MOTOROLA
SEMICONDUCTORS
3501 ED BLUESTEIN BLVD., AUSTIN, TEXAS 78721

MC146805E2

Advance Information

CMOS
(HIGH PERFORMANCE SILICON GATE)

8-BIT MICROPROCESSOR

8-BIT MICROPROCESSOR UNIT

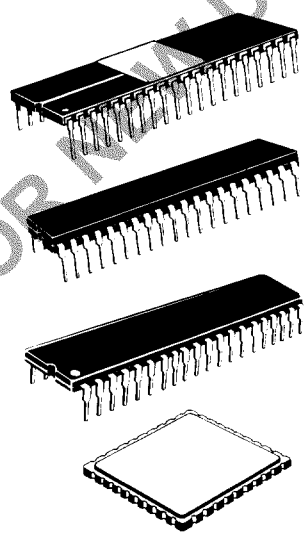
The MC146805E2 Microprocessor Unit (MPU) belongs to the M6805 Family of Microcomputers. This 8-bit fully static and expandable microprocessor contains a CPU, on-chip RAM, I/O, and TIMER. It is a low-power, low-cost processor designed for low-end to mid-range applications in the consumer, automotive, industrial, and communications markets where very low power consumption constitutes an important factor. The following are the major features of the MC146805E2 MPU:

HARDWARE FEATURES

- Typical Full Speed Operating Power of 35 mW @ 5 V
- Typical WAIT Mode Power of 5 mW
- Typical STOP Mode Power of 25 μW
- 112 Bytes of On-Chip RAM
- 16 Bidirectional I/O Lines
- Internal 8-Bit Timer with Software Programmable 7-Bit Prescaler
- External Timer Input
- Full External and Timer Interrupts
- Multiplexed Address/Data Bus
- Master Reset and Power-On Reset
- Capable of Addressing Up to 8K Bytes of External Memory
- Single 3- to 6-Volt Supply
- On-Chip Oscillator
- 40-Pin Dual-In-Line Package
- Chip Carrier Also Available

SOFTWARE FEATURES

- Similar to the MC6800
- Efficient Use of Program Space
- Versatile Interrupt Handling
- True Bit Manipulation
- Addressing Modes with Indexed Addressing for Tables
- Efficient Instruction Set
- Memory Mapped I/O
- Two Power Saving Standby Modes



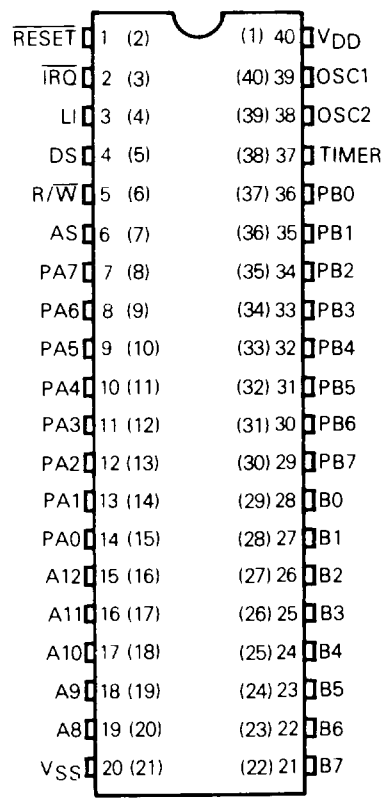
L SUFFIX
CERAMIC PACKAGE
CASE 715

S SUFFIX
CERDIP PACKAGE
CASE 734

P SUFFIX
PLASTIC PACKAGE
CASE 711

Z SUFFIX
CHIP CARRIER
CASE 761

PIN ASSIGNMENT



Pin numbers in parentheses represent equivalent Z suffix chip carrier pins.

GENERIC INFORMATION

Package Type	Frequency (MHz)	Temperature	Generic Number
Ceramic	1.0	0°C to 70°C	MC146805E2L
L Suffix	1.0	−40°C to 85°C	MC146805E2CL
Cerdip	1.0	0°C to 70°C	MC146805E2S
S Suffix	1.0	−40°C to 85°C	MC146805E2CS
Plastic	1.0	0°C to 70°C	MC146805E2P
P Suffix	1.0	−40°C to 85°C	MC146805E2CP
Leadless Chip Carrier	1.0	0°C to 70°C	MC146805E2Z
Z Suffix	1.0	−40°C to 85°C	MC146805E2CZ

ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

This document contains information on a new product. Specifications and information herein are subject to change without notice.

