DOCUMENT NO. 29000 D39
REV. 3, FEBRUARY 1979

PART NUMBER
R650X and R651X

Rockwell

R6500 Microcomputer System DATA SHEET

R6500 MICROPROCESSORS (CPU's)

SYSTEM ABSTRACT

The 8-bit R6500 microcomputer system is produced with N-Channel, Silicon Gate technology. Its performance speeds are enhanced by advanced system architecture. This innovative architecture results in smaller chips — the semiconductor threshold to cost-effectivity. System cost-effectivity is further enhanced by providing a family of 10 software-compatible microprocessor (CPU) devices, described in this document. Rockwell also provides memory and microcomputer system . . . as well as low-cost design aids and documentation.

R6500 MICROPROCESSOR (CPU) CONCEPT

Ten CPU devices are available. All are software-compatible. They provide options of addressable memory, interrupt input, on-chip clock oscillators and drivers. All are bus-compatible with earlier generation microprocessors like the M6800 devices.

The family includes six microprocessors with on-board clock oscillators and drivers and four microprocessors driven by external clocks. The on-chip clock versions are aimed at high performance, low cost applications where single phase inputs, crystal or RC inputs provide the time base. The external clock versions are geared for multiprocessor system applications where maximum timing control is mandatory. All R6500 microprocessors are also available in a variety of packaging (ceramic and plastic), operating frequency (1 MHz and 2 MHz) and temperature (commercial, industrial and military) versions.

MEMBERS OF THE R6500 MICROPROCESSOR (CPU) FAMILY

Microprocessors with On-Chip Clock Oscillator

Model	Addressable Memory
R6502	65K Bytes
R6503	4K Bytes
R6504	8K Bytes
R6505	4K Bytes
R6506	4K Bytes
R6507	8K Bytes

Microprocessors with External Two Phase Clock Output

Model	Addressable Memory
R6512	65K Bytes
R6513	4K Bytes
R6514	8K Bytes
R6515	4K Bytes

FEATURES

- Single +5V supply
- N channel, silicon gate, depletion load technology
- Eight bit parallel processing
- 56 Instructions
- Decimal and binary arithmetic
- Thirteen addressing modes
- True indexing capability
- Programmable stack pointer
- Variable length stack
- Interrupt capability
- Non-maskable interrupt
- Use with any type of speed memory
- 8-bit Bidirectional Data Bus
- Addressable memory range of up to 65K bytes
- "Ready" input
- Direct Memory Access capability
- Bus compatible with M6800
- 1 MHz and 2 MHz operation
- Choice of external or on-chip clocks
- On-the-chip clock options
  - External single clock input
  - RC time base input
  - Crystal time base input
- Commercial, industrial and military temperature versions
- Pipeline architecture

Ordering Information

Order Number: R65XX_ _ _

- Temperature Range:
  - No suffix = 0°C to +70°C
  - E = -40°C to +85°C (Industrial)
  - MT = -55°C to +125°C (Military)
  - M = MIL-STD-883, Class B
- Package:
  - C = Ceramic
  - P = Plastic (Not Avaible for M or MT suffix)
- Frequency Range:
  - No suffix = 1 MHz
  - A = 2 MHz
- Model Designator:
  - XX = 02, 03, 04, ... 15

NOTE: Contact your local Rockwell Representative concerning availability.

R6500 MICROPROCESSORS (CPU's)

R6500 Signal Description

Clocks (ϕ_1, ϕ_2)

The R651X requires a two phase non-overlapping clock that runs at the V_{CC} voltage level.

The R650X clocks are supplied with an internal clock generator. The frequency of these clocks is externally controlled.

Address Bus (A0-A15)

These outputs are TTL compatible, capable of driving one standard TTL load and 130 pF.

Data Bus (D0-D7)

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

Data Bus Enable (DBE)

This TTL compatible input allows external control of the tri-state data output buffers and will enable the microprocessor bus driver when in the high state. In normal operation DBE would be driven by the phase two (ϕ_2) clock, thus allowing data output from microprocessor only during ϕ_2. During the read cycle, the data bus drivers are internally disabled, becoming essentially an open circuit. To disable data bus drivers externally, DBE should be held low.

