he posted to this NG back in 1995 -- asking if anybody was interested in it for free. I emailed him first on it and he mailed it on to me promptly.

I showed it to Woz at a rock concert back in 1996 -- he was astounded when reviewing it and signed the cover page for me. That's why it has that Woz sig. on it. As such -- yes, it is pretty rare -- and -- with mine and the autograph from Woz on it's cover page --- it's a priceless and VERY rare one-of-a-kind.

WDC told me on several occasions in our many telcons: "We have no such manual our self now -- only made a dozen or so and mailed all of them out to clients many years ago." I've never found anybody else that has one either.

Mine may well be the only one left in existence now (???). That's why I digitized it *cover to cover* and put it online to share the info with the world.

Cheers, Tom

W65C832
INFORMATION, SPECIFICATION AND DATA SHEET PRELIMINARY

	ML	I R Q	A B O R T -	R D Y	V P	v s s	R E S	VDA	M / X	P H I 2	B E	
NMI- VPA VDO AO	6 7 8 9	5	4	3	2	1	44	43	42	41	40 39 38 37 36	E8/E16 R/W- VDD DO/A16
A1 VSS A2 A3 A4 A5 A6	11 12 13 14 15 16 17 18	19	20	21		23	24	25	26	27	35 34 33 32 31 30 29 28	D1/A17 D2/A18 D3/A19 D4/A20 D5/A21 D6/A22 D7/A23
	A 7	A 8	A 9	A 1 0	A 1	V S S	V S S	A 1 2	A 1 3	A 1 4	A 1 5	j

163/

TABLE OF CONTENTS

INTRODU	CTION																				
SECTION	1: W65C832 FUNCTION	DES	SCF	II	PTI	101	1														- 7
1.1	Instruction Register	and	1 1	/ec	200	ie.	•	•	*				•	•		•	*	•	*	•	
1.2	Timing Control Unit Arithmetic and Logic	77-				*	•		•		•	T.	•	*					*	•	
1.3	Arithmetic and Logic	Umi	LC	•			*	•		•			•		•			•	*	100	
III I I I I I I I I I I I I I I I I I	Internal Registers.																				
1.5	Accumulators																				
1.6	Data Bank Register.	• •	•	•			•				•		•		•	•		٠		•	
1.7	Direct		٠	•		٠		٠		٠						٠					
	Index																				
	Processor Status																				
	Program Bank Registe:																				
	Program Counter																				
1.12	Stack Pointer	•	٠	٠	*	٠	•	٠	•	•	٠	٠	•	٠	٠	•	٠	٠		٠	. 4
SECTION	2: PIN FUNCTION DESC	CRIE	TI	(0)	1																10
2.1	Abort																				11
2.1	Address Bus								•						5					•	11
	Bus Enable																				
	Data/Address Bus																				
	Emulation Status																				
2.0	Interrupt Request .	• •	*	•				٠	•	•	•		•	•		•	9.		1	10	. 12
2.1	Memory Lock		٠	•	*		•	•	•				•				•	•		٠	.14
2.8	Memory/Index Select :	Stat	us	3.	٠								•				1			٠	.12
2.9	Non-Maskable Interrup	pt.					+						•		•	٠			•		.12
2.10	Phase 2 In		•		•							٠	•					٠			.12
2.11	Read/Write		+							*				+					4		.13
2.12	Ready		+		٠		+			٠			•			•		÷.			.13
2.13	Reset								+		+				+						.13
2.14	Valid Data Address, V	Vali	d	Pr	000	gra	m.	Ac	idi	res	39			٠							.14
2.15	VDD and VSS																				.14
2.16	Vector Pull	•	٠	•	٠	•	•	٠	•	٠	٠			•		٠		٠	٠	٠	.14
SECTION	3: ADDRESSING MODES					1															15
	Reset and Interrupt \																				
	Stack																				
	Direct																				
3.4	Program Address Space							1		5.											.15
3.5	Data Address Space																			•	.15

SECTION	4: TIMING, AC AND DC CHARACTERISTICS
4.1 4.2	Absolute Maximum Ratings
	AC Characteristics, 5V
SECTION	5: ORDERING INFORMATION
SECTION	6: APPLICATION INFORMATION
SECTION	7: ASSEMBLER SYNTAX STANDARDS
7.1	Directives
7.2	Comments
7.3	The Source Line
SECTION	8: CAVEATS
8.1	Stack Addressing
8.2	Direct Addressing
	Absolute Indexed Addressing
8.4	ABORT
	VDA and VPA
8.6	Apple II, IIc, IIc and II+ Disk Systems
	DB/BA Operation
8.8	M/X Output
8.9	M/X Output
8.10	Indirect Jumps
8.11	Switching Modes
8.12	Hardware Interrupts, BRK, and COP Instructions
	Binary Mode
8.14	WAI Instruction
8.15	STP Instruction
	COP Signatures
	WDM Opcode Use
8.18	RDY Pulled During Write
8.19	MVN and MVP
8.20	Interrupt Priorities
8.21	Transfers from differing sized registers
8.22	Stack Transfers
	REP/SEP

TABLE OF CONTENTS

FIGURES

SECTION	1: FUNCTION DESCRIPTION							2
1-1	Internal Architecture Block Diagram		٠					5
1-2	W65C832 Native Mode Programming Model				٠			6
	W65C816 16-bit Emulation Programming Model.							
	W65C02 8-bit Emulation Programming Model							
1-5	W65C832 Status Register Coding					+		9
SECTION	2: PIN FUNCTION DESCRIPTION						1	0
2-1	W65C832 44 PLCC Pinout						.1	0
SECTION	4: TIMING, AC AND DC CHARACTERISTICS						2	4
4-1	Timing Diagram						. 2	8

TABLE OF CONTENTS

TABLES

SECTION	1: FUNCTION DESCRIPTION 2
1-1	W65C832 Emulation and Register Width Control 9
SECTION	2: PIN FUNCTION DESCRIPTION 10
2-1	Pin Function Table
SECTION	3: ADDRESSING MODES 15
	Address Mode Formats
SECTION	4: TIMING, AC AND DC CHARACTERISTICS 24
4-2 4-3A 4-3B	Absolute Maximum Ratings
SECTION	6: APPLICATION INFORMATION 30
6-2 6-3	Instruction Set
SECTION	7: ASSEMBLER SYNTAX STANDARDS 43
7-3-1 7-3-2	Alternate Mnemonics
SECTION	8: CAVEATS 46
8-1	Compatibility Issues

INTRODUCTION

The WDC W65C832 is a CMOS 32-bit microprocessor featuring total software compatibility with their 8-bit NMOS and 8-bit and 16-bit CMOS 6500-series predecessors. The W65C832 is pin-to-pin compatible with 16-bit devices currently available. These devices offer the many advantages of CMOS technology, including increased noise immunity, higher reliability, and greatly reduced power requirements. A software switch determines whether the processor is in the 8-bit or 16-bit "emulation" mode, or in the native mode, thus allowing existing systems to use the expanded features.

As shown in the processor programming model, the Accumulator, ALU, X and Y Index registers have been extended to 32 bits. A 16-bit Program Counter, Stack Pointer and Direct Page register augments the Direct Page addressing mode (formerly Zero Page addressing). Separate Program Bank and Data Bank registers allow 24-bit memory addressing with segmented or linear addressing for program space and 32-bit 4GByte data space for ASIC use although only 24 bits of address are available in the standard pin-out.

Four signals provide the system designer with many options. The ABORT input can interrupt the currently executing instruction without modifying internal register, thus allowing virtual memory system design. Valid Data Address (VDA) and Valid Program Address (VPA) outputs facilitate dual cache memory by indicating whether a data segment or program segment is accessed. Modifying a vector is made easy by monitoring the Vector Pull (VP) output.

KEY FEATURES OF THE W65C832

- * Advanced CMOS design for low power * Separate program and data bank power consumption and increased noise immunity
- Single 1.2-5.25V power supply, as specified
- * Emulation mode allows complete hardware and software compatibility with W65C816 designs
- * 24-bit address bus allows access to 16 MBytes of memory space
- * Full 32-bit ALU, Accumulator, and Index Registers
- * Valid Data Address (VDA) and Valid Program Address (VPA) output allows dual cache and cycle steal DMA implementation
- * Vector Pull (VP) output indicates when interrupt vectors are being addressed. May be used to implement vectored interrupt
- * Abort (ABORT) input and associated vector supports virtual memory system design

- registers allow program segmentation or full 16-MByte linear addressing
- New Direct Register and stack relative addressing provides capability for re-entrant, re-cursive and re-locatable programming
- 24 addressing modes-13 original 6502 modes, plus 11 new addressing modes with 91 instructions using 255 opcodes
- * Wait-for-Interrupt (WAI) and Stop-the Clock (STP) instructions further reduce power consumption, decrease interrupt latency and allows synchronization with external events
- * Co-Processor (COP) instruction with associated vector supports co-processor configurations, i.e., floating point processors
- Block move ability

SECTION 1

W65C832 FUNCTION DESCRIPTION

The W65C832 provides the design engineer with upward mobility and software compatibility in applications where a 32-bit system configuration is desired. The W65C832's 32-bit hardware configuration, coupled with current software allows a wide selection of system applications. In the Emulation mode, the W65C832 offers many advantages, including full software compatibility with 6502, W65C02 or W65C816 coding. In addition, the W65C832's powerful instruction set and addressing modes make it an excellent choice for new 32-bit designs.

Internal organization of the W65C832 can be divided into two parts: 1) The Register Section and 2) The Control Section. Instructions (or opcodes) obtained from program memory are executed by implementing a series of data transfers within the Register Section. Signals that cause data transfers to be executed are generated within the Control Section. The W65C832 has a 32-bit internal architecture with an 8-bit external data bus.

1.1 Instruction Register and Decode

An opcode enters the processor on the Data Bus, and is latched into the Instruction Register during the instruction fetch cycle. This instruction is then decoded, along with timing and interrupt signals, to generate the various Instruction Register control signals.

1.2 Timing Control Unit (TCU)

The Timing Control Unit keeps track of each instruction cycle as it is executed. The TCU is set to zero each time an instruction fetch is executed, and is advanced at the beginning of each cycle for as many cycles as is required to complete the instruction. Each data transfer between registers depends upon decoding the contents of both the Instruction Register and the Timing Control Unit.

1.3 Arithmetic and Logic Unit (ALU)

All arithmetic and logic operations take place within the 32-bit ALU. In addition to data operations, the ALU also calculates the effective address for relative and indexed addressing modes. The result of a data operation is stored in either memory or an internal register. Carry, Negative, Overflow and Zero flags may be updated following the ALU data operation.

selection descharges the Employee Carry and Smaletion Stin (ACS) instruction, the ISS instruction descharges the Employee (Ed and Sill) mode swiech files with the Overfiles the And Carry Fiage. Table I toulation and Register Hidth Coutrol, Liberties the features of the Smaletian modes. The M and S Silver are elected aqual to the Table T-bit Employee wide. When an interior during the Maulation and

1.4 Internal Registers (Refer to Programming Model)

2

1.5 Accumulator

The Accumulator is a general purpose register which stores one of the operands, or the result of most arithmetic and logical operations. In the Native mode the Accumulator can be 8-, 16- or 32-bits wide.

1.6 Data Bank Register (DBR)

During modes of operation, the 8-bit Data Bank Register holds the default bank address for memory transfers. The 24-bit address is composed of the 16-bit instruction effective address and the 8-bit Data Bank address. The register value is multiplexed with the data value and is present on the Data/Address lines during the first half of a data transfer memory cycle for the W65C832. The Data Bank Register is initialized to zero during Reset.

1.7 Direct (D)

The 16-bit Direct Register provides an address offset for all instructions using direct addressing. The effective bank zero address is formed by adding the 8-bit instruction operand address to the Direct Register. The Direct Register is initialized to zero during Reset.

1.8 Index (X and Y)

There are two Index Registers (X and Y) which may be used as general purpose registers or to provide an index value for calculation of the effective address. When executing an instruction with indexed addressing, the microprocessor fetches the opcode and the base address, and then modifies the address by adding the Index Register contents to the address prior to performing the desired operation. Pre-indexing or post-indexing of indirect addresses may be selected. In the Native mode, both Index Registers are 32 bits wide (providing the Index Select Bit (X) equals zero). If the Index Select Bit (X) equals one, both registers will be 8 bits wide, and the high bytes if forced to zero.

1.9 Processor Status (P)

The 8-bit Processor Status Register contains status flags and mode select bits. The Carry (C), Negative (N), Overflow (V), and Zero (Z) status flags serve to report the status of most ALU operations. These status flags are tested by use of Conditional Branch instructions. The Decimal (D), IRQ Disable (I), Memory/Accumulator (M), and Index (X) bits are used as mode select flags. These flags are set by the program to change microprocessor operations.

The Emulation (E8 and E16) select and the Break (B) flags are accessible only through the Processor Status Register. The Emulation (E8) mode select flag is selected by the Exchange Carry and Emulation Bits (XCE) instruction. The XFE instruction exchanges the Emulation (E8 and E16) mode select flags with the Overflow and Carry Flags. Table 1, Emulation and Register Width Control, illustrates the features of the Native and Emulation modes. The M and X flags are always equal to one in the 8-bit Emulation mode. When an interrupt occurs during the Emulation mode, the Break flag is written to stack memory as bit 4 of the Processor Status Register.

MARCH 1990 3361 ADMAR

1.10 Program Bank Register (PBR)

The 8-bit Program Bank Register holds the bank address for all instruction fetches. The 24-bit address consists of the 16-bit instruction effective address and the 8-bit Program Bank address. The register value is multiplexed with the data value and presented on the Data/Address lines during the first half of a program memory read cycle. The Program Bank Register is initialized to zero during Reset. The PHK instruction pushes the PBR register onto the Stack.

1.11 Program Counter (PC)

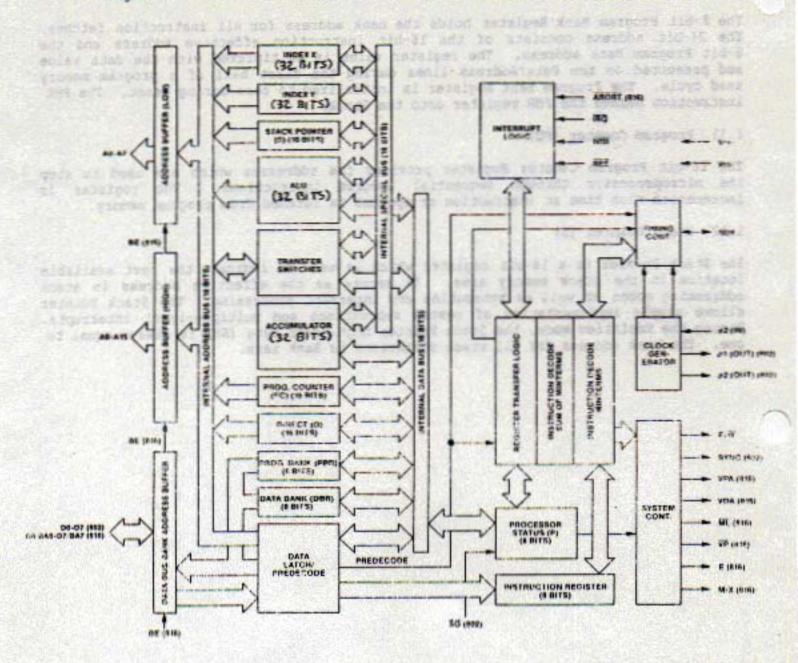
The 16-bit Program Counter Register provides the addresses which are used to step the microprocessor through sequential program instructions. The register is incremented each time an instruction or operand is fetched from program memory.

1.12 Stack Pointer (S)

The Stack Pointer is a 16-bit register which is used to indicate the next available location in the stack memory area. It serves as the effective address in stack addressing modes as well as subroutine and interrupt processing. The Stack Pointer allows simple implementation of nested subroutines and multiple-level interrupts. During the Emulation mode, the Stack Pointer high-order byte (SH) is always equal to one. The bank address for all stack operations is Bank zero.

MARCH 1990 4 766 10469

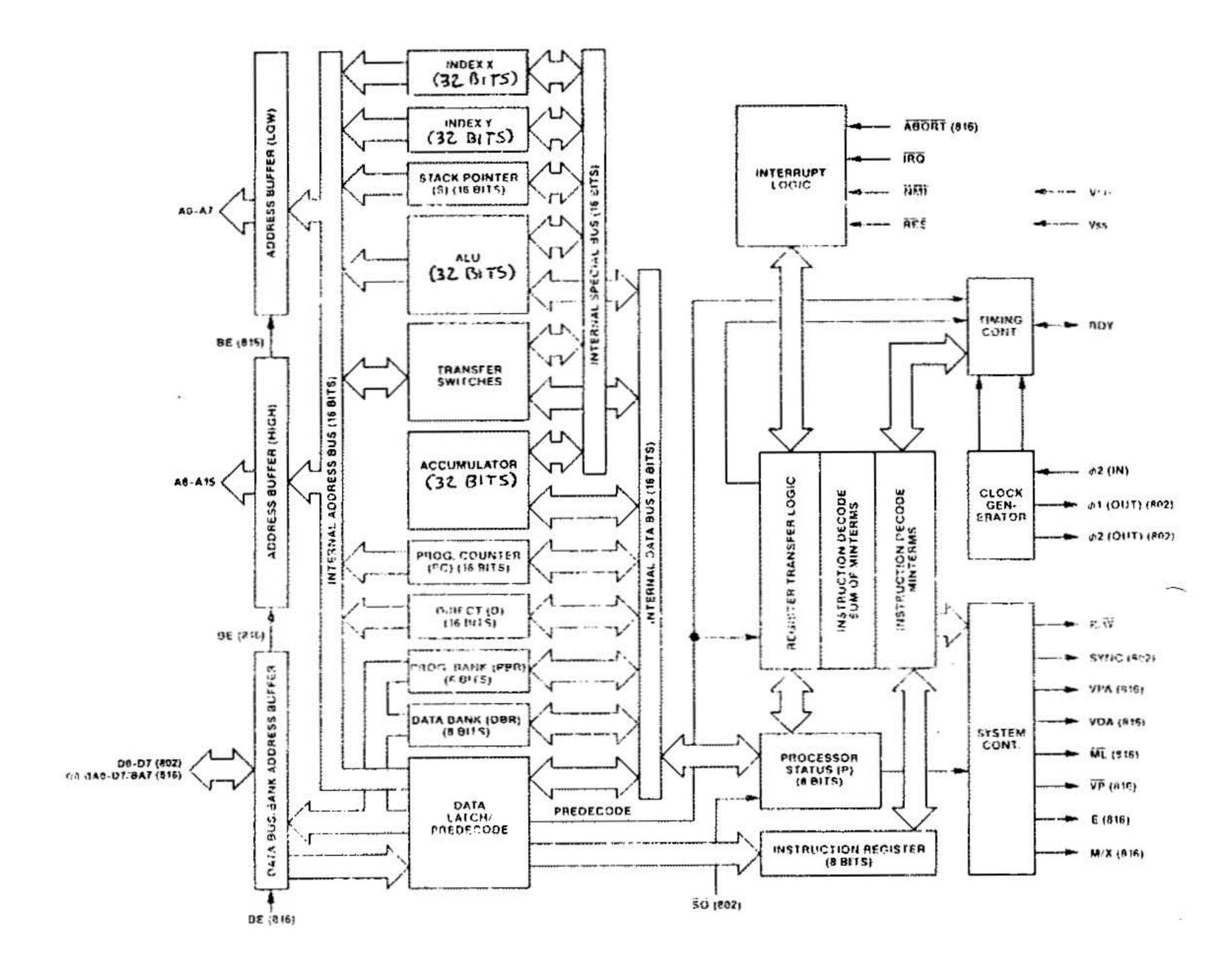
Figure 1-1 W65C832 Internal Architecture Simplified Block Diagram