©MOTOROLA INC., 1984

MAXIMUM RATINGS (voltages referenced to V_{SS})

Ratings	Symbol	Value	Unit
Supply Voltage	V_{DD}	−0.3 to +8.0	V
All Input Voltages Except OSC1	V_{in}	V_{SS}−0.5 to V_{DD}+0.5	V
Current Drain Per Pin Excluding V_{DD} and V_{SS}	I	10	mA
Operating Temperature Range MC146805E2 MC146805E2C	T_A	T_L to T_H 0 to 70 −40 to 85	°C
Storage Temperature Range	T_{stg}	−55 to +150	°C

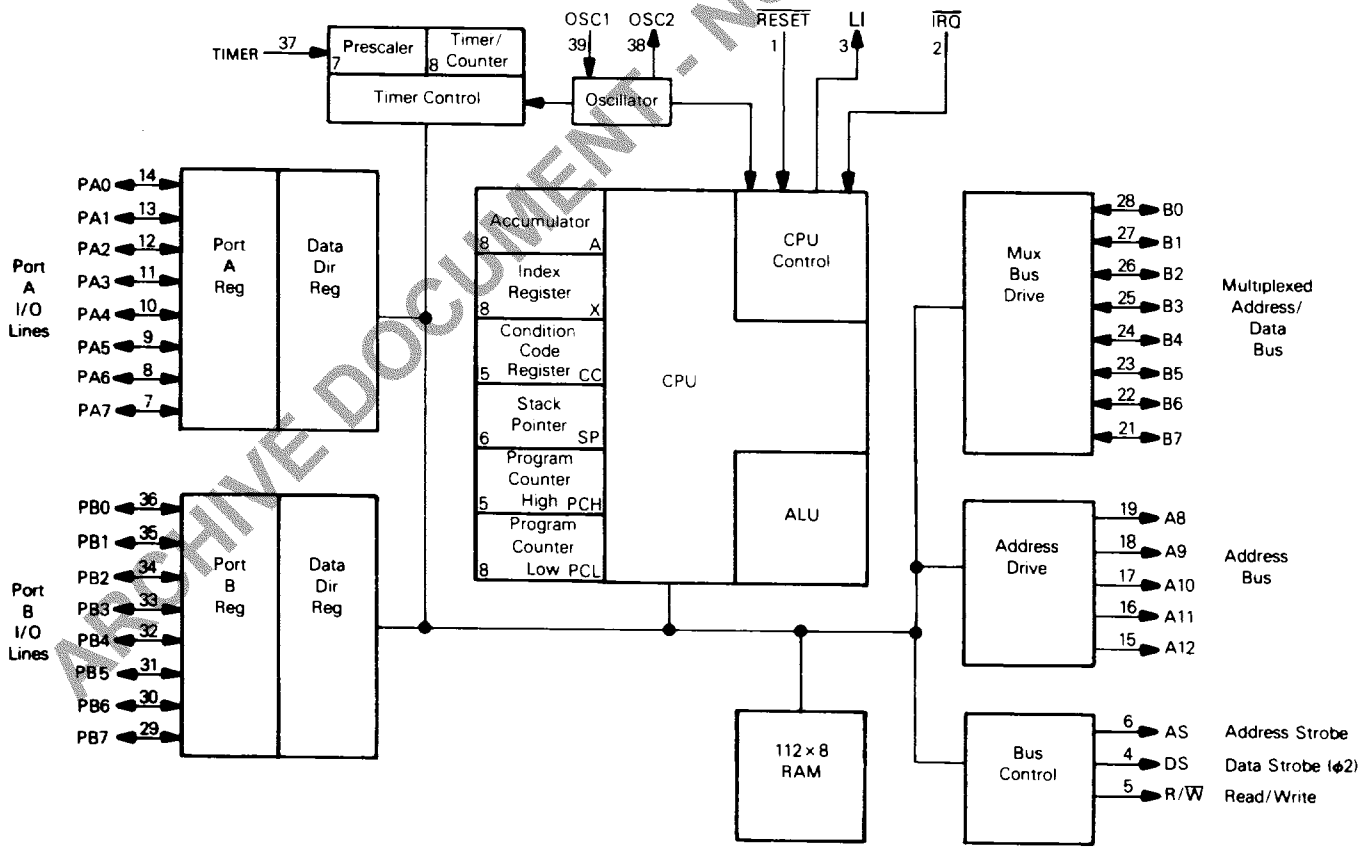
THERMAL CHARACTERISTICS

Characteristics	Symbol	Value	Unit
Thermal Resistance Plastic Cerdip Ceramic Chip-Carrier	θ_{JA}	 100 60 50 TBD	°C/W