Ready (RDY)

This input signal allows the user to halt or single cycle the microprocessor on all cycles except write cycles. A negative transition to the low state during or coincident with phase one (ϕ_1) will halt the microprocessor with the output address lines reflecting the current address being fetched. If Ready is low during a write cycle, it is ignored until the following read operation. This condition will remain through a subsequent phase two (ϕ_2) in which the Ready signal is low. This feature allows microprocessor interfacing with the low speed PROMs as well as fast (max. 2 cycle) Direct Memory Access (DMA).

Interrupt Request ($\overline{IRQ}$)

This TTL level input requests that an interrupt sequence begin within the microprocessor. The microprocessor will complete the current instruction being executed before recognizing the request. At that time, the interrupt mask bit in the Status Code Register will be examined. If the interrupt mask flag is not set, the microprocessor will begin an interrupt sequence. The Program Counter and Processor Status Register are stored in the stack. The microprocessor will then set the interrupt mask flag high so that no further interrupts may occur. At the end of this cycle, the program counter low will be loaded from address FFFE, and program counter high from location FFFF, therefore transferring program control to the memory vector located at these addresses. The RDY signal must be in the high state for any interrupt to be recognized. A 3KΩ external resistor should be used for proper wire-OR operation.

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

A negative going edge on this input requests that a non-maskable interrupt sequence be generated within the microprocessor.

$\overline{NMI}$ is an unconditional interrupt. Following completion of the current instruction, the sequence of operations defined for $\overline{IRQ}$ will be performed, regardless of the state interrupt mask flag. The vector address loaded into the program counter, low and high, are locations FFFA and FFFB respectively, thereby transferring program control to the memory vector located at these addresses. The instructions loaded at these locations cause the microprocessor to branch to a non-maskable interrupt routine in memory.

$\overline{NMI}$ also requires an external 3K Ω register to V_{CC} for proper wire-OR operations.

Inputs $\overline{IRQ}$ and $\overline{NMI}$ are hardware interrupts lines that are sampled during ϕ_2 (phase 2) and will begin the appropriate interrupt routine on the ϕ_1 (phase 1) following the completion of the current instruction.

Set Overflow Flag (S.O.)

A negative going edge on this input sets the overflow bit in the Status Code Register. This signal is sampled on the trailing edge of $\phi 1$ and must be externally synchronized.

SYNC

This output line is provided to identify those cycles in which the microprocessor is doing an OP CODE fetch. The SYNC line goes high during ϕ_1 of an OP CODE fetch and stays high for the remainder of that cycle. If the RDY line is pulled low during the ϕ_1 clock pulse in which SYNC went high, the processor will stop in its current state and will remain in the state until the RDY line goes high. In this manner, the SYNC signal can be used to control RDY to cause single instruction execution.

Reset

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

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

After V_{CC} reaches 4.75 volts in a power up routine, reset must be held low for at least two clock cycles. At this time the R/W and (SYNC) signal will become valid.

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

ADDRESSING MODES

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

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

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

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

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

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

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

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

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

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

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

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

INSTRUCTION SET — ALPHABETIC SEQUENCE

ADC Add Memory to Accumulator with Carry
AND "AND" Memory with Accumulator
ASL Shift left One Bit (Memory or Accumulator)

BCC Branch on Carry Clear
BCS Branch on Carry Set
BEQ Branch on Result Zero
BIT Test Bits in Memory with Accumulator
BMI Branch on Result Minus
BNE Branch on Result not Zero
BPL Branch on Result Plus
BRK Force Break
BVC Branch on Overflow Clear
BVS Branch on Overflow Set

CLC Clear Carry Flag
CLD Clear Decimal Mode
CLI Clear Interrupt Disable Bit
CLV Clear Overflow Flag
CMP Compare Memory and Accumulator
CPX Compare Memory and Index X
CPY Compare Memory and Index Y

DEC Decrement Memory by One
DEX Decrement Index X by One
DEY Decrement Index Y by One

EOR "Exclusive-or" Memory with Accumulator

INC Increment Memory by One
INX Increment Index X by One
INY Increment Index Y by One

JMP Jump to New Location
JSR Jump to New Location Saving Return Address

LDA Load Accumulator with Memory
LDX Load Index X with Memory
LDY Load Index Y with Memory
LSR Shift One Bit Right (Memory or Accumulator)