MARCH 1990

5

Figure 1-1 W65C832 Internal Architecture Simplified Block Diagram



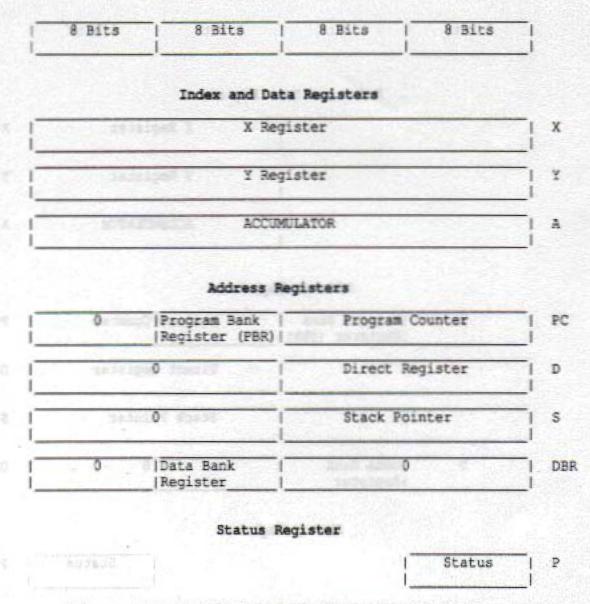


Figure 1-2 W65C832 Native Mode Programming Model

MARCH 1990

6

	Index and Data	Registers
	16382165	X Register
	A SALI	Y Register
	AOTA-DE-	ACCUMULATOR
	Address Regi	sters
0	Program Bank Register (PBR)	Program Counter
3679		Direct Register
3079	Register (PBR)	L sentitos

Figure 1-3 W65C816 16-bit Emulation Programming Model

7085 HORSE MARCH 1990

8 Bits	8 Bits	8 Bits	8 Bits	
	Index and Data	Registers		
	nust + F y	NEW TEN	X Register	
	coss digest w. I	1 5020		
	soil = 1 soos int	1985	Y Register	
	speled distal es word = I void spec = I swill		ACCUMULATOR	
	Address Reg	isters		
0	Program Bank Register (PBR)	Program	Counter	
amed to	0 30 22 362 1036	Direct	Register	
	0	1	Stack Pointer	
0	Data Bank Register	0		
	Status Reg	ister		
			Status	
18 12 Tartisy				
igure 1-4	W65C02 8-bit Emul	lation Progra	mming Model	
N. 560 STE				
His Bella		TO THE PERSON		
Sions Fift	15 000			
siesi VIV				

MARCH 1990

BRELL BOATH

9

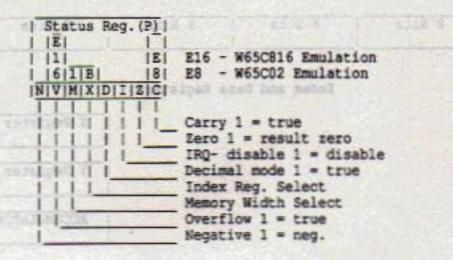


Figure 1-5 W65C832 Status Register Coding

Table 1-1 W65C832 Emulation and Register Width Control

				A and Memory Loads, Stores, Pushes, and Pulls	X,Y Loads, Stores, Pushes, Pulls, and Address	Generation	10 10 10 10 10 10 10 10 10 10 10 10 10 1
E16	E8	М	Х				
0	0	0	0	16	32	W65C832	Native
0	0	0	1	16	8	W65C832	Native
0	0	1	0	8	32	W65C832	Native
0 0 0 0	0	1	1	8	8	W65C832	Native
0	0	0	0	32	32	W65C832	Native
0	1	0	1	32	8	W65C832	Native
0	1	1	0	8	32	W65C832	Native
0	1	1	0 1 0	8	8	W65C832	Native
0 1 1	0	0		16	16	W65C816	Emulation
1	0	0	1	16	8	W65C816	Emulation
1	0	1	0	8	16	W65C816	Emulation
1	0	1	1	8	8	W65C816	Emulation
1	1	1	BRK	8	8	W65C02	Emulation

MARCH 1990

SECTION 2

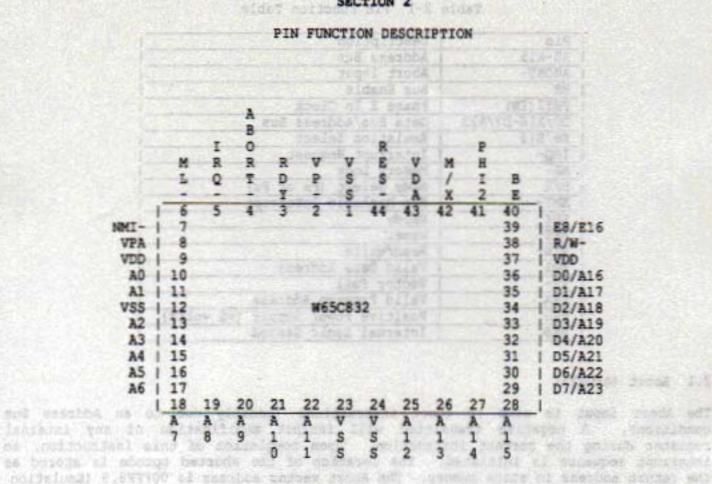


Figure 2-1 W65C832 44 Pin PLCC Pinout

Also fine fineble dopot etgins allows concerns concept of the Address and Data Surface, are as well as the bigs algest. With the English high, the Rife end Address butters are butter. The Data/Address butter are solding disting the fixet half of every cycle and the cooced bull of a write syrie. When HE is ion, these butters are disabled.

These explications succepted and when the first hideside with the detrivative of 17, one extress is present during the first half of a mostly cycle, and the dark value of read or extracted during the second half of the mesony cycle. Four mesony cycles dry required to cranates Albert spaces these lines may be set in the high incredence

the vices of the bars have the edites of the bars the high hour or the high involunce

MARCH 1990

10

HEADASH SE

Codd 2,68750 on (stor Law on about vall) on Line a polyno nie -7800a

Listing 1387 alder mil not ye are

ALCA TO-BULLDER BURN SERVICE F. C.

Torolly (BS) wides will but you are to

Table 2-1 Pin Function Table

Pin	Description					
A0-A15	Address Bus					
ABORT-	Abort Input					
BE	Bus Enable					
PHI2(IN)	Phase 2 In Clock					
DO/A16-D7/A23	Data Bus/Address Bus					
E8/E16	Emulation Select					
IRQ-	Interrupt Request					
ML-	Memory Lock					
M/X	Mode Select (Pm or Px)					
NMI-	Non-Maskable Interrupt					
RDY	Ready					
RES-	Reset					
R/W-	Read/Write					
VDA	Valid Data Address					
VP-	Vector Pull					
VPA	Valid Program Address					
VDD	Positive Power Supply (+5 volts)					
VSS	Internal Logic Ground					

2.1 Abort (ABORT-)

The Abort input is used to abort instructions (usually due to an Address Bus condition). A negative transition will inhibit modification of any internal register during the current instruction. Upon completion of this instruction, an interrupt sequence is initiated. The location of the aborted opcode is stored as the return address in stack memory. The Abort vector address is 00FFF8,9 (Emulation mode) or 00FFE8,9 (Native mode). Note that ABORT- is a pulse-sensitive signal; i.e., an abort will occur whenever there is a negative pulse (or level) on the ABORT- pin during a PHI2 clock.

2.2 Address Bus (A0-A15)

These sixteen output lines form the low 16 bits of the Address Bus for memory and I/O exchange on the Data Bus. The address lines may be set to the high impedance state by the Bus Enable (BE) signal.

2.3 Bus Enable (BE)

The Bus Enable input signal allows external control of the Address and Data Buffers, as well as the R/W- signal. With Bus Enable high, the R/W- and Address Buffers are active. The Data/Address Buffers are active during the first half of every cycle and the second half of a write cycle. When BE is low, these buffers are disabled. Bus Enable is an asynchronous signal.

2.4 Data/Address Bus (DO/A16-D7/A23)

These eight lines multiplex address bits A16-A23 with the data value D0-D7. The address is present during the first half of a memory cycle, and the data value is read or written during the second half of the memory cycle. Four memory cycles are required to transfer 32-bit values. These lines may be set to the high impedance state by the Bus Enable (BE) signal.

THE RESERVE AND ADDRESS OF THE PARTY OF THE

2.5 Emulation Status (E8/E16)

The Emulation Status output E8/E16 reflects the state of the Emulation E8 and E16 mode flags in the Processor Status (P) Register. This signal may be thought of as an opcode extension and used for memory and system management.

2.6 Interrupt Request (IRQ-)

The Interrupt Request input signal is used to request that an interrupt sequence be initiated. When the IRQ Disable (I) flag is cleared, a low input logic level initiates an interrupt sequence after the current instruction is completed. The Wait-for-Interrupt (WAI) instruction may be executed to ensure the interrupt will be recognized immediately. The Interrupt Request vector address is OOFFFE,F (Emulation mode) or OOFFEE,F (Native mode). Since IRQ- is a level-sensitive input, an interrupt will occur if the interrupt source was not cleared since the last interrupt. Also, no interrupt will occur if the interrupt source is cleared prior to interrupt recognition.

2.7 Memory Lock (ML-)

The Memory Lock output may be used to ensure the integrity of Read-Modify-Write instructions in a multiprocessor system. Memory Lock indicates the need to defer arbitration of the next bus cycle. Memory Lock is low during the last three, five or nine cycles of ASL, DEC, INC, LSR, ROL, ROR, TRB, and TSB memory referencing instructions, depending on the state of the M and E8 flags.

2.8 Memory/Index Select Status (M/X)

This multiplexed output reflects the state of the Accumulator (M) and Index (X) select flags (bits 5 and 4 of the Processor Status (P) Register. Flag M is valid during the Phase 2 clock negative transition and Flag X is valid during the Phase 2 clock positive transition. These bits may be thought of as opcode extensions and may be used for memory and system management.

2.9 Non-Maskable Interrupt (NMI-)

A negative transition on the NMI- input initiates an interrupt sequence. A high-to-low transition initiates an interrupt sequence after the current instruction is completed. The Wait for Interrupt (WAI) instruction may be executed to ensure that the interrupt will be recognized immediately. The Non-Maskable Interrupt vector address is OOFFFA, B (8-bit Emulation mode), OOFFEA, B (16-bit Emulation mode) or OOFFDA, B (Native mode). Since NMI- is an edge-sensitive input, an interrupt will occur if there is a negative transition while servicing a previous interrupt. Also, no interrupt will occur if NMI- remains low.

2.10 Phase 2 In (PHI2)

This is the system clock input to the microprocessor internal clock generator. During the low power Standby Mode, PHI2 may held in the high or low state to preserve the contents of internal registers. However, usually it is held in the high state.

When the state of the property of the state of the state

S. NYTYMO AN ANALOGUE MODERN TO ANTI- O

MARCH 1990

18 Language nolinearmy E. C.

2.11 Read/Write (R/W-)

When the R/W- output signal is in the high state, the microprocessor is reading data from memory or I/O. When in the low state, the Data Bus contains valid data from the microprocessor which is to be stored at the addressed memory location. The R/Wsignal may be set to the high impedance state by Bus Enable (BE).

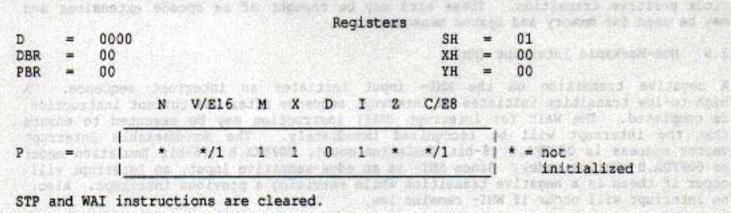
the na Jack Jestops of bear at landle desil resiped tour

2.12 Ready (RDY)

This bidirectional signal indicates that a Wait for Interrupt (WAI) instruction has been executed allowing the user to halt operation of the microprocessor. A low input logic level will halt the microprocessor in its current state. Returning RDY to the active high state allows the microprocessor to continue following the next PHI2 Clock negative transition. The RDY signal is internally pulled low following the execution of a Wait for Interrupt (WAI) instruction, and then returned to the high state when a RES-, ABORT-, NMI-, or IRQ- external interrupt is provided. This feature may be used to eliminate interrupt latency by placing the WAI instruction at the beginning of the IRQ- servicing routine. If the IRQ- Disable flag has been set, the next instruction will be executed when the IRQ- occurs. The processor will not stop after a WAI instruction if RDY has been forced to a high state. However, this feature should only be used on ASIC's and the RDY buffer modified. The Stop (STP) instruction has no effect on RDY. I would be the state of the state of

2.13 Reset (RES-) a for our sign you and the last part of the end to restaurate

The Reset input is used to initialize the microprocessor and start program execution. The Reset input buffer has hysteresis such that a simple R-C timing circuit may be used with the internal pullup device. The RES- signal must be held low for at least two clock cycles after VDD reaches operating voltage. Ready (RDY) has no effect while RES- is being held low. During the Reset conditioning period, the following period, the following processor initialization takes place:



				Signals			
E8	=	1			VDA	-	0
E16		10 10070	Learnedal 3	a the adexapropert	VP-		1
M/X	-	1 wor de	Maker and b	L PAIR Way haid i	VPA	-	0
R/W-	-	list st 3	E Williams	remember a Rowers	Larren	in.	100
SYNC	-	0					

When Reset is brought high, an interrupt sequence is initiated:

- o R/W- remains in the high state during the stack address cycles.
- o The Reset vector address is OOFFFC, D.

5 married and all suit seed, told and befored single the Contents

2.14 Valid Data Address (VDA) and Valid Program Address (VPA)

These two output signals indicate valid memory addresses when high logic 1, and are used for memory or I/O address qualification.

VDA	VPA	
0	0	Internal Operation-Address and Data Bus available. The Address Bus may be invalid.
0	1	Valid program address-may be used for program cache control.
1	0	Valid data address-may be used for data cache control.
1	1	Opcode fetch-may be used for program cache control and single step control

2.15 VDD and VSS

VDD is the positive supply voltage and VSS is system logic ground.

2.16 Vector Pull (VP-)

The Vector Pull output indicates that a vector location is being addressed during an interrupt sequence. VP- is low during the last two interrupt sequence cycles, during which time the processor reads the interrupt vector. The VP- signal may be used to select and prioritize interrupts from several sources by modifying the vector addresses.

MARCH 1990 14

word as an install accelerate above SECTION 3 the so show painteraction only and

ADDRESSING MODES

The W65C832 is capable of directly addressing 16 MBytes of memory for program space and 4GBytes for data space although only 24 bits (16MBytes) of address space are available on the standard product. This address space has special significance within certain addressing modes, as follows:

3.1 Reset and Interrupt Vectors

The Reset and Interrupt Vectors use the majority of the fixed addresses between 00FFD0 and 00FFFF.

3.2 Stack to said brids her bacon set pateration studies will

The Stack may use memory from 0000000 to 00FFFF. The effective address of Stack and Stack Relative addressing modes will be always be within this range.

3.3 Direct

The Direct addressing modes are usually used to store memory registers and pointers. The effective address generated by Direct, Direct, X and Direct, Y addressing modes is always in Bank 0 (000000-00FFFF).

Sabata V makes - Steen -

3.4 Program Address Space

The Program Bank register is not affected by the Relative, Relative Long, Absolute, Absolute Indirect, and Absolute Indexed Indirect addressing modes or by incrementing the Program Counter from FFFF. The only instructions that affect the Program Bank register are: RTI, RTL, JML, JSL, and JMP Absolute Long. Program code may exceed 64K bytes although code segments may not span bank boundaries.

3.5 Data Address Space

The Data Address space is contiguous throughout the 16 MByte address space. Words, arrays, records, or any data structures may span 64 KByte bank boundaries with no compromise in code efficiency. The following addressing modes generate 24-bit effective addresses in W65C816 Emulation mode and some, where noted by (*), generate 32-bit effective address in W65C832 native mode.

includes and the base of the property on

- o Direct Indexed Indirect (d,x)
- * Direct Indirect Indexed (d), y
- o Direct Indirect (d)
- o Direct Indirect Long [d]
- * Direct Indirect Long Indexed [d], y
- o Absolute a
- * Absolute a, x
- * Absolute a, y
- o Absolute Long al
- * Absolute Long Indexed al, x
- * Stack Relative Indirect Indexed (d,x),y

(A) PORTING THE E

For Monthson Cl. 1

Direct Street Street

The Especial prior are lead.

(truly because coming well-last found a

The following addressing mode descriptions provide additional detail as to how effective addresses are calculated.

Twenty-four addressing modes are available for the W65C832. The 32-bit indexed addressing modes are used with the W65C832; however, the high byte of the address is not available to the hardware on the standard W65C832 but is available on the core for ASIC's. Detailed descriptions of the 24 addressing modes are as follows:

3.5.1 Immediate Addressing-# The operand is the second byte in 8-bit mode, second and third bytes when in the 16-bit mode, or 2nd thru 5th bytes in 32-bit mode of the instruction.

3.5.2 Absolute-a

With Absolute addressing the second and third bytes of the instruction form the low-order 16 bits of the effective address. The Data Bank Register contains the high-order 8 bits of the operand address.

Instruction: | opcode | addrl | addrh | Operand Address: | DBR | addrh | addrl |

3.5.3 Absolute Long-al

Instruction: | opcode | addrl | addrh | baddr | Operand Address: | baddr | addrh | addrl |

3.5.4 Direct-d

The second byte of the instruction is added to the Direct Register (D) to form the effective address. An additional cycle is required when the Direct Register is not page aligned (DL not equal 0). The Bank register is always 0.

> Instruction: | opcode | offset | | Direct Register | | offset | Operand Address: 00 |effective address|

3.5.5 Accumulator-A

This form of addressing always uses a single byte instruction. operand is the Accumulator.

3.5.6 Implied-i

Implied addressing uses a single byte instruction. The operand is implicitly defined by the instruction. IN LET TENED IN THE THE WORLD ST

16 **MARCH 1990**

Spenis and Jy

Lamintol

Complete Transaction St. Tolly

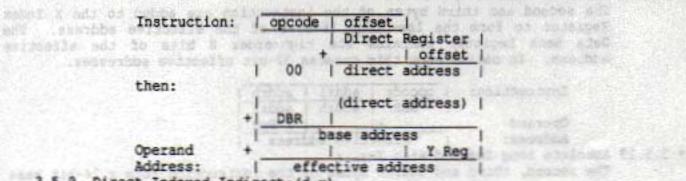
* 3.5.7 Direct Indirect Indexed-(d), y

This address mode is often referred to as Indirect, Y. The second byte of the instruction is added to the Direct Register (D). The 16-bit contents of this memory location is then combined with the Data Bank register to form a 24-bit base address. The Y Index Register is added to the base address to form the effective address. In native mode this creates 32-bit effective addresses.

Instruction:	1	opcod	e offset
	100	4 C	Direct Register
			+ offset
	T	00	direct address
then:			antique l'adit to y
Harrist St. Co.	L	00	(direct address)
	+1	DBR	A second are actually
	1		base address
Operand	+	-	I Y Reg I
Address:	al.	ef	fective address

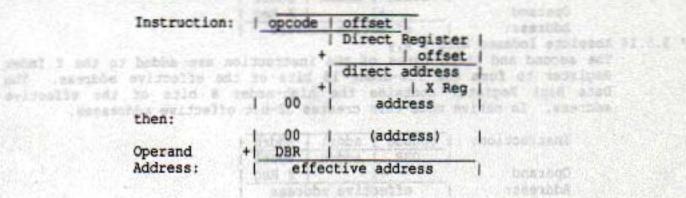
* 3.5.8 Direct Indirect Long Indexed-[d], y

With this addressing mode, the 24-bit base address is pointed to by the sum of the second byte of the instruction and the Direct Register. The effective address is this 24-bit base address plus the Y Index Register. In native mode this creates 32-bit effective addresses.



3.5.9 Direct Indexed Indirect-(d,x)

This address mode is often referred to as Indirect, X. The second byte of the instruction is added to the sum of the Direct Register and the X Index Register. The result points to the low-order 16 bits of the effective address. The Data Bank Register contains the high-order 8 bits of the effective address.



Descaped.

bubble at

said show evil

end we sed because

MAGNETER TIME

thinked Sebucher

3.5.10 Direct Indexed With X-d, x

The second byte of the instruction is added to the sum of the Direct Register and the X Index Register to form the 16-bit effective address.

The operand is always in Bank 0.

(L) -beached complete dearle

DS DEBO PRIO D

LABOURISA.

WILLY SELECT MARKET

bose and to stor

mistic extraolity

Show wellen under

2 30 2526

Instruction:	opcode	offset	Table of the same
		Direct	Register
	+	AND THE PERSON NAMED IN	offset
		direct	address
Operand	44 Kentere	13.4 E	X Reg
Address:	1 00 1	effectiv	ve address

3.5.11 Direct Indexed With Y-d,y

The second byte of the instruction is added to the sum of the Direct Register and the Y Index Register to form the 16-bit effective address. The operand is always in Bank 0.

Instruction:	opcode	offset	Harris Harris
		Direct	Register
	+	21 61 -91	offset
on al sendide	ALCOHOL: USA	direct	address
Operand	a spaties +	100	Y Reg
Address:	1 00	leffecti	ve address!

* 3.5.12 Absolute Indexed With X-a,x

The second and third bytes of the instruction are added to the X Index Register to form the low-order 16-bits of the effective address. The Data Bank Register contains the high-order 8 bits of the effective address. In native mode this creates 32-bit effective addresses.

Instruction:	opcode	addrl	addrh
	DBR	addrh	addrl
Operand	+		X Reg
Address:	effe	ctive add	iress

* 3.5.13 Absolute Long Indexed With X-al,x

The second, third and fourth bytes of the instruction form a 24-bit base address. The effective address is the sum of this 24-bit address and the X Index Register. In native mode this creates 32-bit effective addresses.

8-19010-N	Instruction:	opcode addrl	addrh	baddr
		baddr addrh	addrl	STATE OF THE SECOND
	Operand	+	X Reg	
	Address:	effective ad	dress	

* 3.5.14 Absolute Indexed With Y-a,y

The second and third bytes of the instruction are added to the Y Index Register to form the low-order 16 bits of the effective address. The Data Bank Register contains the high-order 8 bits of the effective address. In native mode this creates 32-bit effective addresses.

Instruction:	opcode	addrl	addrh
	DBR	addrh	addrl
Operand	1000000	No. of Contract of	Y Reg
Address:	effe	ctive add	dress

3.5.15 Program Counter Relative-r

This address mode, referred to as Relative Addressing, is used only with the Branch instructions. If the condition being tested is met, the second byte of the instruction is added to the Program Counter, which has been updated to point to the opcode of the next instruction. offset is a signed 8-bit quantity in the range from -128 to 127. Program Bank Register is not affected.

3.5.16 Program Counter Relative Long-rl

This address mode, referred to as Relative Long Addressing, is usd only with the Unconditional Branch Long instruction (BRL) and the Push Effective Relative instruction (PER). The second and third bytes of the instruction are added to the Program Counter, which has been updated to point to the opcode of the next instruction. With the branch instruction, the Program Counter is loaded with the result. With the Push Effective Relative instruction, the result is stored on the stack. The offset is a signed 16-bit quantity in the range from -32768 to 32767. The Program Bank Register is not affected.

3.5.17 Absolute Indirect-(a)

The second and third bytes of the instruction form an address to a pointer in Bank 0. The Program Counter is loaded with the first and second bytes at this pointer. With the Jump Long (JML) instruction, the Program Bank Register is loaded with the third byte of the pointer.

> Instruction: | opcode | addrl | addrh | Indirect Address = | 00 | addrh | addrl | New PC = (indirect address) with JML: New PC = (indirect address) New PBR = (indirect address +2)

3.5.18 Direct Indirect-(d)

The second byte of the instruction is added to the Direct Register to form a pointer to the low-order 16 bits of the effective address. The Data Bank Register contains the high-order 8 bits of the effective address.

Instruction:	1	opcode	1	offset		
	915	10-138	-	Actual Visit of the Association Co.	Register	8
		THE P	+		offset	
	-1	00	-1	direct	address	
then:		adde y				
	-1	00	-1	(direct	address)	
Operand	+1	DBR	1	L and		
Address:	30	eff	ec	tive ad	dress	-1

3.5.19 Direct Indirect Long-[d]

The second byte of the instruction is added to the Direct Register to form a pointer to the 24-bit effective address.

Instruction:	1	opcode	offset	1	
		I Suprem	Direct	Register	1
		*		offset	1
		00	direct	address	1
then:		ATE \$1500000			
Operand Address:	1	(dir	ect add	ress)	1

zeerbba

Name of Particular St. P. P.

a carpon and of inton

3.5.20 Absolute Indexed Indirect-(a,x)

a evisated toombod mercory 22 5 E The second and third bytes of the instruction are added to the X Index Register to form a 16-bit pointer in Bank 0. The contents of this pointer are loaded in the Program Counter. The Program Bank Register is and north not changed. get he obecome and of shade at he seage and sad most somes and my watermany about bombly a hit isside.

Instruction:	Opcode	addrl	l addrh l	mag sons	
			addrl		BUTTONE !
Texas DA DOOL S	n Smilet a	A SECTION OF THE PARTY OF THE P	X Reg		this sint
Cista institution	PBR	add		benomial)	ME THE
then:	947 J.198	Wi moulde	rateur pa	LINETON C	miscoulus.
dan spilin del	PC = (add	ress)	ox webs	STAR CO.	APPLICATION.

3.5.21 Stack-s

wine bear at the

MINE TO REPORT DE

Stack addressing refers to all instructions that push or pull data from the stack, such as Push, Pull, Jump to Subroutine, Return from Subroutine, Interrupts, and Return from Interrupt. The bank address is always 0. Interrupt Vectors are always fetched from Bank 0.

3.5.22 Stack Relative-d, s

The low-order 16 bits of the effective address is formed from the sum of the second byte of the instruction and the stack pointer. The high-order 8 bits of the effective address is always zero. The relative offset is an unsigned 8-bit quantity in the range of 0 to 255.

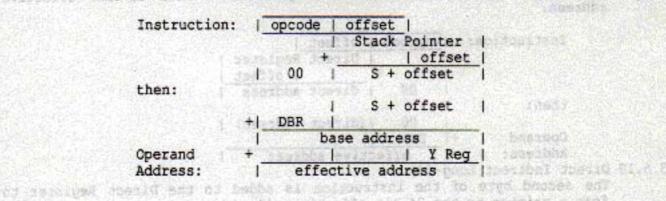
Instruction:	opcode	offset	obge in this
	ent participation	Stack	Pointer
Operand	+	ACCITECTS	offset
Address:	1 00	effectiv	ve address

* 3.5.23 Stack Relative Indirect Indexed-(d,s),y

The second byte of the instruction is added to the Stack Pointer to form a pointer to the low-order 16-bit base address in Bank 0. The Data Bank Register contains the high-order 8 bits of the base address. The effective address is the sum of the 24-bit base address and the Y Index Register. In the native mode this creates 32-bit effective addresses.

form a pointer to the 24-bit effective address.

Testing | modern | collected | differ



neinaged Joseph 1

taeschie Santibl

the design

Bearings

grade and A beautiful power bank the fire

An Ithel Sept.

The second of the second The state of the state of

A ve becober assistant

THE RESERVE OF

3.5.24 Block Source Bank, Destination Bank-xya

This addressing mode is used by the Block Move instructions. The second byte of the instruction contains the high-order 8 bits of the destination address. The Y Index Register contains the low-order 16 bits of the destination address. The third byte of the instruction contains the high-order 8 bits of the source address. The X Index Register contains the low-order bits of the source address. The Accumulator contains one less than the number of bytes to move. When the Accumulator is zero it will move one byte. The second byte of the block move instructions is also loaded into the Data Bank Register. In W65C832 native mode this X Index Register contains the entire source address and the X Index Register contains the entire destination address; therefore, the instruction is shorter by two bytes and two cycles per byte moved.

Instruction: | opcode | dstbnk | srcbnk | dstbnk -> DBR Source Address: | srcbnk | X Reg Destination DBR | Y Reg Address:

> Increment (MVN) or decrement (MVP) X and Y. Decrement C (if greater than zero), then PC+3->PC.

In W65C832 native mode these addressing modes creates 32-bit effective data space addresses.

Lorsont sorthers

princephoa Morro

9114

States Readillive

MARCH 1990 tras ishbigo 21

and to souly of bear of those posteriors to suppost a size than

Table 3-1 Address Mode Formats

Addressing Mode	Format	Addressing Mode	Format
Immediate	rdex Redbit er	Absolute Indexed by Y	id, y
10100nt 2012 19 5215 19	‡al	A STATE OF THE STATE OF	d, y a, y
1 6 907	#EXT		la,y
the double address.	‡ <d< td=""><td></td><td>lal.v</td></d<>		lal.v
9 PE 07 PERSON OF THE BUTTON	da canda da da	Sec. Disc. March 1980	!EXT, y
No analy concess and west	# <al< td=""><td>Absolute Long Indexed</td><td>EXT, y</td></al<>	Absolute Long Indexed	EXT, y
the Date Such Additions	#>d	by X	>a, x
on estimation and estimation	#>a	of Fraction spoots across a	>al,x
animant ethine eds on	#>al	MAN AND A SUL MAN S	al,x
ment by ave bytes and	#>EXT	Program Counter Relative	>EXT,x
	#^a	and Program Counter	a
	#^al	Relative Long	al
	#^EXT	in abordul and the can	(EXT)
Absolute	d	Absolute Indirect	(d) (!d)
	la a	STATE OF STA	(a)
	[al	200 200 200	(!a)
	! EXT		(!al)
A STATE OF THE RESERVE OF THE	EXT	War and Table and	(EXT)
Absolute Long	>d >a	Direct Indirect	(d) (<a)< td=""></a)<>
	Sal		(<a1)< td=""></a1)<>
	al		(<ext)< td=""></ext)<>
MANAGED STREET, SELECTOR	>EXT	Direct Indirect Long	[d]
Direct Page	d		>a ,
	<d <a< td=""><td></td><td>[>al] [>EXT]</td></a<></d 		[>al] [>EXT]
	<al< td=""><td>Absolute Indexed</td><td>(d, x)</td></al<>	Absolute Indexed	(d, x)
	<ext< td=""><td></td><td>(!d,x)</td></ext<>		(!d,x)
Accumulator	Α	100 E - 100 E	(a, x)
Implied Addressing	(no operand)		(!a,x)
Direct Indirect Indexed	(<d), td="" y<=""><td></td><td>(!al,x)</td></d),>		(!al,x)
	(<a) (y<="" td=""><td></td><td>(!EXT, x)</td></a)>		(!EXT, x)
	(Sal).V	Stack Addressing	(no
Disset Indisset	(<ext); td="" y<=""><td>Ctack Dalating</td><td>operand)</td></ext);>	Ctack Dalating	operand)
Direct Indirect Indexed Long	[d] y <di,v< td=""><td>Stack Relative Indirect Indexed</td><td>(d,s),y (<d,s),y< td=""></d,s),y<></td></di,v<>	Stack Relative Indirect Indexed	(d,s),y (<d,s),y< td=""></d,s),y<>
Indexed bong	(d), y	Indirect indexed	(<a, s),="" td="" y<=""></a,>
	(<a11.v< td=""><td></td><td>(<a1.s).v< td=""></a1.s).v<></td></a11.v<>		(<a1.s).v< td=""></a1.s).v<>
	[<ext], td="" v<=""><td></td><td>(<ext, s),="" td="" y<=""></ext,></td></ext],>		(<ext, s),="" td="" y<=""></ext,>
Direct Indexed Indirect	(<d,x)< td=""><td>Block Move</td><td>d, d d, a</td></d,x)<>	Block Move	d, d d, a
Indirect	(<a, td="" x)<=""><td></td><td>d, al</td></a,>		d, al
	(<a1,x)< td=""><td></td><td>d, EXT</td></a1,x)<>		d, EXT
	(<ext,x)< td=""><td></td><td>a, d</td></ext,x)<>		a, d
Direct Indexed by X	d,x		a, a
	<d, td="" x<=""><td></td><td>a, al a, EXT</td></d,>		a, al a, EXT
	<a,x <a1,x< td=""><td></td><td>al,d</td></a1,x<></a,x 		al,d
	<ext, td="" x<=""><td></td><td>al,a</td></ext,>		al,a
Direct Indexed by Y	d,y <d,y< td=""><td></td><td>al.al</td></d,y<>		al.al
	<d, td="" y<=""><td></td><td>al,EXT</td></d,>		al,EXT
	<a1,y< td=""><td></td><td>EXT, d EXT, a</td></a1,y<>		EXT, d EXT, a
	₹EXT, y		EXT, al
Absolute Indexed by X	d,x		EXT, al EXT, EXT
	!d, x		
	a,x		
	a,x		
	al,x EXT,x		
	EXT, x		

Note: The alternate ! (exclamation point) is used in place of the | (vertical bar).

Table 3-2 Addressing Mode Summary

	Instruction In Memory	Cycles	Memory Util In Number o Sequence	f Program Bytes
Address Mode	Original 8-bit NMOS 6502		Original 8-bit NMOS 6502	
1. Immediate	1 2 1	2(3)	1 2 1	2(3)
2. Absolute	1 4(5) 1	4(3,5)	1 3 1	3
3. Absolute Long	1 - 1	5 (3)	1 - 1	4
4. Direct	1 3(5) 1	3(3,4,5)		2
5. Accumulator	1 2 1	2	1	1
6. Implied	1 2 1	2	1 1 1	1
7. Direct Indirect Indexed (d),y	5(1)	5(1,3,4)	1 2 1	2
 Direct Indirect Indexed Long [d], y 	-	6(3,4)	-	2
9. Direct Indexed Indirect (d,x)	6	6(3,4)	2 1	2
10. Direct, X	1 4(5) 1	4(3,4,5)	1 2 1	2
11. Direct, Y	1 4 1	4(3,4)		2 2 3
12. Absolute, X	1 4(1,5)	4(1,3,5)	1 3 1	3
13. Absolute Long, X	-	5(3)	SERVICE SHIPPING	4
14. Absolute, Y	1 4(1) 1	4(1,3)	1 3 1	3 2
15. Relative	1 2(1,2) 1	2(2)	3 2	2
16. Relative Long	1 - 1	3(2)	-	3
17. Absolute Indirect (Jump)	5 1	5	3	3 2 2 3
18. Direct Indirect	1 1	5(3,4)	- 1	2
19. Direct Indirect Long	1 - 1	6(3,4)	-	2
20. Absolute Indexed Indirect (Jump)	- 1	6	-	3
21. Stack	1 3-7	3-11	1-3	1-4
22. Stack Relative	1 -	4(3)	- 1	2
23. Stack Relative Indirect Indexed		7 (3)	- 	2 2
24. Block Move X,Y,C (Source, Destination, Block Length)	-	7 (6)	- 	3 (6)

Notes (these are indicated in parentheses):

- Page boundary, add 1 cycle if page boundary is crossed when forming address.
- Branch taken, add 1 cycle if branch is taken.
- 16 bit operation, add 1 cycle, add 1 byte for immediate.
 32 bit operation, add 3 cycles, add 3 bytes for immediate.
- 4. Direct register low (DL) not equal zero, add 1 cycle.
- Read-Modify-Write, add 2 cycles for 8-bit, add 4 cycles for 16-bit, add 8 cycles for 32-bit operation.
- For W65C832 native mode, subtract 2 cycles and 2 bytes.

A VOICE CONTROL OF THE STATE OF

SECTION 4

TIMING, AC AND DC CHARACTERISTICS

85.00 SH MAN

4.1 Absolute Maximum Ratings: (Note 1)

Table 4-1 Absolute Maximum Ratings

Rating	Symbol	Value
Supply Voltage	VDD	-0.3 to +7.0V
Input Voltage	VIN	1-0.3 to VDD +0.3V
Operating Temperature	TA	1 0 °C to +70°C
Storage Temperature	TS	1 -55 °C 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.

Notes:

Exceeding these ratings may result in permanent damage.
 Functional operation under these conditions is not implied.

V 1.0 1.0		
BENTAN TO STATE OF THE STATE OF		Closel
	dal (620 - dun plat (1467 M) Indian't y	
Au I	1150-5101-1000, 81-00-502, 101	
70 01	Ance Start Volet Start Control and Control	Load

emphilor.

THE THE

DEPT.