NOP No Operation

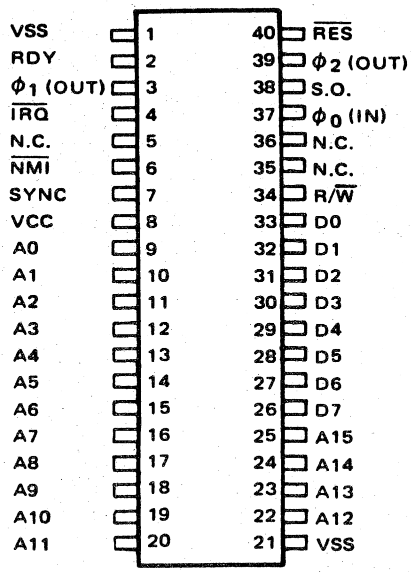
ORA "OR" Memory with Accumulator

PHA Push Accumulator on Stack
PHP Push Processor Status on Stack
PLA Pull Accumulator from Stack
PLP Pull Processor Status from Stack

ROL Rotate One Bit Left (Memory or Accumulator)
ROR Rotate One Bit Right (Memory or Accumulator)
RTI Return from Interrupt
RTS Return from Subroutine

SBC Subtract Memory from Accumulator with Borrow
SEC Set Carry Flag
SED Set Decimal Mode
SEI Set Interrupt Disable Status
STA Store Accumulator in Memory
STX Store Index X in Memory
STY Store Index Y in Memory

TAX Transfer Accumulator to Index X
TAY Transfer Accumulator to Index Y
TSX Transfer Stack Pointer to Index X
TXA Transfer Index X to Accumulator
TXS Transfer Index X to Stack Register
TYA Transfer Index Y to Accumulator



R6502 — 40 Pin Package

Features of R6502

- 65K Addressable Bytes of Memory (A0-A15)
- IRQ Interrupt
- On-the-chip Clock
  - TTL Level Single Phase Input
  - RC Time Base Input
  - Crystal Time Base Input
- SYNC Signal
  (can be used for single instruction execution)
- RDY Signal
  (can be used to halt or single cycle execution)
- Two Phase Output Clock for Timing of Support Chips
- NMI Interrupt