4.2 DC Characteristics: VDD = 5.0V +/- 5%,

VSS = OV,

TA = 00C to +700C

Table 4-2 DC Characteristics

Parameter	Symbol	Min	Max	Unit
Input High Voltage RES-, RDY, IRQ-, Data, BE PHI2, NMI-, ABORT-	Vih	The second secon	VDD+0.3 VDD+0.3	Committee of the Commit
Input Low Voltage RES-, RDY, IRQ-, Data, BE PHI2, NMI-, ABORT-	Vil	-0.3 -0.3	0.8 0.1*VDD	V V
<pre>Input Leakage Current (Vin = 0.4 to 2.4) RES-, NMI-, IRQ-, BE, ABORT- (Internal Pullup)</pre>	Tin	-100	1	uA
RDY (Internal Pullup, Open Drain) PHI2 Address, Data, R/W-, (Off State, BE=0)	Iin	-100 -1 -10	10 1 1 10	uA uA uA
Output High Voltage (Ioh=-100uA) Data, Address, R/W-, ML-, VP-, M/X, E8/E16 VDA, VPA	Voh	0.7 VDD	382 069 0010000	V
Output Low Voltage (Iol = 1.6mA) Data, Address, R/W-, ML-, VP-, M/X, E8/E16 VDA, VPA	Vol		0.4	v
Supply Current (No Load)	Idd		4	mA/MHz
Standby Current (No Load, Data Bus = VSS orVDD RES-, NMI-, IRQ-, SO-, BE, ABORT-, PHI2=VDD)		-	1	uA
Capacitance (Vin=0V, TA=25oC, f=2MHz) Logic,PHI2 Address, Data, R/W-(Off State)	Cin Cts	-	10 15	pF pF

4.3 General AC Characteristics: VDD= 5.0V +/- 5%, VSS= 0V, Ta= 0oC to +70oC

Table 4-3A W65C832 General AC Characteristics, 4-7MHz

Sold to the second second second		1 4 1	MHz	5 1	MHz	16	MHz	1 7	MHz	Comment.
Parameter Pha Carl	Symbol	Min	Max	Min	Max	Min	Max	Min	Max	Unit
Cycle Time	tCYC	250	DC	200	IDC .	1165	IDC	1140	IDC	l nS
Clock Pulse Width Low	tPWL	.125	10	.10	110	.082	110	1.07	110	us
Clock Pulse Width High	tPWH	1125	a, WA	100		182	E A	170	-	nS
Fall Time, Rise Time	tF, tR	-	10	-	110	-	1 5	1 -	5	l nS
A0-A15 Hold Time	tAH	110	**	110	-	10	1 -	110	-	nS
A0-A15 Setup Time	tADS	-	75	-	67	-	160	10-	160	nS
A16-A23 Hold Time	tBH	110	-	10	-	10	1 -	110	1 4	nS
A16-A23 Setup Time	tBAS	-	190	3.5	77	DE.	165	-	55	l nS
Access Time	LACC	130	-	1115	-	87	1-	160	-	l nS
Read Data Hold Time	t DHR	110	+	110	-	10	1	110	-	nS
Read Data Setup Time	tDSR	130	-	25	-	20	-	125	-	nS
Write Data Delay Time	tMDS	1	70	10-	65	-	160	-	155	nS
Write Data Hold Time	tDHW	110	-	110	-	10	-	110	-	l ns
Processor Control Setup Time	tPCS	130	+ 4	25	-	20	L.	120	-	l nS
Processor Control Hold Time	tPCH	110	-	110	+	10	-	10	1 -	l nS
E8/E16, MX Output Hold Time	tEH	110	-	10	3.0	5	1 -	1 5	-	l nS
E8/E16,MX Output Setup Time	tES	150		137	•	25	100	125	-	l nS
Capacitive Load *1	CEXT		100	10-3	100	-	135	-	135	pF
BE to Valid Data *2	tBVD	-	130	-	130	-	130	1	130	nS

Table 4-3B W65C832 General AC Characteristics, 8-10MHz

		8 1	MHz	9	MHz	110	MHz	100
Parameter	Symbol	Min	Max	Min	Max	Min	Max	Unit
Cycle Time	tCYC	125	DC	1110	DC	1100	DC	nS
Clock Pulse Width Low	tPWL	.062	10	.055	110	1.05	10	uS
Clock Pulse Width High	tPWH	62	-	155	1 -	150	1 -	l nS
Fall Time, Rise Time	tF,tR	-	5	-	1 5	-	1 5	l nS
A0-A15 Hold Time	tAH	10		110	-	110	1	l nS
AO-A15 Setup Time	tADS	7.0	140	-	140	-	140	l nS
A16-A23 Hold Time	tBH	10	-	110	1 -	110	-	l nS
Al6-A23 Setup Time	tBAS	-	45	-	145	-	145	nS
Access Time	tACC	70	-	170	1 -	170	-	nS
Read Data Hold Time	tDHR	10	-	110	-	110	-	l ns
Read Data Setup Time	tDSR	15	-	115	-	115		l ns
Write Data Delay Time	tMDS		40	-	140	-	40	l ns
Write Data Hold Time	tDHW	10	-	110	1 -	110	-	l ns
Processor Control Setup Time	tPCS	15	-	115	-	115	-	l ns
Processor Control Hold Time	tPCH	10	-	10	1 -	110	-	l nS
E8/E16,MX Output Hold Time	tEH	5	-	5	1000	1 5	-	l nS
E8/E16,MX Output Setup Time	tES	15	25	15	-	115	-	nS
Capacitive Load *1	CEXT	4-1	35	-	135	-	35	pF
BE to Valid Data	tBVD		30	-	130	-	30	nS

^{*1} Applies to Address, Data, R/W

26 191 BURAN MARCH 1990

^{*2} BE to High Impedence State is not testable but should be the same amount of time as BE to Valid Data

enst reachail de la lage de globe des

0 6267 81.55 8 8277 8 288 87 8885 828

4.4 General AC Characteristics: VDD= 1.2V, VSS= 0V,

Ta= 0oC to +70oC

Table 4-4A W65C832 General AC Characteristics, 40 KHz

THIS WALL WINGS IS GROW

90 7 10 29 12 12 14 90 7 10 29 12 12 14

med bios and been med quiel ared book mod to said by a mod book bios

ALT GENET COMMENTS OF STREET OF STRE

The Asset to Address, Date, SA

EN DE LO PALLE DATE

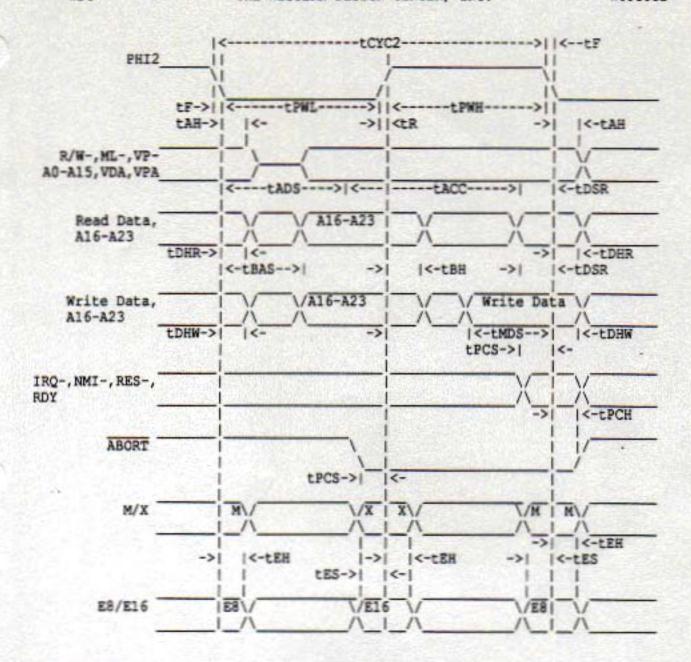
C Min Max Michigan to the Charles and	belmeE1	40 1	KHZ	ks:108
Parameter	Symbol	Min	Max	Unit
Cycle Time	tCYC	20	25	uS
Clock Pulse Width Low	tPWL	12.5	13	uS
Clock Pulse Width High	tPWH.	12.5	-	uS
Fall Time, Rise Time	tF,tR	-	10	nS
AO-A15 Hold Time	tAH	10	17.59	nS
AO-A15 Setup Time	tADS	Marie I	2	uS
AAO-A23 Hold Time	tBH	10	-	nS
A16-A23 Setup Time	tBAS		2	uS
Access Time	tACC	35	N 700	uS
Read Data Hold Time	tDHR	100	-	nS
Read Data Setup Time	tDSR	1.5	1	uS
Write Data Delay Time	tMDS.		2	uS
Write Data Hold Time	tDHW	10	5-1	nS
Processor Control Setup Time	tPCS	1.5	27.0	uS
Processor Control Hold Time	tPCH	100	THE W	nS
E8/E16, MX Output Hold Time	tEH	10	250	nS
E8/E16, MX Output Setup Time	tES	100	-	nS
Capacitive Load *1	CEXT	-	100	pF
BE to Valid Data *2	tBVD	-	30	nS

^{*1} Applied to Address, Data, R/W

MARCH 1990 27 86 1 1990

many la propose some gris of bloods and planted for what we conseque folk on Ha . Ex

^{*2} BE to High Impedance State is not testable but should be the same amount of time as BE to Valid Data



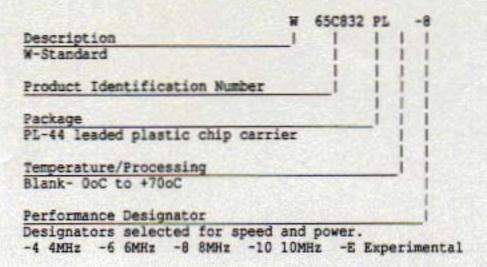
Timing Notes:

- Voltage levels are V1<0.4V, Vh>2.4V.
- 2. Timing measurement points are 0.8V and 2.0V.

Figure 4-1 General Timing Diagram

SECTION 5

ORDERING INFORMATION



General sales or technical assistance, and information about devices supplied to a custom specification may be requested from:

The Western Design Center, Inc. 2166 East Brown Road Mesa, Arizona 85213

Phone: 602-962-4545 Fax: 602-835-6442

WARNING:

MOS CIRCUITS ARE SUBJECT TO DAMAGE FROM STATIC DISCHARGE

Internal static discharge circuits are provided to minimize part damage due to environmental static electrical charge build-ups. Industry established recommendations for handling MOS circuits include:

- Ship and store product in conductive shipping tubes or conductive foam plastic. Never ship or store product in non-conductive plastic containers or non-conductive plastic foam material.
- Handle MOS parts only at conductive work stations.
- 3. Ground all assembly and repair tools.

MATCH 1995

SECTION 6

APPLICATION INFORMATION

Table 6-1 W65C832 Instruction Set-Alphabetical Sequence

ADC		PHA	Push Accumulator on Stack Push Data Bank Register on Stack
AND	"AND" Memory with Accumulator	PHD	Push Direct Register on Stack
ASL	AND* Memory with Accumulator Shift One Bit Branch on Carry Clear (Pc=0) Branch on Carry Set (Pc=1) Branch if Equal (Pz=1) Bit Test Branch if Result Minus (Pn=1) Branch if Not Equal (Pz=0) Branch if Result Plus (Pn=0) Branch Always Force Break Branch Always Long Branch on Overflow Clear (Pv=0)	PHK	Push Program Bank Register on
BCC	Branch on Carry Set (Po=1)	PHP	Push Processor Status on Stack
BEO	Branch if Equal (Pr=1)	PHX	Push Index X on Stack
BIT	Bit Test	PHY	Push Index Y on Stack
BMI	Branch if Result Minus (Pn=1)	PLA	Pull Accumulator from Stack
BPL	Branch if Result Plus (Pn=0)	PLD	Pull Direct Register from Stack
BRA	Branch Always	PLP	Pull Processor Status from Stack
BRK	Force Break	PLX	Pull Index X from Stack
BVC	Branch on Overflow Clear (Py=0)	REP	Reset Status Bits
BVS	Branch Always Long Branch on Overflow Clear (Pv=0) Branch on Overflow Set (Pv=1) Clear Carry Flag	ROL	Rotate One Bit Left (Memory or
CLC	Clear Carry Flag	DOD	Accumulator)
CLD	Clear Decimal Mode Clear Interrupt Disable Bit Clear Overflow Flag Compare Memory and Accumulator	ROR	Rotate One Bit Right (Memory or Accumulator)
CLV	Clear Overflow Flag	RTI	Return from Interrupt
CMP		nme.	Datama farm Cabanastian
COP	Compare Memory and Index X	RTS	Return from Subroutine Subtract Memory from Accumulator
COP CPX CPY	Compare Memory and Index Y	-	with Borrow
DEC	Decrement Memory or Accumulator	SEP	Set Processor Status Bite
DEX	Decrement Index Y by One	STP	Store Accumulator In Memory
DEY	Decrement Index Y by One	STX	Store Index X in Memory
EOR	"Exclusive OR" Memory with	STY	Store Index Y in Memory
TMC	Accumulator	STZ	Transfer Accumulator to Index X
Line	by One	TAY	Transfer Accumulator to Index Y
INX	Increment Index X by One	TCD	Transfer C Accumulator to Direct
INT	Compare Memory and Index X Compare Memory and Index Y Decrement Memory or Accumulator by One Decrement Index X by One Decrement Index Y by One "Exclusive OR" Memory with Accumulator Increment Memory or Accumulator by One Increment Index X by One Increment Index X by One Increment Index Y by One Jump Long Jump to New Location Jump Subroutine Long Jump to New Location Saving Return	TOS	Transfer C Accumulator to Stack
JMP	Jump to New Location		Pointer Register
JSL	Jump Subroutine Long	TDC	Transfer Direct Register to C
LDA	Jump to New Location Saving Return	TRB	Accumulator Test and Reset Bit
LDX	Load Accumulator with Memory Load Index X with Memory	TSB	Test and Set Bit
LDY	Load Index X with Memory Load Index Y with Memory Shift One Bit Right (Memory or	TSC	Transfer Stack Pointer Register
LSR	Shift One Bit Right (Memory or	TCY	to C Accumulator Transfer Stack Pointer Register
			to Index X
MVP	Block Move Negative Block Move Positive No Operation "OR" Memory with Accumulator Push Effective Absolute Address	TXA	Transfer Index X to Accumulator
NOP	No Operation	TXS	Transfer Index X to Stack
PEA	Push Effective Absolute Address	TXY	Transfer Index X to Index Y
	on Stack	LIM	Transfer index i to accumulator
PEI	Push Effective Absolute Address	TYX	Transfer Index Y to Index X
PER	on stack Push Effective Program Counter	WAI	Wait for Interrupt Reserved for Future Use
T LLEN	Relative Address on Stack	XBA	Exchange B and A Accumulator
		XCE	Exchange Carry and Emulation E8
		XFE	Exchange Carry and Emulation E8 and Exchange Overflow and Emulation E16
			Emulation E16

For alternate mnemonics, see Table 7-3-1.

(2-14) 150 NOVINCE IN 1910

most buy though states, make has though make and at though and be

THE REAL PROPERTY.

and the A depart proves and is I maked leases

and proof and the second and the sec

mark housed this process

1-1-7 alder best ablement planted in the

surrably estimate action values

with Wilsertice Progres Constant

The second section is select than the second second

Mined his is

SEE 21280

Table 6-2 Vector Locations

W65C02 8- Emulation		W65C816 16-bit	Emulation
OOFFFE, F-IRQ-/BRK	Hardware/Software	OOFFEE, F-IRQ-	Hardware
OOFFFC, D-RESET-	Hardware	00FFEC, D- (Rese	rved)
OOFFFA, C-NMI-	Hardware	OOFFEA, B-NMI-	Hardware
OOFFF8, 9-ABORT-	Hardware	OOFFE8, 9-ABORT	-Hardware
00FFF6, 7- (Reserved)	CANELS AND THE STATE OF THE STA	OOFFE6,7-BRK	Software
OOFFF4, 5-COP	Software	OOFFE4,5-COP	Software

W65C832 Native	
OOFFDE, F-IRQ-	Hardware
OOFFDC, D- (Reserved)	
OOFFDA, B-NMI-	Hardware
OOFFD8, 9-ABORT-	Hardware
OOFFD6,7-BRK	Software
00FFD4,5-COP	Software

The VP output is low during the two cycles used for vector location access. When an interrupt is executed, D=O and I=1 in Status Register P.

16113 145 800

mild enter tracks him a start a lighter in heading

Y mobil or sold limited better Timber of the sold of the Sold of the sold of the sold of

Month of heavyleter to have the

beinger Stack Jointer Percenter

A sabirt

and replay to be retent

and the property of bearings

and the property of the

107 Til Nock Pointer Seguiter

Index I to Acquesiance Index N to Stack

to landic T

SCHOOL ST

THE RESERVE AND ADDRESS OF THE PERSON ADDRESS OF THE PERSON

TRUE PAR BOTTLE

TRIGIST

31

SET

32

Table 6-3 Opcode Matrix

								LSD	E							1.00	
		1	1	1	4	1		1			A		C	0			Ė
,	2 8	ORA Idai	COP s	ORAds 2 4	158 d 2 5	ORAd 2 3	ASLIS 2 5	CHA Idi	PHP 5	CRA #	ASL A	PHD:	758 a	ORA#	ASL s	ORAS 4 5	6
d	8PL:	ORA NO.Y	ORA Idi	ORA (d.st.y	TRB d 2 5	ORA d.s	ASL da	ORA Joly	1 2	ORALY 3 4	INCA	TCS:	TR6 #	ORA EX	ASL RA	ORA HL	1
4	359 a	AND (4.1)	250,00	AND de	90T d 2 3	AND d	BOL a	AND (d)	PUPs 1 ¢	MID s	ROLA 1 2	PLD:	BIT a	AND:	HOLa 3 6	AND H	
	2 2	AND Idly 2 5	AND (d)	AND Ideas	BIT 4.	AND da	POL 4x	AND Jdly	SEC.	AND RY	DEC A	TBC	BIT A.	AND AX	BCL E	AND NA	ŀ
1000	ATIA 1 /	EGR (6,1)	MDW MDW	EUR 4.5	MVF tyc	EDR d	158 a	EOR (e)	PHAS 1 3	EOR +	LSRA 1 2	PHK 1	JMP a	EORa 3 4	LSR a	EOR M	
	BVG#	EOR HILLY	EOR (d)	EOR (d *Ly	MVIII NYC	SORds 2 4	LSR da	EOR Jolly	CLI I	EOR sy	PHY 1	TCD	MAP at 4 4	EOR as	SRay 3 7	ECH III	
	HIS &	ADC (d.x)	PERS 3 6	ADC ds	STZ d	ADC 8	non d	ADC [d]	PLAs 1 4	ADC#	ROR A	RT). 6	JMP (a) 3 5	ADC a	ROR a	ADC M	
1	BVS r	ADC Int.y	ADG Idi	ADC (da).y	STZ d.a	ADC d,x	ROH d.x	ADC Jdly	SEII 1 2	ADC MY	PLY:	1001	JMP (a,x)	ADC Ax	ROR a.x	ADC NA	1000
	BRAT 2	STA (d,x) 2 6	BHL /1	STA d,s	2 3	STAd 2 3	STX d 2 3	\$14 0 2 6	DEY I	BIT *	TXA:	PHR a	STY a	STA a	5TX a	STA at	
	BCC /	STA Idly	STA (d)	STA (d.5).y	STY d.x	STAd.x	STX d.y	STA (d).y	TYA I	STA a.y 3 5	TX81	TXY!	STZ:	STA a.x 3 5	STZA.A	STA el.x	
-	LDY #	LDA (d,x)	LDX#	LDA d.s	LDY d	LDAd 2 3	LDX d	LDA (d)	TAY I	LDA+	TAX:	PLB:	LDV a	LDA a	LDX a	LDA al	1
i	9CS / 2 2	LDA (d).y	LDA (d)	LDA (d.s).y	LOV d.x	LOAds	LDX My	LDA [d).y	CLV!	LDA A.y	TSX:	TYXI	LDY ax	LDA a.s.	LDX 8.9	LOA at a	1
1	CITY I	CMF (d.x)	REP.4 2*3	CMP d.s	CPV d	CMP d 2 3	DEC d	CMP d	INY I	CMP #	DEX.	WALL	CPY a	CMP a	DEC A	CMP al	4
Ì	UNE -	CMP Idby	CMF (d)	CMP (d.s).y	PEI s	CMP d.x	DEC d.s.	CMP (d).y	CLD:	CMP ay	794X S	51P1	JAR 119	CMP ax	DEC AX	CMPHA	4
-	CPA+	58C (6.4)	SEP ·	38C d.s	CPX d 2 3	SBCd 2 3	INC 4	SHC [d]	100x :	SBC + 2 2	NOP I	XBA I	CPX a	SBC + 3 4	INC a	SBC M	1
	2 2	SNC HEY	58C (4)	STIC (d.st.)	PEA:	58C-d.x	114G-d,× 2 6	SBC JeLy	3ED i	SBC Ay	PLX :	XCE I	JSR (2.1) 3 * 6	58C a.x	INC ax	SBC at a	
i	0	1	2	3	4	5		1		9	A		c	0	E		

symbol	addressing mode	symbol	addressing mode
	immediate	[6]	direct indirect long
A	accumulator	(dL)	Greet indirect long indexed
	program counter relative	2	absolute
18	program counter sentine tony	2.5	absolute indexed (with x)
1	impried	2.9	absolute indexed (with y)
	Statk	al .	sbsolute long
d	direct	elx	absolute long indexed
d.	direct indexed (with x)	d.s	stack relative
et.y	direct indexed (with y)	(d.51.y	stack relative indirect indexed
(d)	direct indirect	(a)	absolute indirect
(d.xl	direct indexed indirect	(a,x)	absolute indexed indirect
idl.y	direct indirect indexed	xyc	block move
503-NO-L		1 1 2 2 2 To 1 1 1 1 1 1 1 1 1	AND ASSESSMENT OF THE PARTY OF

Op Code Malrix Legend

INSTRUCTION MNEMONIC	* = New W65C816/802 Opcodes	ACDRESSING MODE
BASE NO. BYTES	New W65C02 Opcodes Blank - NMOS 6502 Opcodes	BASE NO. CYCLES