INSTRUCTION SET

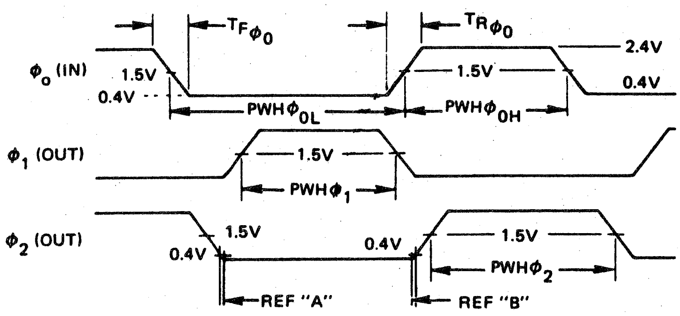
INSTRUCTIONS		IMMEDIATE			ABSOLUTE			ZERO PAGE			ACCUM			IMPLIED		
MNEMONIC	OPERATION	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#
ADC	A + M + C → A (4) (1)	69	2	2	6D	4	3	65	3	2						
AND	A ∧ M → A (1)	29	2	2	2D	4	3	25	3	2						
ASL	C ← [7 0] ← 0				0E	6	3	06	5	2	0A	2	1			
BCC	BRANCH ON C = 0 (2)															
BCS	BRANCH ON C = 1 (2)															
BEQ	BRANCH ON Z = 1 (2)															
BIT	A ∧ M				2C	4	3	24	3	2						
BMI	BRANCH ON N = 1 (2)															
BNE	BRANCH ON Z = 0 (2)															
BPL	BRANCH ON N = 0 (2)															
BRK	BREAK													00	7	1
BVC	BRANCH ON V = 0 (2)															
BVS	BRANCH ON V = 1 (2)															
CLC	0 → C													18	2	1
CLD	0 → D													D8	2	1
CLI	0 → I													58	2	1
CLV	0 → V													B8	2	1
CMP	A − M	C9	2	2	CD	4	3	C5	3	2						
CPX	X − M	E0	2	2	EC	4	3	E4	3	2						
CPY	Y − M	C0	2	2	CC	4	3	C4	3	2						
DEC	M − 1 → M				CE	6	3	C6	5	2						
DEX	X − 1 → X													CA	2	1
DEY	Y − 1 → Y													88	2	1
EOR	A ⊻ M → A (1)	49	2	2	4D	4	3	45	3	2						
INC	M + 1 → M				EE	6	3	E6	5	2						
INX	X + 1 → X													E8	2	1
INY	Y + 1 → Y													C8	2	1
JMP	JUMP TO NEW LOC				4C	3	3									
JSR	JUMP SUB				20	6	3									
LDA	M → A (1)	A9	2	2	AD	4	3	A5	3	2						
LDX	M → X (1)	A2	2	2	AE	4	3	A6	3	2						
LDY	M → Y (1)	A0	2	2	AC	4	3	A4	3	2						
LSR	0 → [7 0] → C				4E	6	3	46	5	2	4A	2	1			
NOP	NO OPERATION													EA	2	1
ORA	A V M → A	09	2	2	0D	4	3	05	3	2						
PHA	A → Ms S − 1 → S													48	3	1
PHP	P → Ms S − 1 → S													08	3	1
PLA	S + 1 → S Ms → A													68	4	1
PLP	S + 1 → S Ms → P													28	4	1
ROL	[7 0] ← [C] ←				2E	6	3	26	5	2	2A	2	1			
ROR	→ [C] → [7 0]				6E	6	3	66	5	2	6A	2	1			
RTI	RTRN INT													40	6	1
RTS	RTRN SUB													60	6	1
SBC	A − M − $\bar{C}$ → A (1)	E9	2	2	ED	4	3	E5	3	2						
SEC	1 → C													38	2	1
SED	1 → D													F8	2	1
SEI	1 → I													78	2	1
STA	A → M				8D	4	3	85	3	2						
STX	X → M				8E	4	3	86	3	2						
STY	Y → M				8C	4	3	84	3	2						
TAX	A → X													AA	2	1
TAY	A → Y													A8	2	1
TSX	S → X													BA	2	1
TXA	X → A													8A	2	1
TXS	X → S													9A	2	1
TYA	Y → A													98	2	1