Table 6-3 Opcode Matrix

8						2 5 =		LSO	THE STREET							500	
	0	51	2	3	4	5		1		9		8	C	D	E	F	17/2
2	BRK s	ORA (d.x)	COP s	ORAds 2*4	TSB d	ORAd 2 3	ASL d 2 5	OFA [d]	PHPs	ORA P	ASL A	PHD s	TSB a	ORA a	ASL 8	ORA al	Sept Services
1000	BPL:	ORA (d).y	ORA (d)	ORA (d.s).y	TRB d 2°5	ORAd,x	ASL d.x 2 6	ORA (d).y	CLCI	OHA a.y	100	TCS I	TRB a	ORA a,x	ASL a.s	ORA al.x	
	JSR #	AND (d,x) 2 6	151 M	ANO d.s	BIT d 2 3	AND d	ROL d 2 5	AND (d)	PLPs 1 4	AND a	ROL A	PLD s	BIT A	AND a	ROL a	AND al	The same of
	(3M) r	AND Idly 2 5	AND (d)	AND (d,s),y	BIT da	AND d.x	AOL d.x	AND Idly	SEC.	AND R.y	DEC A	TSC I	BIT a.	AND ax	BOL BA	AND BLX	
and district	ATLS 1 7	EOR (d,x)	WDM 2 2	EUR d.s	MVF xyc	EORd 2 3	LSR d 2 5	EOG (d)	PHA s	2 2	LSR A	PHK s	JMP a	EORa 3 4	LSR a	EGR at	C 120
-	BVG r	EOR (d) y	EOR (d)	EOR (d.st.y	MVN xyc	EOH da	LSR d.x	EOR Idly	CLI:	EOR s.y	PHY'S	rcp 1 * 2	JMP al 4 * 4	BORAX 3 4	LSRax 3 7	EOR al	THE PARTY
1	ATS 5	ADC (d.x)	PER S	ADC d.s	STZd 2°3	ADC 8	RORd 2 5	ADC [d]	PLAS 1 4	ADC #	HOR A	RTL 8	JMP (a) 3 5	ADC a	ROR a	ADC al	10000
1	BVS r	ADC (d),y	ADC Idi	ADC (d.s).y	STZ d.x	ADC d.x	ROR d.x	ADC Jdl y	SEII	ADC My	PLY 5	TDC I	JMP (a,x)	ADC Ax	ROR a.x	ADC alx	SS112-1
	BRA r	STA (d,x) 2 6	8AL ri	STA d.s	5TY d 2 3	STAd 2 3	STX d 2 3	STA IOI	DEY!	BIT #	TXA:	PHB s	STY a	STA a	STX a	STA al	
	8CC r	STA Idl.y	STA (d)	STA (d.s).y 2 * 7	STY d,x	STAdx 2 4	STX d.y	STA [d].y	TYAI 1 2	STA a.y	TXS (TXY 1	STZ 8	STA a.x	STZAX	STA el.x	190000
	LDY #	2 6	LDX#	LDA d,s	LOY d	LDAd 2 3	LDX d	LDA [d]	TAY I	LDA +	TAX:	PLBs 1 4	LDY a	LDA 8	LDX a	LDA al	
	BCS r	LDA Idi.y	LDA (d)	LDA (d,s),y	LDY d.x	LDA dx	LDX d.y	LDA [d].y	CLVI 1 2	LDA 8.9 3 4	TSX:	TYX:	LDY ax	LDA a.s.	LDX a.y	LDA alx	
l	2.2	CMF (d.x) 2 6	REPA 2*3	CMP d.s	CPY d 2 3	CMP d 2 3	DEC d	CMP d	INY I	CMP #	DEX:	WAII	CPY a	CMPa 3 4	DEC A	CMP at	
	UNE .	CMP (d).y	CMF (d)	CMP (d.s).y	PEIS 2 * 8	CMP d,x	DEC dx	CMP Jol.y	CLD i	CMP ay	PHX s	5TP1	JML ini	CMP 3,x 3 4	DEC ax	CMP at v	A
	CPX *	SBC (d.x) 2 6	SEP +	88C d.s	CPX d 2 3	SBCd 2 3	INC d	SBC [d]	INX I	SBC #	NOP I	XBA 1 1 * 3	CPX a	SBC a	INC a	SBC all	
	BEQ1	SBC (d).y	\$8C (d)	SDC (d.sl.y	PEAS 3 5	SBC d.x	NC d,x 2 6	SBC [d].y	SED:	SBC a.y		XCE I	JSR (0,x) 3 * 6	SBC a.x	INC ax	SBC al,x	W 100 mm
ì	0		2	3		5	8	7	8	9	A	8	C	D	EW	F	

symbol	addressing mode	symbol	addressing mode
	immediate	[d]	direct indirect long
A	accumulator	Idl.y	direct indirect long indexed
10000	program counter relative	4	absolute
rt	program counter relative long	8,3	absolute indexed (with x)
	imphed	3.9	absolute indexed (with y)
	stack	al	absolute long
d	direct	al.x	absolute long indexed
d.s	direct indexed (with x)	d,s	stack relative
dy	direct indexed (with y)	(d.51.y	stack relative indirect indexed
(d)	direct indirect	(a)	absolute indirect
(d,x)	direct indexed indirect	(a,x)	absolute indexed indirect
Idl.y	direct indirect indexed	хус	block move

Op Code Malrix Legend

MNEMONIC	* = New W65C816/802 Opcodes	ACDRESSING MODE
BASE NO. BYTES	New W65C02 Opcodes Blank = NMOS 6502 Opcodes	BASE NO. CYCLES

ATTENDED TO SERVICE

\$8.363

3758

報.

SMA

Table 6-4 Operation, Operation Codes and Status Register

MNE-							110			6	1					1				24	-			9.4			STAT	us c	-			MNE
MONIC		-		=	0	4	-	(6)	[4],	(d,k)	4	A.	3	al,x	3	-	=	0	Ē	E	(a)		2	(d.s),y	xyc	-	5 M	4 3 X D		Z C		MCNI 0
	OPERATION	1	2	3	4	5	6	7	8	9	10	12000	12	-	14	16	16	17	40.4460	18	20	21		20	24	N A		8 0	1	z c	ę.	1
ADC AND ASL BCC BCS	A+M+C+A A/M+A C- 15/7 0 -0 BRANCH IF C+1	29	B008	6F 2F	55 25 06	OA.	A	31	37	61 21	75 36 16	667	P P P P P P P P P P P P P P P P P P P		22	90 80		A COLUMN	72 32				63 23	EB		222	100		The second	ZZZC		ADC AND ASL BCC BCS
BEO BIT BMI BNE BPL	BRANCH IF Z * 1 AAM (NOTE 1) BRANCH IF N * 1 BRANCH IF Z * 0 BRANCH IF N * 0	153	20	100	24		の日日の	W. F. LT.			X		ac	The second		F0 30 00 10	0.5			をから		The same of		150	S 14 12	M- 1/				z		BEC BIT BM BNE BPL
BRA BRK BRL BVC BVS	BRANCH ALWAYS BREAK (NOTE 2) BRANCH LONG ALWAYS BRANCH IF V • 0 BRANCH IF V • 1		200		N. La	SELECTION OF SELEC	No. of P.		THE STATE OF					100	0186	50 50 70	92		The second			00	のはなが					• 0	でない			BRA BRK BRL BVC BVS
CLC CLU CLV CMP	0 - C 0 - D 0 - 1 0 - V	CS	CO	OF.	C6		18 D8 56 B8	01	07	CI	06	が	00	OF	09	The State of		0.00	CS	67	Topic .		3	01	1	N		á	0	20		CLOCK
CPY CPY CEC DEC	CO-PROCESSOR X-M Y-M DECREMENT X-1-X	E0 00	ECCE	BALLEY.	E4 C4 C8	34	GA	211	2 10	4	D6		0€		A STATE OF THE PARTY OF THE PAR	500		N IN		200		02				2222		0	2.1	NAMA OO		COP CPX CPY DEC
YEG ROS INC INX INY	Y-1 - Y AVM - A NGREMENTS X+1 - X	49	98	45	45 68	14	84 E8 C8	51	57	41	56 FB		SD FE	50	19	100	DO CO	1000	52	47	11.24	Part of the	43	53		2222		1		ZZ .	241	DEY ECA INC INX INY
JML PMC JSL JSL AGU	JUMP LONG TO NEW LOC. JUMP TO NEW LOC. JUMP LONG TO SUB. JUMP TO SUB. M - A	A9	4C 20 AD	55	Aá		100	Bi	87	Al	85	TO THE	80	er.	89	5) 117		DC 6C	82	AT	7C F0	1000	A3	60	100				4-8-6	ż	1000	JML JMP JSL JSR LDA
LDX LDY LSR MVN MVP	M - X M - Y 0 - [15/7 0] - C M - M NECATIVE M - M POSITIVE	SA	-		A6 A4 48	44		S THE S			84 56	500	SE SE	No. of	đE		The Later	1000	21	State of the last	STATE OF		ましま	2000年	5.5	N N O	100	01-1	記される	2 0	The second	LOX
ORA PEA	NO OPERATION AVM - A Mpc - 1, Mpc - 2 - Ms - 1, Ms S - 2 - 5 Mpd, Mpd - 1) - Ms - 1, Ms	09	co	oF	05		EA	11	17	01	15		10	1F	19		0.0000000		12	07		F4 04	03	13	***	N						NOP ORA PEA
PHA	S - 2 - 5 Mac + rl. Wpc + rl - 1 - Ms - 1. Wr S - 2 - S A - Ms, S - 1 - S		100	100		10.00		500								100						62 48 8B	100	100	9						1965	PER
PHB PHD PHK PHP	DBR - Ms, S - 1 - S D - Ms, Ms - 1, S - 2 - S PDR - Ms, S - 1 - S P - Ms, S - 1 - S X - Ms, S - 1 - S			100					11/2/1	No. of							1000				+	0B 46 56						1	1		*	PHB PHK PHP PHX
PHY PLB PLD	Y - Ms, S - 1 - S S + 1 - 5, Ms - A S + 1 - 5, Ms - CBR S + 2 - S, Ms - 1, Ms - D			0.000	191			-61			7						de	80	21			5A 58 AB 2B		加加		2222			_	2 .	*	PHY PLA PLB PLD
PLX PLY PLY POL	S-1-5 Ms - P S-1-5 Ms - X S-1-3 Ms - Y MAP - P MS-7 D - C	CZ	2E		26	2A			N 10 10 10		36	100	Æ		1						2-4	FA				NNNN N	M		1	2 0 2 0	*	PLX PLY REP ROL
IOR ITI ITL ITS IBC	RTRN FROM INT. RTRN FROM SUB LONG RTRN SUBROUTINE A-M-C-A	69	6E	er.	66 E5	AÞ		Fi	67	E1	78 F5		E	FF	F9				F2	E7	590	40 58 60	E3	F3		N V . V		k D		NO C	•	RCA RTI ATL ATS SBC
EC ED ED EP	1-C 1-0 1-1 MVP-P	€2	8D		85	1 N W	38 78 75	91	0	81	4.0			E COLOR	99					67	10.00	16. 16. 18.		80	COMPA	N V	м .		,	2 0	*	SEC SED SEI SEP STA
TP TX TY TZ	STOP (1 - φ2) X - Μ Y - Μ 100 - Μ A - X		8E 8C 9C		86 64 64		OB AA					96	JE .				a lung									N					•	STP STX STV STZ TAX
AY CD CS DC RB	6-5 0-5		ıc		14		A8 18 18 78												HANDER OF THE PERSON NAMED IN COLUMN TWO IS NOT THE PERSON NAMED IN COLUMN TWO IS NAMED IN COLUMN TWO IS NOT THE PERSON NAMED IN COLUMN TWO IS NAMED IN COLUMN T			n		100		N N		17.7	PARTICION			TAV TCD TCS TDC TRB
SB SC SX XA XA	AVM - M S - C S - X X - A X - S	\$100 to	oc.		04	38 BA 8A 9A					100					STATE OF						F	24	100	- 43	N N			. 2	-	•	TSB TSC TSX TXA TXA
XY YA YX YX YOM	X - Y Y - A Y - X O - RDY NO OPERATION (RESERVED)					96 98 98 CB 42			1000														No. of		- 0	N			2 2 2	1	io.	TXY TYA TYX WA.
BA CE	B - A C - E					EB FB								1			1		1							N	777	100	Z		•	XBA NCE

Notes:

1. Bit immediate N and V flags not affected. When M = 0, M15 - N and M14 -V.

2. Break Bit (B) in Status register indicates hardware or software break.

* • New W65C616/802 Instructions
 * • New W65C02 Instructions
Blane • NMO6 6502

- Add V OR - Subtract V Exclusive OR A ANO

Table 6-4 Operation, Operation Codes and Status Register

MNE-						13		*	7	C I			*	×							2			(8.	9	7			COCI			MON
IONIC		-		3	0	<	-	9	2	6	3	ş	9	3	4	-	E	0	9	2	=	*	4	9	×	N.	V M	X I	1 (2 (E:	0
	OPERATION	1	2	3	4	5	6	7	8	9	10	11	12			16	16	17	-	18	20	21	22	23	24	N	V 1	В	1 (2 (E •	_
ADC AND ASL BCC	A - M - C - A AAM - A C - 15/7 0 - 0 BRANCH IF C - 0	29	888	SF 25	92 25 06	a	N K	31	37	21	75 35 16	100	3D 1E	35	39	90	H	AND TO	32	27	100		63	H2		222				2 0	0	AND AND ASL BCG BCS
BCS	BRANCH IF C • 1	1			-	100	7						-0	4.5	4	1000	1	160	100	- 1	100			6.00	20		100		li isi	RD.		BEC
BEQ BIT BMI BNE	BRANCH IF Z = 1 AAM (NOTE 1) BRANCH IF N = 1 BRANCH IF Z = 0	829	20	T T	24	2	1.02	101.00	48 100		34	(0)	30	and the same	100	900	1	11.34	100		100		100			M- A	4			2		BM
BPL	BRANCH IF N = 0						2					100	201		-	10									2.5	200		200	100.7	NAME OF TAXABLE		364
BAK BAL BVC BVS	BRANCH LONG ALWAYS BRANCH LONG ALWAYS BRANCH IF V = 0 BRANCH IF V = 1		200 18		100		4 - 74	SAI S		357		我 他 5		and a second	900	200	82					00	100			100		•				BFX BRL BVC BVS
CLC	0-C						16	8		- 1			1					20	23	13	15			-	20	1		-		. (CLL
CLO	0-0	11	00	電	41	96	16 D8 55	KO!	6	133		211	TS.	13	n b	40,	B	4.13	W	19	254	253	100	28	131	100			0	RO	10	CTI
CLV	A-M	CS	CO	OF.	CS	9	Co i	01	D7	CI	DG	1	00	OF	De	.3			CZ	C7	15	d.	CS	03		N	0	1		2		CM
COP	CO-PROCESSOR	4			100	93	9.6	0.8	0.0	1		21	200		R.	Delka	10	15		1	154	02	5.0	85	-		974		0 1		*	CO
CPX CPY DEC	Y-M DECREMENT X-1-X	E	E885	1	64 64 68	34	CA	5.		13	D6	in in	0€		100	1	17	0	100		5	100	20	100	68	2222				ZZZZ		CP
CEY	Y-1 - Y				provi		84				100				100	-	100	-			100	7			5	N	1	3	S. A.	Z		DE
NC NX NY	NCREMENTS	49	2E	45	45 E6	(S)	ES CS	51	57	41	56 F6	100	SD FE	54	19		2000	0 11	52	4	70	1	47	33	100	2222				ZVZ	8000	INX INX
JML	JUMP LONG TO NEW LCC.		1	-			1	3			1	7		7	. 3	8		DC 6C		1	70	V.	13	31		. 9		3017		8 8	*	M
JMP JSL	JUMP TO NEW LOC. JUMP LONG TO SUB.		40	22	8.	= 14			8	100			100		100	100		~	53	30	FC		3	100	1	12	Kin.	3	12	3633	1	SE
愚	M-A	AS	AD	AF	Aś	E.	B	81	87	AI	85	- 2	80	8F	89	1 21	11		82	AT		100	A3	63		N			1	ż	1 3	LDV
LDX	M · X		AE		AG Ad		6.70	0.07	1	12%	94	136	BC		BΕ	863	1	100	33	50.3	. Pro	1	GI.	12.1	A.	N	100	3		2		LO
LDY LSR MVN MVP	M - M NEGATIVE	-	46	1		44		3			56	100	SE		-		-	-	188	1		1	- 1	0.00	54	0		-	No. of	2 (:	MY
NOP	NO OPERATION	200	co	16	05	18	EA	11	17	01	15	1.5	10	1F	19	18	6	20	12	07			63	13		N	100		. 1	ż	-	NO OR
PEA PEI	AVM - A Mpc - 1, Mpc - 2 - Ms - 1, Ms S - 2 - S M(d), M(d - 1) - Ms - 1, Ms	ľ	0		us			9	*			ch				100	i i				2	F4 04			1					Can.		PEI
PER	S - 2 - 3 Moc - rt. Wpc - n - 1 - Ms - 1, W S - 2 - S		100	73	1	1		100		Ç.			1			000		200	2		200	62	4	Se					-	5	*	PE
PHA	A - Ms. S - 1 - 5 DBR - Ms. S - 1 - 5	il is	10	78	28		0.5	3				w 22	130	12	YN	(Dec	4	5	90	18	12	88	C GP	12.	100	1		1				PHI
PHID PHIK PHIP	D - Ms, Ms - 1, S - 2 - S PIR - Ms, S - 1 - S P - Ms, S - 1 - S	-		7		201		-	1	100		-	T			*		3			100	48 38		4							*	PHI
PHY	X-Ma.S-1-8 Y-Ma.S-1-8		13		700	PAL.			8								00		3		15	DA SA	33		1				DE A			PH
PLA PLB PLD	S - 1 - 5, Ms - A S - 1 - 5, Ms - CBR S - 2 - 5, Ms - 1, Ms - D				SIT			250	0.55			egi gi		200			ni ni	DIR.	94	in the	Polis	58 AB 28	i	100	28	777	Vá			Z	*	PLE
PLX	S-1-5 Ms-P S-1-5 Ms-X		18	5		2	No.		10		46	14		18			0		原	Lie!	10	FA		1	15	22	· M	× 1		ž'		PLE
PLY	9-1-3 Ms-Y	CZ	jes jes	183		000						5			18	8				3	MESO	**		5		N	V M	× 1	0 1	Z	*	PE
ROL	(15/7 0) - C-		2E	K	26	24					36	3	36	100						6					10	N				2 (RO
ROR	-C - (15/7 0)		65	100	66	6A		6	896		76		75		1	3			18	13	761			18		N				2 9		RO
ATL	ATRN FROM INT. ATRN FROM SUB LONG		3				19	2	6	8		10		12			B		1	00	3	42 58 60		98		No.		× 1		2 (ATI
ATS SBC	A-M-C-A	63	EO	27	25	000	150	FI	FT	E1	FS	100	FD	FF	69				F2	E7	3	60	E3	F3		N	v			ž (SBC
	1-0			U.			38		100	1	60	10				126	22	100	Section	P.LO	-	-			00	1		-	2 4	2		SEC
SEC SED SEP	1-0			100	100	200	75		(80				fa			F			16	100	107	ul.	k		0				. 1			SEL
SEP	MVP - P A - M	E2	ap	aF	85	80	1.01	91	97	51	95	1	90	gr	99	U B	411	rich!	92	57	127	23	as	20	181	N	V M	X .		2 9		SEI
STP	STOP (1 - 42)		100	-	200		08	.80		1	68	10		130	13	16		3	EU		12		le,	10		4		V		200	•	STE
STX	X-W		9C	2.00	84	58	F	55		100	94	98	1	18		1			1	1	192		1/2									ST
STZ	00 - M A - X		90	131	64	1	AA	die			74	QUI	9E	E					THE STATE OF	Ē		1				Ň.	-:			Z	1	TAX
TAY	A-Y		12	1	V.		AB	-83	145	12	6	CQ)	6.	13	50	4	N			12	ev.	18	8	E S		N	100	30		Z	*	TAN
TCS	C-5		rei.		1	100	18 18 78			155			1000 1100	100			0	1	HQ.	63			4	100	12.0						*	IC
TDC	D-C	8	IC	93	14	E3	78		50		Si			200	2				1			924	City.	G.	1	1	4			Ž	•	TRE
TSB	AVM - M		OC.	_	04			10	V		77	'n	(55)	1	23	W = 1.0	45	SH	133			30		7		è				2	:	TSE
TSC TSX TXA TXS	S-C S-X X-A	3	No.			28 A A A	100	(Allen		900			IN-				60		100	116		9	200	8	10	N				2		TSC TSX TXX TXX
TXY	X-Y	+				96					E h				+ 9	1	1	127		13	- 43		T			N	•000			Z	•	TXY
TYA	Y-A		100		13	98			1	32		NI.	217							3	23		1			N	1/15		200	2	*	TYX
WA	Q-ROY			15		CB 42		2		100		22		-87		200	×	7		-			100	8	1	93					:	WA
		+		100		_			10	10			57			7-0-				1	129	5				N	No.		FIL	Z		XBA
TYX	Y-X	SERVEDI	SERVEDI	ERVEDI	SERVEDI	SERVEDI	ERVEDI EB												The same of the sa	THE RESERVE THE PERSON NAMED IN COLUMN TWO IS NOT THE PERSON NAMED IN COLUMN TWO IS NAMED I			EA FB	EA FB	En FB		ER FB	ER P	EB FB N	ER P	EB FB N Z	EB FB N Z E