INSTRUCTIONS	(IND, X)			(IND), Y			Z PAGE, X			ABS, X			ABS, Y			RELATIVE			INDIRECT			Z PAGE, Y			PROCESSOR STATUS CODES 7 6 5 4 3 2 1 0	
MNEMONIC	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	OP	n	#	N V • B D I Z C	MNEMONIC
ADC	61	6	2	71	5	2	75	4	2	7D	4	3	79	4	3										N V • • • • Z C	ADC
AND	21	6	2	31	5	2	35	4	2	3D	4	3	39	4	3										N • • • • • Z •	AND
ASL							16	6	2	1E	7	3													N • • • • • Z C	ASL
BCC																90	2	2							• • • • • • • •	BCC
BCS																B0	2	2							• • • • • • • •	BCS
BEQ																F0	2	2							• • • • • • • •	BEQ
BIT																									M_7 M_6 • • • • Z •	BIT
BMI																30	2	2							• • • • • • • •	BMI
BNE																D0	2	2							• • • • • • • •	BNE
BPL																10	2	2							• • • • • • • •	BPL
BRK																									• • • 1 • 1 • •	BRK
BVC																50	2	2							• • • • • • • •	BVC
BVS																70	2	2							• • • • • • • •	BVS
CLC																									• • • • • • • 0	CLC
CLD																									• • • • 0 • • •	CLD
CLI																									• • • • • 0 • •	CLI
CLV																									• 0 • • • • • •	CLV
CMP	C1	6	2	D1	5	2	D5	4	2	DD	4	3	D9	4	3										N • • • • • Z C	CMP
CPX																									N • • • • • Z C	CPX
CPY																									N • • • • • Z C	CPY
DEC							D6	6	2	DE	7	3													N • • • • • Z •	DEC
DEX																									N • • • • • Z •	DEX
DEY																									N • • • • • Z •	DEY
EOR	41	6	2	51	5	2	55	4	2	5D	4	3	59	4	3										N • • • • • Z •	EOR
INC							F6	6	2	FE	7	3													N • • • • • Z •	INC
INX																									N • • • • • Z •	INX
INY																									N • • • • • Z •	INY
JMP																			6C	5	3				• • • • • • • •	JMP
JSR																									• • • • • • • •	JSR
LDA	A1	6	2	B1	5	2	B5	4	2	BD	4	3	B9	4	3										N • • • • • Z •	LDA
LDX													BE	4	3							B6	4	2	N • • • • • Z •	LDX
LDY							B4	4	2	BC	4	3													N • • • • • Z •	LDY
LSR							56	6	2	5E	7	3													0 • • • • • Z C	LSR
NOP																									• • • • • • • •	NOP
ORA	01	6	2	11	5	2	15	4	2	1D	4	3	19	4	3										N • • • • • Z •	ORA
PHA																									• • • • • • • •	PHA
PHP																									• • • • • • • •	PHP
PLA																									N • • • • • Z •	PLA
PLP																									(RESTORED)	PLP
ROL							36	6	2	3E	7	3													N • • • • • Z C	ROL
ROR							76	6	2	7E	7	3													N • • • • • Z C	ROR
RTI																									(RESTORED)	RTI
RTS																									• • • • • • • •	RTS
SBC	E1	6	2	F1	5	2	F5	4	2	FD	4	3	F9	4	3										N V • • • • Z (3)	SBC
SEC																									• • • • • • • 1	SEC
SED																									• • • • 1 • • •	SED
SEI																									• • • • • 1 • •	SEI
STA	81	6	2	91	6	2	95	4	2	9D	5	3	99	5	3										• • • • • • • •	STA
STX																						96	4	2	• • • • • • • •	STX
STY							94	4	2																• • • • • • • •	STY
TAX																									N • • • • • Z •	TAX
TAY																									N • • • • • Z •	TAY
TSX																									N • • • • • Z •	TSX
TXA																									N • • • • • Z •	TXA
TXS																									• • • • • • • •	TXS
TYA																									N • • • • • Z •	TYA

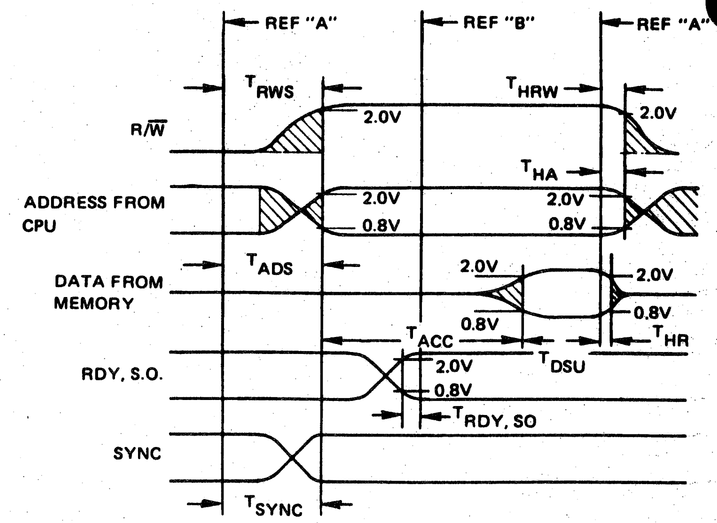
(1) ADD 1 to N IF PAGE BOUNDARY IS CROSSED
(2) ADD 1 TO N IF BRANCH OCCURS TO SAME PAGE
ADD 2 TO N IF BRANCH OCCURS TO DIFFERENT PAGE
(3) CARRY NOT = BORROW
(4) IF IN DECIMAL MODE Z FLAG IS INVALID
ACCUMULATOR MUST BE CHECKED FOR ZERO RESULT

X	INDEX X	+	ADD	M_7	MEMORY BIT 7
Y	INDEX Y	−	SUBTRACT	M_6	MEMORY BIT 6
A	ACCUMULATOR	∧	AND	n	NO. CYCLES
M	MEMORY PER EFFECTIVE ADDRESS	V	OR	#	NO. BYTES
Ms	MEMORY PER STACK POINTER	⊻	EXCLUSIVE OR		