Notes:

1. Bit immediate N and V flags not affected. When M = 0, M15 - N and M14 -V.

2. Break Bit (B) in Status register indicates hardware or software break.

^{3. * -} New W65C616-802 Instructions • - New W65C02 Instructions Blans - NWC6 6502

⁻ Add - Subtract A AND

V OR V Excusive OR

Table 6-5 Instruction Operation

	ADDRESS MODE	C	YCLE	VI	P, ML	(14) VDA,		(15) ADDRESS BUS	DATA BUS I	R/W
1.	Immediate-# (LDY, CPY, CPX, LDX, ORA,		1.	1	1	1 0	1	PBR, PC PBR, PC+1	OpCode IDO	1
	AND, EOR, ADC, BIT, LDA, CMP, SBC, REP, SEP) (14 OpCodes) (2, 3 and bytes) (2, 3 and 5 cycles)	(1)	2a.	1	1	0	1	PBR, PC+2-4	ID1-3	1
2a.	Absolute-a		1.	1	1	1	1	PBR, PC	OpCode	1
	(BIT, STY, STZ, LDY,		2.	1	1	0	1	PBR, PC+1	AAL	1
	CPY, CPX, STX, LDX,		3.	1	1	0	1	PBR, PC+2	AAH	1
	ORA, AND, EOR, ADC,	600			1	1	0	DBR, AA	Byte 0 1/	
	STA, LDA, CMP, SBC)	(1)	4a.	1	1	1	0	DBR, AA+1-3	Bytes1-3 1/	0
	(18 OpCodes) (3 bytes) (4, 5 and 7 cycles									
2h.	Absolute-(R-M-W)-a		155	1	1	1	1	PBR, PC	OpCode	1
	(ASL, ROL, LSR, ROR		2.	i	i	ō	i	PBR, PC+1	AAL	î
	DEC, INC, TSB, TRB)		3.	1	1	0		PBR, PC+2	AAH	1
	(6 OpCodes)		4.	1	0	1	0	THE RESERVE OF THE PARTY OF THE	Byte 0	1
	(3 bytes)	(1)	4a.	1	0	1	0	DBR, AA+-3		1
	(6 for 8-bit data,	(3)	5.	1	0	0	0	DBR, AA+1 or 3	10	1
	8 for 16-bit data,	(1)	6a.	1	0	1			Bytes 3-1	0
	12 for 32-bit data)		6.	1	0	1		DBR, AA	Byte 0	0
2c.	Absolute (JUMP) -a	+34	1.	1	1	1		PBR, PC	OpCode	1
	(JMP) (4C)		2.	1	1	0		PBR, PC+1	New PCL	1
	(1 OpCode)		3.	1	1	0		PBR, PC+2	New PCH	1
	(3 bytes)		1.	1	1	1	1	PBR, NEW PC	New OpCode	1
24	(3 cycles)		100					000 00		
20.	Absolute (Jump to subroutine) -a	108	350	i		0		PBR, PC	OpCode	1
	(JSR)		3.		-	0		PBR, PC+1 PBR, PC+2		1
	(1 OpCode)			î	1	0		PBR, PC+2		1
	(3 bytes)		5.		i	1		0,5		Ô
	(6 cycles)				i			0,5-1		0
	(different order from N6502)			1	1	1		PBR, NEW PC	New OpCode	1
+3a.	Absolute Long-al		1.	1	1	1	1	PBR, PC	OpCode	1
	(ORA, AND, EOR, ADC		2.	1	1	0		PBR, PC+1	AAL	1
	STA, LDA, CMP, SBC)	22	3.	1	1	0		PBR, PC+2		1
	(8 OpCodes)		4.	1	1	0		PBR, PC+3	AAB	1
	(4 bytes)		5.	1	1	1		AAB, AA	Byte 0 1/	0
	(5, 6 and 8 cycles)	(1)	5a.	1	1	1		AAB, AA+1	Bytes1-3 1/	0
*3b.		1	1.	1	1			PBR, PC	OpCode	1
	(JMP)		2.	1	1	0		PBR, PC+1		1
	(1 OpCode)		3.	1	1	0		PBR, PC+2		1
	(4 bytes)		4.	1	1	0		PBR, PC+3		1
	(4 cycles)	1	1.	1	1	1	1	NEW PBR, PC	OpCode	1

MARCH 1990 34

	ADDRESS MODE	CYCLE	VI	, ML	, VDA,	VPA	ADDRESS BUS	DATA BUS	R/W
*3c.	Absolute Long (JUMP to	1.	1	1	1	1	PBR, PC	OpCode	1
	Subroutine Long) -al	2.	1	1	0	1	PBR, PC+1	New PCL	1
	(JSL)	3.	1	1	0	1	PBR, PC+2	New PCH	1
	(1 OpCode)	4.	1	1	1	0	0,8	PBR	0
	(4 bytes)	5.	1	1	0	0	0,5	IO	1
	(7 cycles)	6.	1	1	0	1	PBR, PC+3	New PBR	1
		7.	1	1	1	0	0, S-1	PCH	0
		8.	1	1	1	0	0, S-2	PCL	0
		1.	1	1	1	1	NEW PBR, PC	New OpCode	1
4a.	Direct-d	1.	1	1	1	1	PBR, PC	OpCode	1
	(BIT, STZ, STY, LDY,	2.	1	1	0	1	PBR, PC+1	DO	1
	CPY, CPX, STX, LDX, (2		1	1	0	0	PBR, PC+1	IO	1
	ORA, AND, EOR, ADC,	3.	1	1	1	0	0, D+D0		/0
	STA, LDA, CMP, SBC) (1	200 H200 H20	1	1	1	0	0, D+DO+1-3	Bytes1-3 1	
	(18 OpCodes)	A. P. C.	-3					SE CONTRACT	100
	(2 bytes)								
	(3, 4, 5 and 7 cycles)								
4b.	Direct (R-M-W)-d	1.	1	1	1	1	PBR, PC	OpCode	1
	(ASL, ROL, LSR, ROR	2.	1	1	0	1	PBR, PC+1	DO	1
	DEC, INC, TSB, TRB) (2		1	1	0	0	PBR, PC+1	10	1
	(6 OpCodes)	3.	1	0	1	0	0,D+D0	Byte 0	1
	(2 bytes) (1	ALC: U.S. Color	1	0	1	0	0,D+D0+1-3	Bytes 1-3	1
	(5, 6, 7, 8, 11 and (3	CALL STREET,	1	0	0	0	0,D+D0+1 or 3	10	1
	12 cycles) (1		1	0	1	0	0,D+D0+3-1	Bytes 3-1	0
	10 -10 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	5.	1	0	ī	0	0,D+D0	Byte 0	0
5.	Accumulator-A	1.	1	1	ī	1	PBR, PC	OpCode	1
	(ASL, INC, ROL, DEC, LSR, ROR) 2.	1	1	0	0	PBR, PC+1	IO	ī
	(6 OpCodes)	4,440				V.			
	(1 byte)								
	(2 cycles)	3.635							512
6a.	Implied i	1.	1	1	1	1	PBR, PC	OpCode	1
	(DEY, INY, INX, DEX, NOP,	2.	1	1	0	0	PBR, PC+1	IO	1
	XCE, TYA, TAY, TXA, TXS,	9.3种有						E-discreto	185
	TAX, TSX, TCS, TSC, TCD,					1.0			
	TDC, TXY, TYX, CLC, SEC,					43		Supply	Mark.
	CLI, SEI, CLV, CLD, SED)	17.0			ACT TO	14			R ME
	(25 OpCodes)	1-8-8	314		2	- 4			
	(1 byte)	黄、另称"	. 4		F		10 E 10 MOSE (MATERIAL PROPERTY.	
	(2 cycles)								
*6b.	Implied i	9.488	1	1	1	1	PBR, PC	OpCode	1
J.	(XBA)	2	1	ī	ō	ō	PBR, PC+1	10	1
	(1 OpCode)	3	1	1	o	0	PBR, PC+1	IO	ī
	(1 byte)	\$, 800			5 4	1		THE RESERVE	
	(3 cycles)	4.344		3	1 1 1			1 1 1 1 1 1	
#6c.	Wait-for-Interrupt	A BAA			1		EDY YOU	OVE E DEA 3	1000
	(WAI)	9 999	1	1	1	1	1 PBR, PC	OpCode	1
	(1 OpCode) (9)	2	i	ī		ō,	1 PBR, PC+1	IO	1
	(1 byte)	2.	i	ī	0 0 1	ŏ	0 PBR, PC+1	10	1
	(3 cycles) IRQ, NMI	1.	1	1	1 1	1	1 PBR, PC+1	IRQ (BRK)	î
	(2 cycles) Ing, Net	3 Nak	*	130			I IDM/ICTI	TUT (DUTY)	27 41

MARCH 1990 35

	ADDRESS MODE	(CYCLE	VE	, ML	, VDA,	VPA	ADDRESS BUS	DATA BUS	R/W
16d.	Stop-the-Clock	10	1 1865			1		RDY	Yabet	VIDBO
	(STP)	AU	1.	1	1	1	1	1 PBR, PC	OpCode	1
	(1 OpCode)		2.	1	1	0	0	1 PBR, PC+1	IO	1
	(1 byte) RES-=1		3.	1	1	0	0	1 PBR, PC+1	10	1
	(3 cycles) RES-=0		lc.	1	1	0	0	1 PBR, PC+1	RES (BRK)	1
	RES-=0		1b.		1	0	0	1 PBR, PC+1	RES (BRK)	1
	RES-=1		la.		1	0	0	1 PBR, PC+1	RES (BRK)	1
	(See 21a. Stack		1.	1	1	1	1	1 PBR, PC+1	BEGIN	1
	Hardware Interrupt)		2,7445						ADDITION OF THE	
7.	Direct Indirect	N, HA	A 4180	1	1	1	1	PBR, PC	OpCode	1
62	Indexed-(d),y	Atta	2.	ī	1	0	1	PBR, PC+1	DO	1
	(ORA, AND, EOR, ADC,	121	2a.	1	1	0	0	PBR, PC+1	10	1
	STA, LDA, CMP, SBC)	121	3.	1	i	1	0	0, D+D0	AAL	
	(8 OpCodes)		4.	i	i	1	0	0,D+DO+1	AAH	The Party
	(2 bytes)	(4)		i	î	ō	o.	DBR, AAJ, AAL+YL	IO	
	(5, 6, 7, 8, 9 and 10	1.	5.	1	1	1	0	DBR, AA+Y	Byte 0	1/0
	cycles)	(1)		i	i	i	0	DBR, AA+Y+1-3	Bytes1-3	
	Direct Indirect	17	1.	1	i	î	1	PBR, PC	OpCode	1
Q.	Indexed Long-[d],y	100	2.	1	i	ō	1	PBR, PC+1	DO	DE E
	(ORA, AND, EOR, ADC,	(2)		1	i	0	0	PBR, PC+1	IO	1
	STA, LDA, CMP, SBC)	121	3.	i	1	1	0	0, D+DO	AAL	1
		To A	4	1	i	i	0	0,D+DO+1	AAH	1
	(8 OpCodes)	45.0	5.	1	i	i	0	0,D+DO+2	AAB	300
	(2 bytes)	(17)		i	i	ō	0	0,0+DO+2	IO	场长,对
	(6,7,8,9 and 10	17.11	6.	+	1	1	0			1/0
	cycles)	171	1000	1				AAB, AA+Y	Byte 0	
ĸ		(1)		1	1	1	0	AAB, AA+Y+1-3	Bytes1-3	110
• 3	Direct Indexed		1.	+	1	0		PBR, PC	OpCode	1
	Indirect-(d,x)		2.	+			1	PBR, PC+1	DO	
	(ORA, AND, EOR, ADC,	(2)		1	1	0	0	PBR, PC+1	10	4
	STA, LDA, CMP, SBC)	148	3.	1	1	0	0	PBR, PC+1	10	39.50
	(8 OpCodes)	Sali	4.	+	1	1	0	0,D+DO+X	AAL	28 48
	(2 bytes)	. 44	5.	1	1	1 1	0	O, D+DO+X+1	AAH	+ /0
	(6,7,8,9 and 10 cycles		6.	+	MILLS 35.54		0	DBR, AA	Byte 0	1/0
	A STATE OF THE STA	(1)		1	1	1	0	DBR, AA+1-3	Bytes1-3	1/0
a.	Direct, X-d, x		1.	1	1	1	1	PBR, PC	OpCode	11.00
	(BIT, STZ, STY, LDY,		2.	1	1	0	1	PBR, PC+1	DO	Market
	ORA, AND, EOR, ADC,	(2)			1	0 0 1	0	PBR, PC+1	10	12
	STA, LDA, CMP, SBC)	iria?	3.	1	1	0	0	PBR, PC+1	10	. 1
	(11 OpCodes)		4.	1	1		0	0,D+DO+X	Byte 0	1/0
	(2 bytes)		4a.	1	1	1	0	0,D+DO+X+1	Bytes1-3	1/0
m	(4,5,6,7 and 8 cycles)					Se in			Laterage 2 of	wills.
	Direct, X(R-M-W)-d, x	S Fair	1.	1	1	1	1	PBR, PC	OpCode	
	(ASL, ROL, LSR, ROR,	Ser	2.	1	1	0	1	PBR, PC+1	DO	doct.
	DEC, INC)	(2)		1	1	0	0	PBR, PC+1	IO	1
	(6 OpCodes)		3.	1	1	0	0	PBR, PC+1	IO	1
	(2 bytes)		4.	1	0	1	0	0, D+DO+X	Byte 0	1
	(6,7,8,9,12 and		4a.	1	0	1	0	0, D+DO+X+1	Bytes1-3	1
	13 cycles)	(3)	5.	1	0	0	0	0, D+D0+X+1	10	1
			6a.	1	0	1	0	0, D+DO+X+1	Bytes 3-1	. 0
	THE RESIDENCE OF STREET	P	6.	1	0	1	0	0, D+DO+X	Byte 0	0

36 MARCH 1990

ADDRESS MODE	C	YCLE	VP	, ML	, VDA, V	/PA	ADDRESS BUS	DATA BUS R/W
11. Direct, Y-d, y		1.	1	1	1	1	PBR, PC	OpCode 1
(STX, LDX)		2.	1	1	0	1	PBR, PC+1	DO 1
(2 OpCodes)	(2)		1	1	0	0	PBR, PC+1	10 1
(2 bytes)	34,348	3.	1	1	0	0	PBR, PC+1	10 1
(4,5,6,7 and 8 cycles)		4.	1	1	1	0	0, D+DO+Y	Byte 0 1/0
(4,5,6), and a clearer,	(1)	4a.	1	1	1	0	0, D+DO+Y+1-3	Bytes1-3 1/0
12a.Absolute, X-a, x	77.73	1.	ī	1	91	1	PBR, PC	OpCode 1
(BIT, LDY, STZ,		2.	1	1	0	1	PBR, PC+1	AAL 1
ORA, AND, EOR, ADC,		3.	i	i	0	1	PBR, PC+2	AAR 1
STA, LDA, CMP, SBC)	(4)	3a.	ī	1	0	0	DBR, AAH, AAL+XL	
(11 OpCodes)		4.	î	1	1	0	DBR, AA+X	Byte 0 1/0
	(1)	4a.	î	1	1	0	DBR, AA+X+1-3	Bytes1-3 1/0
(3 bytes) (4,5,6, 7 and 8 cycles		3,653	•				Daily less in	21.000
12b. Absolute, X (R-M-W) -a, x	' G	1.	1	1	1	1	PBR, PC	OpCode 1
(ASL, ROL, LSR, ROR,	2:0	2.	î	1	0	1	PBR, PC+1	AAL 1
DEC, INC)		3.	i	i	0	1	PBR, PC+2	AAE 1
		4.	i	î	0	0	DBR, AAH, AAL+XL	
(6 OpCodes)		5.	i	0	1	0	DBR, AA+X	Byte 0 1
(3 bytes)	773	5a.	i	0	î	0	DBR, AA+X+1-3	Bytes 1-3 1
(7,9 and 13 cycles)	(1)	6.	i	0	Ô	0	DBR, AA+X+lor3	10 1
			1	0		0	THE RESIDENCE OF THE PARTY OF T	Bytes 3-1 0
	(1)			0	1	0	DBR, AA+X+3-1	Byte 0 0
and the late Year Waller		7.	1	1	1		DBR, AA+X	OpCode 1
*13. Absolute Long, X-al, x		1.			0	:	PBR, PC	AAL 1
(ORA, AND, EOR, ADC,	310	2.	1	1	0	*	PBR, PC+1	AAH 1
STA, LDA, CMP, SBC)		3.		1		:	PBR, PC+2	
(8 OpCodes)	1274	4.	1	1	0		PBR, PC+3	AAB 1 IO 1
(4 bytes)	(17)	4a.	1		1	0	PBR, PC+3	
(5,6,7 and 8 cycles)	14.2	5.	1	1		0	AAB, AA+X	Byte 0 1/0
	(1)	5a.	1	1		0	AAB, AA+X+1-3	Bytes1-3 1/0
14. Absolute, Y-a, y		1.	1	1	-		PBR, PC	OpCode 1
(LDX, ORA, AND, EOR, ADC,	RACK	2.	1	1	0	1	PBR, PC+1	AAL 1
STA, LDA, CMP, SBC)	48.4/5	3.	1	1		1	PBR, PC+2	AAH 1
(9 OpCodes)	(4)	3a.	1	1	0	0	DBR, AAH, AAL+Y	10 1
(3 bytes)	- 124	4.	1	1	1	0	DBR, AA+Y	Byte 0 1/0
(4,5,6,7 and 8 cycles)	(1)		1	1	1	0	DBR, AA+Y+1-3	Bytes1-3 1/0
15. Relative-r	330	1.	1	1	1	1	PBR, PC	OpCode 1
(BPL, BMI, BVC, BVS, BCC,	1.40	2.	1	1	0	1	PBR, PC+1	OFF 1
BCS, BNE, BEQ, BRA)	(5)		1	1	0	0	PBR, PC+1	10 1
(9 OpCodes)	(6)	2b.		1	0	0	PBR, PC+1	10 1
(2 bytes)		1.	1	1	1	1	PBR, PC+Offset	OpCode 1
(2,3 and 4 cycles)				519	ALT: LINE		and the last on	m to bear the 2
*16. Relative Long-rl		1.	1	1	1	1	PBR, PC	OpCode 1
(BRL)		2.	1	1	0	1	PBR, PC+1	OFF Low 1
(1 OpCode)	145	3.	1	1	0	1	PBR, PC+2	OFF High 1
(3 bytes)		4.	1	1	0	0	PBR, PC+2	10 1
(4 cycles)		1.	1	1	1	1	PBR, PC+Offset	OpCode 1
17a.Absolute Indirect-(a)		1.	1	1	1	1	PBR, PC	OpCode 1
(JMP)		2.	1	1	0	1	PBR, PC+1	AAL 1
(1 OpCode)		3.	1	1	.0	1	PBR, PC+2	AAH 1
(3 bytes)	- A-116		1	1	1	0	O, AA	New PCL 1 New PCH 1
(5 cycles)		5.	1	1	1	0	0, AA+1	
		1.	1	1	1	1	PBR, NEW PC	OpCode 1

37

ADDRESS MODE	CYCLE	V2	, ML	, VDA,	VPA	ADDRESS BUS	DATA BUS R/W	
*17b.Absolute Indirect-(a)	1.	1	1	1	1	PBR, PC	OpCode 1	
(JML)	2.	1	1	0	1	PBR, PC+1	AAL 1	
(1 OpCode)	3.	1	1	0	1	PBR, PC+2	AAH 1	
(3 bytes)	4.	1	1	1	0	O, AA	New PCL 1	
(6 cycles	5.	1	1	1	0	0, AA+1	New PCH 1	
0.00	6.	1	1	1	0	0, AA+2	New PBR 1	
	1.	1	1	1	1	NEW PBR, PC	OpCode 1	
#18. Direct Indirect-(d)	1.	1	1	1	1	PBR, PC	OpCode 1	
(ORA, AND, EOR, ADC,	2.	1	1	0	1	PBR, PC+1	DO 1	
STA, LDA, CMP, SBC) (2	MDF 20001	1	1	0	0	PBR, PC+1	10 1	
(8 OpCodes)	3.	1	1	1	0	0, D+DO	AAL 1	
(2 bytes)	4.	1	1	Tig.	0	0, D+DO+1	AAR 1	
(5,6,7,8 and 9 cycles)	5.	1	15	1	0	DBR, AA	Byte 0 1/0	
(3, 6, 7, 6 and 3 Cycles)		î	0	1	0	PBR, AA+1-3	Bytes1-3 1/0	
*19. Direct Indirect Long-[d]	1.	1	07	1	1	PBR, PC	OpCode 1	
	2.	4	6	0	i	PBR, PC+1	DO 1	
(ORA, AND, EOR, ADC	THE PERSON NAMED IN		1	0	0	PBR, PC+1	10 1	
STA, LDA, CMP, SBC (2		:	-	1	0		AAL 1	
(8 OpCodes)	3.		100			0,D+D0	AAH 1	
(2 bytes)	4.		1	1	0	0,D+DD+1		
(6,7,8,9 and 10 cycles)	5.		1	1	0	0,D+DO+2	AAB 1	
	6.	1	-	1	0	AAB, AA	Byte 0 1/0	
(1		1	11	1	0	AAB, AA+1-3	Bytes1-3 1/0	
20a.Absolute Indexed Indirect	10.00		3.5	1			A-A-4-	
(a, x)	1.	-	1	1	1	PBR, PC	OpCode 1	
(JMP)	2.	-	1	0	1	PBR-PC+1	AAL 1	
(1 OpCode)	3.	1	1	0	1	PBR-PC+2	AAH 1	ĸ
(3 bytes)	4.	1	1	0	0	PBR, PC+2	10 1	
(6 cycles)	5.	1	1	0	1	PBR, AA+X	New PCL 1	
	6.	1	32	0	1	PBR, AA+X+1	New PCH 1	
	1.	1	1	1	1	PBR, NEW PC	OpCode 1	
*20b.Absolute Indexed Indirect		1	1	1	1	PBR, PC	OpCode 1	
(a,x)	2.	1	1	0	1	PBR, PC+1	AAL 1	
(JSR)	3.	1	1	1	0	0,5	PCH 0	
(1 OpCode)	4.	1	1	1	0	0,S-1	PCL 0	
(3 bytes)	5.	1	1	0	1	PBR, PC+2	AAH 1	
(8 cycles)	6.	1	1	0	0	PBR, PC+2	10 1	
경영 가 그렇게 내내가 없었는데 그 모양이		1	1	0	1	PBR, AA+X	New PCL 1	
집에 그 아이트리 아빠워선 그 나를 맛	8.	1	1	0	1	PBR, AA+X+1	New PCH 1	
	1.	1	1	1	1	PBR, NEW PC	New OpCode 1	
21a. Stack (Hardware	1.	1	1	1	1	PBR, PC	10 1	
Interrupts) -s (3)	2.	1	1	0	0	PBR, PC	10 1	
(IRQ, NMI, ABORT, RES) (7)		1	1	1	0	0,5	PBR 0	
(4 hardware interrupts) (1		1	1	1	0	0,5-1	PCH 0	
	5.	1	1	1	0	0,8-2	PCL 0	
(7 and 8 cycles) (10) (11)		1	1	1	0	0,5-3	P 0	
	7.	0	1	1	0	O, VA	AAVL 1	
Market State of State	8.	0	1	1	0	0, VA+1	AAVH 1	
	1.	1	1	1	1	O, AAV	New OpCode 1	
	11.00	CO.	100	N. T. W.	die		CO. DOMESTICAL AND ADDRESS OF REAL PROPERTY.	

MARCH 1990

39

ADDRESS MODE C	YCLE	VI	, ML	VDA,	VPA	ADDRESS BUS	DATA BUS	R/₩
21b.Stack (Software (16)	1.	1	1	1	1	PBR, PC	OpCode	1.08 81
Interrupts) -s (16) (3)		1	1	0	1	PBR, PC+1	Signature	1
	3.	1	1	1	0	0,5	PBR	0
(2 OpCodes)	4.	1	.1		0	0, S-1	PCH	0
(2 bytes)	5.	1	1	1 1	0	0, S-2	PCL	0
(7 and 8 cycles)	6.	1	1	1	0	0,5-3 (16)	P	0
	7.	0	1	1	0	O, VA	AAVL	1
AND DESCRIPTION OF THE PARTY OF	8.	0	1	1	0	0, VA+1	AAVH	1
### ### ### ##########################	1.	1	1	1	1	O, AAV	New OpCode	ī
21c.Stack (Return from		1	1	1	1	PBR, PC	OpCode	1
Interrupt) -s	2.	ī	i	1 0	ō	PBR, PC+1	IO	1
(RTI) (3)		ī	1	0	0	PBR, PC+1	IO	1
(1 Op Code)	4.	ī	200	1	0	0, S+1	P	i
(1 byte)	5.	î	1	ī	0	0, S+2	PCL	-
(6 and 7 cycles)	6.	i	î	i	0	0, 5+3	PCH	•
(different order from (7)	7.	i	i	i	0	0, 5+4	PBR	
		+	ī	+	1	CONTRACTOR OF THE PROPERTY OF	11-140-400 F3/1-200 M/10C/1-1-0/11 W	
N6502)	1.	+	1	1 1 0	1	PBR, PC	New OpCode	
21d.Stack (Return from	1.	+	1	7		PBR, PC	OpCode	3 10
Subroutine) -s	2.		1	0	0	PBR, PC+1	10 (90)	
(RTS)	3.	:		0	0	PBR, PC+1	IO BANGE	* S
(1 Op Code)	4.	+	1	1 0 1	0	0, S+1	PCL	
(1 byte)	5.	:	1	7	0	0, S+2	PCH	
(6 cycles)	6.	1	1	,	0	NEW PC-1	10	ACRON AGE
400 OL IN INC. 10 IN I	1.	1	1		1	PBR, PC	OpCode	PROPERTY SON
*21e.Stack (Return from	1.	1		1	1	PBR, PC	OpCode	Hattor
Subroutine Long) -s	2.	+	1	0	0	PBR, PC+1	10	
(RTL)	3.	1	1	0	0	PBR, PC+1	10	
(1 Op Code)	4.	1	1	1	0	0, S+1	New PCL	
(1 byte)	5.	1	1	1	0	0, S+2	New PCH	1 000
(6 cycles)	6.	1	1	1	0	0, S+3	New PBR	
	1.	1	1	1	1	NEW PBR, PC	New OpCode	Postal State
21f.Stack (Push)-s	1.	1	1	1	1	PBR/PC	OpCode	1 Sales
(PHP, PHA, PHY, PHX,	2.	1	1	0	0	PBR, PC+1	10	1
PHD, PHK, PHB) (1) (11)		1	1	1	0	0,5	Bytes3-1	1
(7 Op Codes)	3.	1	1	1	0	0, 5-1-3	Byte0	1
(1 byte)	178	1:	0	100			1681	15 (22)
(3,4, and 6 cycles)	a de		4	6				
21g.Stack (Pull)-s	1.	1	1	1		PBR, PC	OpCode	1
(PLP, PLA, PLY, PLX, PLD, PLB)	2.	1	1	0	0	PBR, PC+1	IO	1
(Different than N6502)	3.	1	1	0	0	PBR, PC+1	IO TOTAL	1
(6 Op Codes)	4.	1	1	1		0, S+1	Byte 0	1
(1 byte) (1)	4a.	1	1	1	0	0,5+2-4	Bytes1-3	1000
(4,5 and 7 cycles)	10.2					File State of State o		
*21h.Stack (Push Effective	1.	1	1	1	1	PBR, PC	OpCode	1
Indirect Address)-s	2.	1	1	0	1	PBR, PC+1	DO	1
(PEI) (2)	2a.	1	1	0	0	PBR, PC+1	IO	THE RESERVE SHARE AND ADMINISTRATION OF THE PARTY OF THE
(1 Op Code)	3.	1	1	1	0	0, D+D0	AAL	1
(2 bytes)	4.	1	1	1	0	0, D+D0+1	AAH	1
(6 and 7 cycles)	5.	1	1	1	0	0, S-1	AAH	0
	6.	1	1	1	0	0, S-1	AAL	0
	W. T. T. S. L.			X4235151637	00000	STREET, STREET		A CANADA LINE CONTRACTOR

MARCH 1990

HE

54

31

E A bbš.

XIII QHS.

op# MET.

ADDRESS MODE	CYCLE	VE	, MI,	VDA,	VPA	ADDRESS BUS	DATA BUS	R/W
*21i.Stack (Push Effective	1.	1	1	1	1	PBR, PC	OpCode	1
Absolute Address)-s	2.	1	1	0	1	PBR, PC+1	AAL	1
(PEA)	3.	1	1	0	1	PBR, PC+2	AAH	1
(1 Op Code)	4.	1	1	1	0	0,5	AAH	0
(3 bytes)	5.	1	1	1	0	0,5-1	AAL	0
(5 cycles)	1.7681							HEED!
*21j.Stack (Push Effective	1.	1	1	1	1	PBR, PC	OpCode	1
Program Counter Relative			1	0	1	PBR, PC+1	OFF LOW	1
Address) -s	3.	1	1	0	1	PBR, PC+2	OFF High	1
(PER)	4.	1	1	0	0	PBR, PC+2	10	1
(1 Op Code)	5.	1	1	1	0	0,8	PCH+OFF+	0
(3 bytes)	X, Lill	jīš:		ATT CO		11 11 11 11	Carry	Par I
(6 cycles)	6.	1	1	1	0	0,8-1	PCL+OFF	0
*22. Stack Relative-d,s		1	1	1	1	PBR, PC	OpCode	1
(ORA, AND, EOR, ADL,	2.	1	1	0	1	PBR, PC+1	SO	1
STA, LDA, CMP, SBC)	3.		1	0	0	PBR, PC+1	10	1
(8 Op Codes)	4.		1	1	0	0, \$+\$0		1/0
(2 bytes) (1	1001 26100		1	1	0	0, \$+\$0+1-3	Bytes1-3	
(4,5 and 7 cycles)	200			1		CARL THEFE	The second	
*23. Stack Relative Indirect	1.	1	1	1	1	PBR, PC	OpCode	1
Indexed-(d, s), y	2.	i		0	1	PBR, PC+1	SO	- 1
(ORA, AND, EOR, ADC, STA, LDA,		1	1	0	0	PBR+PC+1	10	1
CMP, SBC)	4.	1 1 1	î	1	0	0,5+50	AAL	1
(8 Op Codes)	5	1	1	1	0	0,5+50+1	AAH	1
	6.	i	i	1 0	0	0, 5+50+1	10	1
(2 bytes) (7,8 and 10 cycles)	7.	1	1	1	Ö	DBR, AA+Y	AND DESCRIPTION OF THE PERSON	1/0
(1, s and 10 cycles)	CALL TO LEAD TO SERVICE	î	1	i	0	DBR, AA+Y+1-3	Bytes1-3	
*24a.Block Move Positive	171.	1	1	1	1	PBR, PC	OpCode	1
(forward)-xyc (18)	and the same of the same of	1	1	0	1	PBR, PC+1	DBA	1
(MVP) (18)	400	i	1	0	1	PBR, PC+2	SBA	1
(1 Op Code) N-2		1	10	1	0	SBA,X	SRC Data	1
(3 bytes) Byte	7	1	1	i	0	DBA, Y	DEST Data	0
(5 and 7 cycles) C=2		1	1	0	0	DBA, Y	10	1
to ame , cleans,	1 7.	1	1	0	0	DBA, Y	10	1
x=Source Address	in the	000	Pla				COLUMN TO SELECT	9057
y=Destination	171	1	1	1	1	PBR, PC	OpCode	1
c=#of bytes to move-1(18)	2.	1	1	0	1	PBR, PC+1	DBA	1
x,y Decrement (18)		1	1	0	1	PBR, PC+2	SBA	i
MVP is used when the N-1		i	i	1	Ô	SBA, X-1	SRC Data	1
dest. start address Byte		î	ī	1	0	DBA, Y-1	DEST Data	Ô
	1 6.	î	ī	ō	0	DBA, Y-1	10	994
positive) than the source	The same of the sa	î	î	0	0	DBA, Y-1	10	1
start address.		20	£398	dieta	60	milio bilisy jo	a Man Alexan	
Start address.	11.	4		104	4	PBR, PC	Op Code	1
FFFFFF (18)		1	1	0	70	PBR, PC+1	DBA	1
^ Dest Start (18)		;	1	o	1	PBR, PC+2	SBA	1
THE RESERVE OF THE PROPERTY OF		1	1	1	ô	SBA, X-2	SRC Data	88 2
N Byte		1	1	1	0	DBA, Y-2	DEST Data	n
		i	1	0	0	DBA, Y-2	IO Data	1
	7.	1	-	0	0	DBA, Y-2	IO	1
Source End		*	150	1		PBR, PC+3		0 1
000000	1_1.	+				EBRIEGTS	New OpCod	

40

ADDRESS MODE	CYCLE	VP	, ML	, VDA,	/PA	ADDRESS BUS	DATA BUS	R/W
*24b.Block Move Negative	171.	1	1	1	1	PBR, PC	OpCode	1
(backward) -xyc	(18) 2.	1	1	0 .	1	PBR, PC+1	DBA	1
(MVN)	(18) 3.	1	1	. 0	1	PBR, PC+2	SBA	1
(1 Op Code)	N-2 4.	1	1	1	0	SBA, X	SRC Data	1
(3 bytes)	Bytel 5.	1	1	1 0	0	DBA, Y	DEST Data	0
(7 cycles)	C-21 6.	1	1	0	0	DBA, Y	10	1
	17.	1	1	- 0 -	0	DBA, Y	10	1
x=Source Address	-		5 200					200
y=Destination	11.	1	. 1	1	1	PBR, PC	OpCode	1
c=#of bytes to move-1	(18) 2.	1	1	0	1	PBR, PC+1	DBA	1
x, y Increment	(18) 3.	1	1	0	1	PBR, PC+2	SBA	1
MVN is used when the	N-1 4.	1	1	1	0	SBA, X+1	SRC Data	1
dest. start address	Bytel 5.	1	1	1 ,	0	DBA, Y+1	DEST Data	0
is lower (more	C=1 6.	1	1	0	0	DBA, Y+1	IO	1
negative) than the so	purcel 7.	1	1	.0	0	DBA, Y+1	10	1
start address.								
	11.	1	1	1	1	PBR, PC	OpCode	1
FFFFF	(18) 2.	1	1	0	1	PBR, PC+1	DBA	1
Source End	(18) 3.	1	1	0	1	PBR, PC+2	SBA	1
1 1 N	Bytel 4.	1	1	1	0	SBA, X+2	SRC Data	1
_ _Dest.End	C=01 5.	1	1	1	0	DBA, Y+2	DEST Data	0
Source Start	1 6.	1	1	0	0	DBA, Y+2	10	1
V Dest.Start	1.7.	1	1	0	0	DBA, Y+2	10	1
000000	1_1.	1	1	1	1	PBR, PC+3	New OpCode	1

- Add 1 byte (for immediate only) for 16-bit data, add 3 bytes for 32-bit data, add 1 cycle for 16-bit data and 3 cycles for 32-bit data.
- 2. Add 1 cycle for direct register low (DL) not equal 0.
- Special case for aborting instruction. This is the last cycle which may be aborted or the Status, PBR or DBR registers will be updated.
- Add 1 cycle for indexing across page boundaries, or write, or 16-bit or 32-bit Index Registers. When 8-bit Index Registers or in the emulation mode, this cycle contains invalid addresses.
- Add 1 cycle if branch is taken.
- 6. Add 1 cycle if branch is taken across page boundaries in 6502 emulation mode.
- Subtract 1 cycle for 6502 emulation mode.
- 8. Add 1 cycle for REP, SEP.
- 9. Wait at cycle 2 for 2 cycles after NMI- or IRQ- active input.
- 10. R/W- remains high during Reset.
- 11. BRK bit 4 equals "0" in Emulation mode.
- 12. PHP and PLP.
- Some OpCodes shown are not on the W65C02.
- 14. VDA and VPA are not valid outputs on the W65C02 but are valid on the W65C832. The two signals, VDA and VPA, are included to point out the upward compatibility to the W65C832. When VDA and VPA are both a one level, this is equivalent to SYNC being a one level.
- 15. The PBR is not on the W65C02.
- Co-processors may monitor the signature byte to aid in processor to co-processor communications.
- 17. Add 1 cycle for 32-bit Index Register mode.
- 18. Subtract 2 bytes and 2 cycles when in W65C832 Native mode for MVN and MVP.