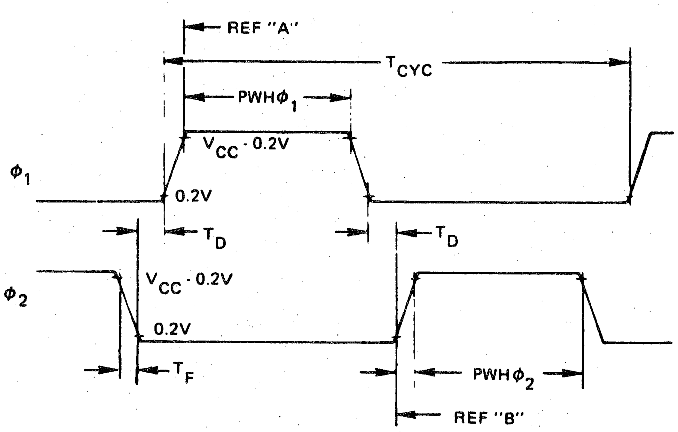
Clock Timing – R6502, 03, 04, 05, 06, 07



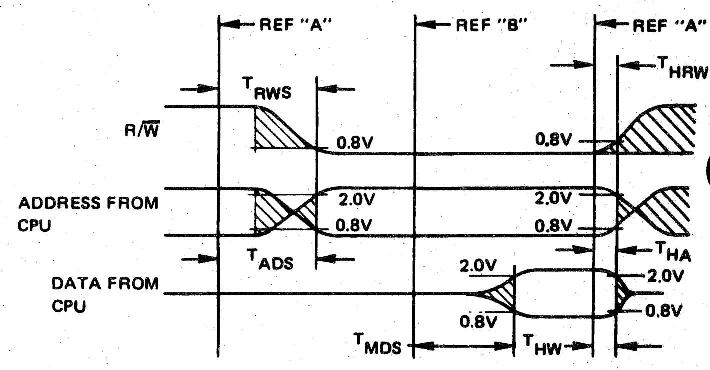
Timing for Reading Data from Memory or Peripherals