I IT IN ANY BOTH BOTH AT THE

DO LINE

lates committee fracts county

A BIGREY (GOVE)

AAB	Absolute Address Bank	OFF	Off
AAH	Absolute Address High	P	Sta
AAL	Absolute Address Low	PBR	Pro
AAVH	Absolute Address Vector High	PC	Pro
AAVL		PCH	Pro
Byte 0	Data Byte 0	PCL	Pro
Bytes 1-3	Data Bytes 1-3	R-M-W	Rea
C	Accumulator	S	Sta
D	Direct Register	SBA	Sou
DBA	Destination Bank Address	SRC	Sou
DBR	Data Bank Register	SO	Sta
DEST	Destination	VA	Vec
DO	Direct Offset	x, y	Ind
ID0	Immediate Data Byte 0		
	Immediate Data Bytes 1-3		
10	Internal Operation		

OFF Offset
P Status Register
PBR Program Bank Register
PC Program Counter
PCH Program Counter High
PCL Program Counter Low
R-M-W Read-Modify-Write
S Stack Address
SBA Source Bank Address
SRC Source
SO Stack Offset
VA Vector Address
x,y Index Register

^{* =} New W65C816/802 Addressing Modes

^{# =} New W65C02 Addressing Modes Blank = NMOS 6502 Addressing Modes

SECTION 7

RECOMMENDED ASSEMBLER SYNTAX STANDARDS

7.1 Directives

Assembler directives are those parts of the assembly language source program which give directions to the assembler; this includes the definition of data area and constants within a program. This standard excludes any definitions of assembler directives.

7.2 Comments

An assembler should provide a way to use any line of the source program as a comment. The recommended way of doing this is to treat any blank line, or any line that starts with a semi-colon or an asterisk as a comment. Other special characters may be used as well.

7.3 The Source Line

Any line which causes the generation of a single machine language instruction should be divided into four fields: a label field, the operation code, the operand, the comment field.

- 7.3.1 The Label Field--The label field begins in column one of the line. A label must start with an alphabetic character, and may be followed by zero or more alphanumeric characters. An assembler may define an upper limit on the number of characters that can be in a label, so long as that upper limit is greater than or equal to six characters. An assembler may limit the alphabetic characters to upper-case characters if desired. If lower-case characters are allowed, they should be treated as identical to their upper-case equivalents. Other characters may be allowed in the label, so long as their use does not conflict with the coding of operand fields.
- 7.3.2 The Operation Code Field--The operation code shall consist of a three character sequence (mnemonic) from Table 6-2. It shall start no sooner than column 2 of the line, or one space after the label if a label is coded.
- 7.3.2.1 Many of the operation codes in Table 6-2 have duplicate mnemonics; when two or more machine language instruction have the same mnemonic, the assembler resolves the difference based on the operand.
- 7.3.2.2 If an assembler allows lower-case letters in labels, it must also allow lower-case letters in the mnemonic. When lower-case letters are used in the mnemonic, they shall be treated as equivalent to the upper-case counterpart. Thus, the mnemonics LDA, Ida and LdA must all be recognized, and are equivalent.
- 7.3.2.3 In addition to the mnemonics shown in Table 6-2, an assembler may provide the alternate mnemonics show in Table 7-3-1.

MARCH 1990

Are relieved to the first regularities and sell apprecious and representations

to it if the beautiful and result of the directive promised for the same

Table 7-3-1 Alternate Mnemonics

BLT BGE CMA	powiose 16 L.ft
CMA	
A STATE OF THE RESIDENCE OF THE PARTY OF THE	
DEA	
INA	Activities directions are the
	na elle og stickingstik ever
TLIN	manager a more level and the companies
TAD	
TAS	
TDA	Dispusion 2.7
TSA	
SWA	Ar asserber should prove
The second secon	JSR JMP TAD TAS TDA TSA

- 7.3.2.4 JSL should be recognized as equivalent to JSR when it is specified with a long absolute address. JML is equivalent to JMP with long addressing forced.
- 7.3.3 The Operand Field--The operand field may start no sooner than one space after the operation code field. The assembler must be capable of at least twenty-four bit address calculations. The assembler should be capable of specifying addresses as labels, integer constants, and hexadecimal constants. The assembler must allow addition and subtraction in the operand field. Labels shall be recognized by the fact that they start alphabetic characters. Decimal numbers shall be recognized as containing only the decimal digits 0...9. Hexadecimal constants shall be recognized by prefixing the constant with a "\$" character, followed by zero or more of either the decimal digits or the hexadecimal digits "A"... "F". If lower-case letters are allowed in the label field, then they shall also be allowed as hexadecimal digits.

7.3.3.1 All constants, no matter what their format, shall provide at #2 8.51 \$ 18.02 ; least enough precision to specify all values that can be represented by a twenty-four bit signed or unsigned integer represented in two's complement notation.

7.3.3.2 Table 7-3-2 shows the operand formats which shall be recognized by the assembler. The symbol d is a label or value which the assembler Sendos on 3 can recognize as being less than \$100. The symbol a is a label or value which the assembler can recognize as greater than \$FF but less than \$10000; the symbol al is a label or value that the assembler can recognize as being greater than \$FFF. The symbol EXT is a label which cannot be located by the assembler at the time the instruction is assembled. Unless instructed otherwise, an assembler shall assume that EXT labels are two bytes long. The symbols r and rl are 8 and 16 bit signed displacements calculated by the assembler.

7.3.3.3 Note that the operand does not determine whether or not immediate address loads one or two bytes, this is determined by the setting of the status register. This forces the requirement for a directive or directives that tell the assembler to generate one or two bytes of space for immediate loads. The directives provided shall allow

separate settings for the accumulator and index registers.

7.3.3.4 The assembler shall use the <, >, and ^ characters after the f character in immediate address to specify which byte or bytes will be selected from the value of the operand. Any calculations in the operand must be performed before the byte selection takes place. Table 7-3-2 defines the action taken by each operand by showing the effect of the operator on an address. The column that shows a two byte immediate value show the bytes in the order in which they appear in memory. The coding of the operand is for an assembler which uses 32 bit address calculations, showing the way that the address should be reduced to a 24 bit value.

Table 7-3-2 Byte Selection Operator

Operand	One Byte	Result	Two	Byte	Result	Four	Byt	e Re	sult
#\$01020304 #<\$01020304	04 04			03	04 04	01	02	03	04
#>\$01020304 #^\$01020304	03 02			02	03				

7.3.3.5 In any location in an operand where an address, or expression resulting in an address, can be coded, the assembler shall recognize the prefix characters <, |, and >, which force one byte (direct page), two byte (absolute) or three byte (long absolute) addressing. In cases where the addressing modes is not forced, the assembler shall assume that the address is two bytes unless the assembler is able to determine the type of addressing required by context, in which case that addressing mode will be used. Addresses shall be truncated without error in an addressing mode is forced which does not require the entire value of the address. For example,

LDA \$0203 LDA |\$010203

are completely equivalent. If the addressing mode is not forced, and the type of addressing cannot be determined from context, the assembler shall assume that a two byte address is to be used. If an instruction does not have a short addressing mode (as in LDA< which als no direct page indexed by Y) and a short address is used in the operand, the assembler shall automatically extend the address by padding the most significant bytes with zeroes in order to extend the address to the length needed. As with immediate address, any expression evaluation shall take place before the address is selected; thus, the address selection character is only used once, before the address of expression.

7.3.3.6 The ! (exclamation point) character should be supported as an alternative to the | (vertical bar).

7.3.3.7 A long indirect address is indicated in the operand field of an instruction field of an instruction by surrounding the direct page address where the indirect address is found by square brackets; direct page addresses which contain sixteen-bit addresses are indicated by being surrounded by parentheses.

7.3.3.8 The operands of a block move instruction are specified as source bank, destination bank-the opposite order of tzz object bytes generated.

7.3.4 Comment Field--The comment field may start no sooner than one space after the operation code field or operand field depending on instruction type.

SECTION 8

CAVEATS

Table 8-1 W65C816 Compatibility Issues

		W65C816/802	W65C02	NMOS 6502
	(Stack)	bits when (E=0)	bits	bits
2. X	(X Index Reg)	Indexed page zero always in page 0 (E=1), Cross page (E=0)	1	Always page 0
3. Y	(Y Index Reg)	Indexed page zero always in page 0 (E=1), Cross page (E=0)		Always page 0
4. A	(Accumulator)	8 bits (M-1), 16 bits (M-0)	1	8 bits
	Flag Reg)	N,V, and Z flags valid in decimal mode. D=0 after reset/interrupt.	walid in dec. mode. D=0 after reset/interrupt	N,V, and 2 flags invalid in decimal mode. D=unknown after reset. D not modified after interrupt
A	iming A. ABS,X ASL, LSR, RCL, ROR With No Fage Crossing Jump Indirect Operand-XXFF	17 cycles 15 cycles	6 cycles 	7 cycles 5 cycles and invalid page
	Eranch Across	4 cycles (E-1) 13 cycles (E-0)	14 cycles	crossing 4 cycles
	RK Vector	ODFFFE,F (E=1) BRK bit=C on	FFFE,F BRK bit= 0 on stack if IRQ-, NMI	No add. cycle FFFE,F BRK bit=C on stack if IRC-, NM1
3	nterrupt or Break ank Address	PBR not pushed (E=1), RTI PBR not pulled (E=1), PBR pushed (E=C), RTI PBR pulled (E=C)	Not available - -	Not available
9. M	enory Lock (ML-)	(ML-=0 during Read Modify and Write cycles.		Not available

	W65C816/802	W65C02	NMOS 6502
<pre>10.Indexed Across Page Boundary (d),y;a,x; a,y</pre>	invalid address.		Extra read of invalid address.
11.RDY Pulled During Write Cycle	Ignored (E=1) for W65C816 only. Processor stops (E=0).	Processor stops.	Ignored.
12.WAI & STP instruct.	Available	Available	Not available
13.Unused OP Codes	One reserved OP Code specified as WDM will be used in future systems The W65C816 performs a no- operation.	loug bases and	Unknown and some "hang up" processor.
14.Bank Address Handling	PBR=00 after re- set or interrupts	SALES AND SERVICE TO SERVE AND A SERVICE AND	Not available
15.R/W- During Read- Modify-Write Instructions	E=1,R/W-=0 during Modify and Write cycles. E=0,R/W-= 0 only during Write cycle.	ing Write cycle.	
16.Pin 7		SYNC	SYNC
17.COP Instruction Signatures 00-7F defined. Signatures 80-FF reserved	Available	Not available 	Not available

8.1 Stack Addressing

When in the Native mode, the Stack may use memory locations 000000 to 00FFFFF. The effective address of Stack, Stack Relative, and Stack Relative Indirect Indexed addressing modes will always be within this range. In the Emulation mode, the Stack address range is 000100 to 0001FF. The following opcodes and addressing modes will increment or decrement beyond this range when accessing two or three bytes.

JSL; JSR(a,x); PEA, PEI, PER, PHD, PLD, RTL; d, s; (d,s), y

8.2 Direct Addressing

8.2.1 The Direct Addressing modes are often used to access memory registers and pointers. The effective address generated by Direct; Direct, X and Direct, Y addressing modes will always be in the Native mode range 000000 to 00FFFF. When in the Emulation mode, the direct addressing range is 000000 to 0000FF, except for [Direct] and [Direct], Y addressing modes and the PEI instruction which will increment from 0000FE or 0000FF into the Stack area.

has williast affew ben without

VESLIES BURES

8.2.2 When in the Emulation mode and DH is not equal to zero, the direct addressing range is 00DH00 to 00DHFF, except for [Direct] and [Direct].Y addressing modes and the PEI instruction which will increment from 00DHFE or 00DHFF into the next higher page.

8.2.3 When in the Emulation mode and DL in not equal to zero, the direct

M and to Silay will assessed by

addressing range is 000000 to 00FFFF.

8.3 Absolute Indexed Addressing

The Absolute Indexed addressing modes are used to address data outside the direct addressing range. The W65C02 and W65C832 addressing range is 0000 to FFFF. Indexing from page FFXX may result in a 00YY data fetch when using the W65C02 or W65C832. In contrast, indexing from page ZZFFXX may result in ZZ+1,00YY when using the W65C832.

8.4 ABORT- Input

8.4.1 ABORT- should be held low for a period not to exceed one cycle. Also, if ABORT- is held low during the Abort Interrupt sequence, the Abort Interrupt will be aborted. It is not recommended to abort the Abort Interrupt. The ABORT- internal latch is cleared during the second cycle of the Abort Interrupt. Asserting the ABORT- input after the following instruction cycles will cause registers to be modified:

8.4.1.1 Read-Modify-Write: Processor status modified if ABORT- is

asserted after a modify cycle.

8.4.1.2 RTI: Processor status modified if ABORT- is asserted after cycle 3.

8.4.1.3 IRQ-, NMI-, ABORT- BRK, COP: When ABORT- is asserted after cycle 2, PBR and DBR will become 00 (Emulation mode) or PBR will become 00 (Native mode).

8.4.2 The Abort- Interrupt has been designed for virtual memory systems. For this reason, asynchronous ABORT's- may cause undesirable results due to

the above conditions.

8.5 VDA and VPA Valid Memory Address Output Signals

When VDA or VPA are high and during all write cycles, the Address Bus is always valid. VDA and VPA should be used to qualify all memory cycles. Note that when VDA and VPA are both low, invalid addresses may be generated. The Page and Bank addresses could also be invalid. This will be due to low byte addition only. The cycle when only low byte addition occurs is an optional cycle for instructions which read memory when the Index Register consists of 8 bits. This optional cycle becomes a standard cycle for the Store instruction, all instructions using the 16-bit Index Register mode, and the Read-Modify-Write instruction when using 8- or 16-bit Index Register modes.

8.6 Apple II, IIe, IIc and II+ Disk Systems

VDA and VPA should not be used to qualify addresses during disk operation on Apple systems. Consult your Apple representative for hardware/software configurations.

edi da eres vel edi dan

8.7 DB/BA Operation when RDY is Pulled Low

When RDY is low, the Data Bus is held in the data transfer state (i.e., PHI2 high). The Bank address external transparent latch should be latched when the PHI2 clock or RDY is low.

ACCRECATING ESTINGS IS BORDOO to CONFESS,

8.8 M/X Output

The M/X output reflects the valid of the M and X bits of the processor Status Register. The REP, SEP and PLP instructions may change the state of the M and X bits. Note that the N/X output is invalid during the instruction cycle following REP, SEP and PLP instruction execution. This cycle is used as the opcode fetch cycle of the next instruction.

- 8.9 All Opcodes Function in All Modes of Operation
 - 8.9.1 It should be noted that all opcodes function in all modes of operation.

 However, some instructions and addressing modes are intended for W65C832
 24-bit addressing and are therefore less useful for the W65C832. The
 following is a list of instructions and addressing modes which are
 primarily intended for W65C832 use:

JSL; RTL; [d]; [d], y; JMP al; JML; al, al, x 8.9.2 The following instructions may be used with the W65C832 even though a Bank Address is not multiplexed on the Data Bus: PHK; PHB; PLB

- 8.9.3 The following instructions have "limited" use in the Emulation mode: 8.9.3.1 The REP and SEP instructions cannot modify the M and X bits when in the Emulation mode. In this mode the M and X bits will always be high (logic 1). 8.9.3.2 When in the Emulation mode, the MVP and MVN instructions use the X and Y Index Registers for the memory address. Also, the MVP and MVN instructions can only move data within the memory range 0000 (Source Bank) to 00FF (Destination Bank) for the W65C832, and 0000 to 00FF for the W65C832.
- 8.10 Indirect Jumps

The JMP (a) and JML (a) instructions use the direct Bank for indirect addressing, while JMP (a,x) and JSR (a,x) use the Program Bank for indirect address tables.

8.11 Switching Modes

When switching from the Native mode to the Emulation mode, the X and M bits of the Status Register are set high (logic 1), the high byte of the Stack is set to 01, and the high bytes of the X and Y Index Registers are set to 00. To save previous values, these bytes must always be stored before changing modes. Note that the low byte of the S, X and Y Registers and the low and high byte of the Accumulator (A and B) are not affected by a mode change.

MARCH 1990 49

Applied average the second Apple representative for handred software

- 8.12 How Hardware Interrupts, BRK, and COP Instructions Affect the Program Bank and the Data Bank Registers
- 8.12.1 When in the Native mode, the Program Bank register (PBR) is cleared to 00 when a hardware interrupt, BRK or COP is executed. In the Native mode, previous PBR contents is automatically saved on Stack.

8.12.2 In the Emulation mode, the PBR and DBR registers are cleared to 00 when a hardware interrupt, BRK or COP is executed. In this case, previous

contents of the PBR are not automatically saved.

8.12.3 Note that a Return from Interrupt (RTI) should always be executed from the same "mode" which originally generated the interrupt.

8.13 Binary Mode

The Binary Mode is set whenever a hardware or software interrupt is executed. The D flag within the Status Register is cleared to zero.

8.14 WAI Instruction

The WAI instruction pulls RDY low and places the processor in the WAI "low power" mode. NMI-, IRQ- or RESET will terminate the WAI condition and transfer control to the interrupt handler routine. Note that an ABORT- input will abort the WAI instruction, but will not restart the processor. When the Status Register I flag is set (IRQ- disabled), the IRQ- interrupt will cause the next instruction (following the WAI instruction) to be executed without going to the IRQ- interrupt handler. This method results in the highest speed response to an IRQ- input. When an interrupt is received after an ABORT- which occurs during the WAI instruction, the processor will return to the WAI instruction. Other than RES- (highest priority), ABORT- is the next highest priority, followed by NMI- or IRQ- interrupts.

8.15 The STP instruction disables the PHI2 clock to all circuitry. When disabled, the PHI2 clock is held in the high state. In this case, the Dta Bus will remain in the data transfer state and the Bank address will not be multiplexed onto the Data Bus. Upon executing the STP instruction, the RES- signal is the only input which can restart the processor. The processor is restarted by enabling the PHI2 clock, which occurs on the falling edge of the RES- input. Note that the external oscillator must be stable and operating properly before RES- goes high.

8.16 COP Signatures

Signatures 00-7F may be user defined, while signatures 80-FF are reserved for instructions on future microprocessors. Contact WDC for software emulation of future microprocessor hardware functions.

8.17 WDM Opcode Use

The WDM opcode will be used on future microprocessors.

more and less by a

MALL MEGGEO COME CIAN

8.18 RDY Pulled During Write and provided the bos , RNE absence of the state of the

The NMOS 6502 does not stop during a write operation. In contrast, both the W65C02 and the W65C832 do stop during write operations. The W65C832 stops during a write when in the Native mode, but does not stop when in the Emulation NAMES OF DEPOSIT OF THE OWNERS OF THE OWNERS OF THE PARTY OF THE OWNERS OF THE OWNERS

8.19 MVN and MVP Affects on the Data Bank Register

The MVN and MVP instructions change the Data Bank Register to the value of the second byte of the instruction (destination bank address).

8.20 Interrupt Priorities

The following interrupt priorities will be in effect should more than one interrupt occur at the same time:

> RES-Highest Priority ABORT-

ray NMI- or nowahoors and spouls how was 100 allog police day letter IRQ- Lowest Priority

sortion selbook Japaneses and at Leasure 8.21 Transfers from differing register sizes

All transfers from one register to another will result in a full 32-bit output from the source register. The destination register size will determine the number of bits actually stored in the destination register and the values stored in the processor Status Register. The following are always 16-bit transfers, regardless of the accumulator size:

the fift then by the little of the best the little of the

Amore a section to a section of the account of the

TAS: TSA: TAD: TDA

8.22 Stack Transfers

When in the W65C02 Emulation mode, a 01 is forced into the high byte of the 16-bit stack pointer. When in the Native mode or W65C816 Emulation mode, the A Accumulator is transferred to the 16-bit stack pointer. Note that in both the Emulation and Native modes, the full 16 bits of the Stack Register are transferred to the A Accumulator regardless of the state of the M bit in the Status Register.

8.23 REP/SEP

WDC had problems using the REP and SEP instructions in early versions of the high-speed W65C816 and W65C802 devices and has been corrected on all W65C832 devices.

512/ inaker MARCH 1990