Clock Timing – R6512, 13, 14, 15

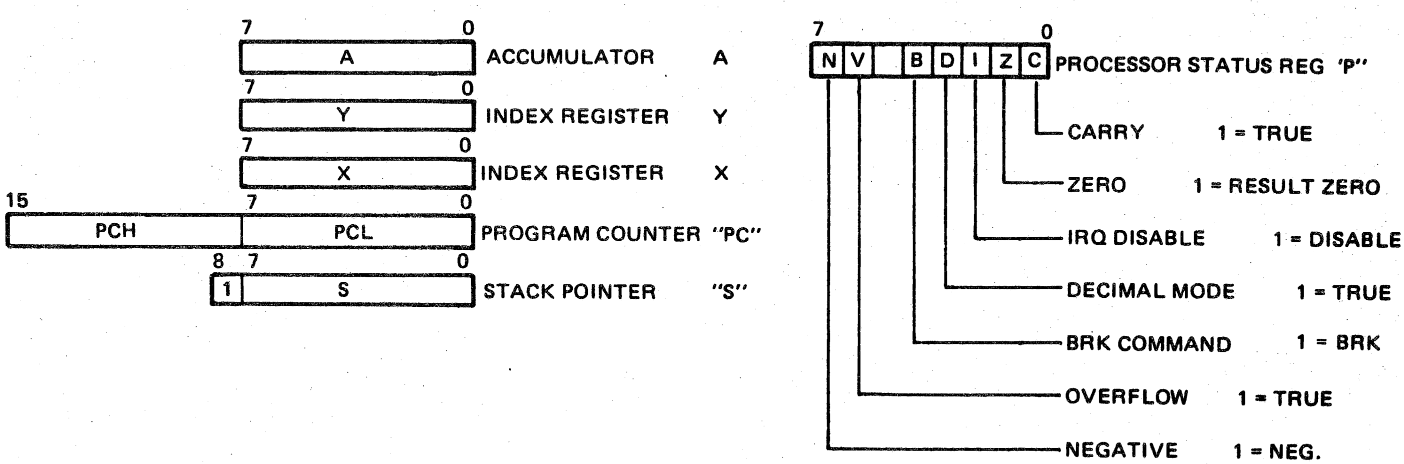


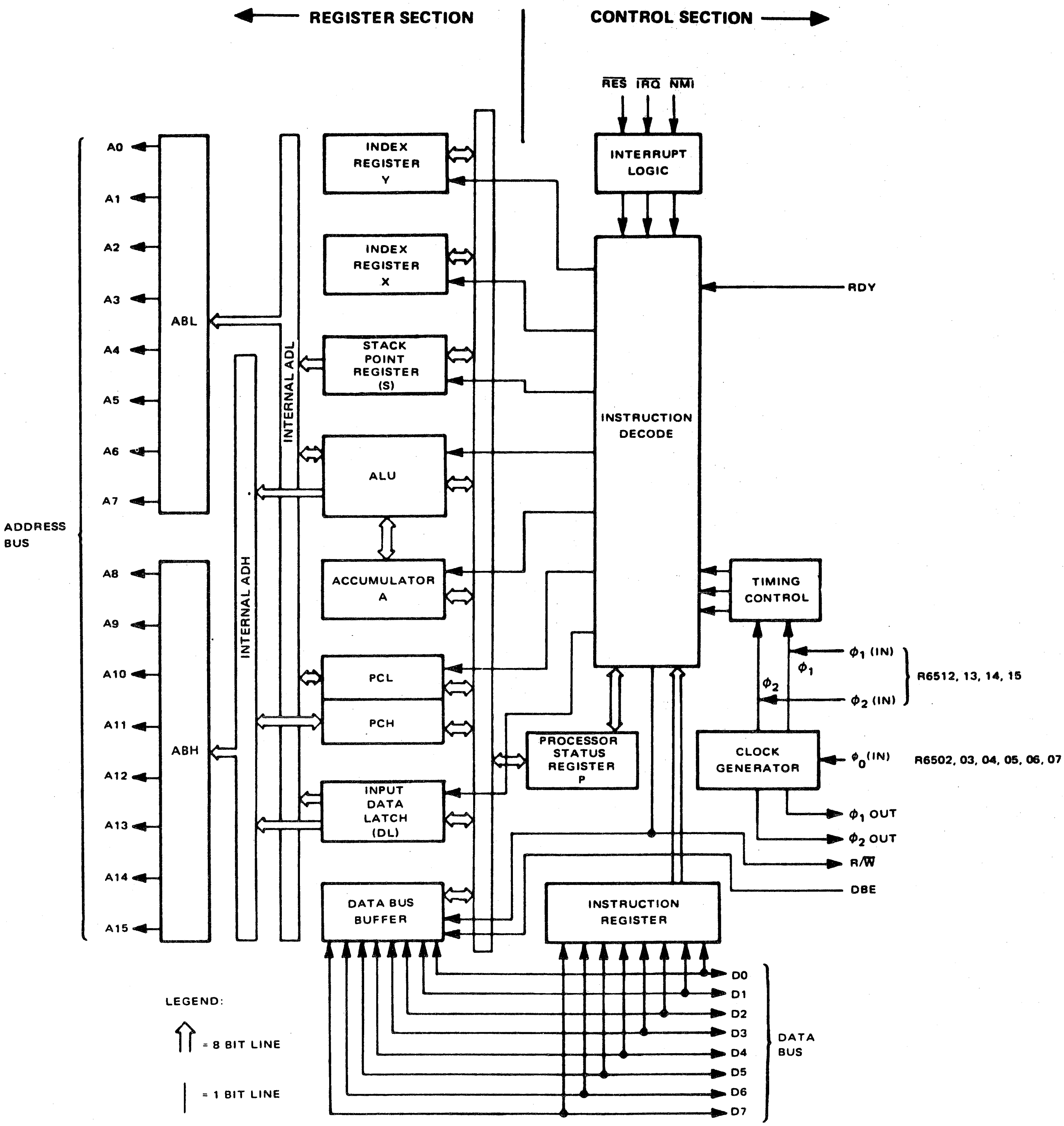
Timing for Writing Data to Memory or Peripherals



Note: "REF." means Reference Points on clocks.

PROGRAMMING MODEL





Note:
1. Clock Generator is not included on R6512, 13, 14, 15
2. Addressing Capability and control options vary with each of the R6500 Products.

R6500 Internal Architecture