This device contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high impedance circuit. For proper operation it is recommended that V_{in} and V_{out} be constrained to the range $V_{SS} \le (V_{in}$ or $V_{out}) \le V_{DD}$. Reliability of operation is enhanced if unused inputs are tied to an appropriate logic voltage level (e.g., either V_{SS} or V_{DD}).

ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

FIGURE 1 — MICROPROCESSOR BLOCK DIAGRAM



DC ELECTRICAL CHARACTERISTICS @ 3.0 V (V_{DD} = 3.0 Vdc, V_{SS} = 0, $T_A = T_L$ to T_H, unless otherwise noted)

Characteristics	Symbol	Min	Max	Unit
Output Voltage ($I_{Load} \leq 10.0\ \mu A$)	V_{OL} V_{OH}	– $V_{DD}-0.1$	0.1 –	V
Total Supply Current (C_L = 50 pF — No dc Loads, t_{cyc} = 5 µs)				
Run (V_{IL} = 0.2 V, $V_{IH} = V_{DD} - 0.2$ V)	I_{DD}	–	1.3	mA
Wait (Test Conditions — See Note Below)	I_{DD}	–	200	µA
Stop (Test Conditions — See Note Below)	I_{DD}	–	100	µA
Output High Voltage				
(I_{Load} = 0.25 mA) A8-A12, B0-B7, DS, AS, R/$\overline{W}$	V_{OH}	2.7	–	V
(I_{Load} = 0.1 mA) PA0-PA7, PB0-PB7	V_{OH}	2.7	–	V
Output Low Voltage				
(I_{Load} = 0.25 mA) A8-A12, B0-B7, PB0-PB7, DS, AS, R/$\overline{W}$, PA0-PA7	V_{OL}	–	0.3	V
Input High Voltage				
PA0-PA7, PB0-PB7, B0-B7	V_{IH}	2.1	–	V
$\overline{TIMER}$, $\overline{IRQ}$, $\overline{RESET}$	V_{IH}	2.5	–	V
OSC1	V_{IH}	2.1	–	V
Input Low Voltage (All Inputs)	V_{IL}	–	0.5	V
Frequency of Operation				
Crystal	f_{osc}	–	1.0	MHz
External Clock	f_{osc}	dc	1.0	MHz
Input Current				
$\overline{RESET}$, $\overline{IRQ}$, TIMER, OSC1	I_{in}	–	±1	µA
Hi-Z Output Leakage				
PA0-PA7, PB0-PB7, B0-B7	I_{TSL}	–	±10	µA
Capacitance				
$\overline{RESET}$, $\overline{IRQ}$, TIMER	C_{in}	–	8.0	pF
Capacitance				
DS, AS, R/$\overline{W}$, A8-A12, PA0-PA7, PB0-PB7, B0-B7	C_{out}	–	12.0	pF

NOTE: Test conditions for Quiescent Current Values are:
Port A and B programmed as inputs.
V_{IL} = 0.2 V for PA0-PA7, PB0-PB7, and B0-B7.
$V_{IH} = V_{DD} - 0.2$ V for $\overline{RESET}$, $\overline{IRQ}$, and TIMER
OSC1 input is a squarewave from V_{SS} + 0.2 V to V_{DD} − 0.2 V.
OSC2 output load (including tester) is 35 pF maximum.
Wait mode I_{DD} is affected linearly by this capacitance.

ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

DC ELECTRICAL CHARACTERISTICS @ 5.0 V (V_{DD}=5.0 Vdc ±10%, V_{SS}=0, T_A=T_L to T_H, unless otherwise noted)

Characteristics	Symbol	Min	Max	Unit
Output Voltage (I_{Load}≤10.0 μA)	V_{OL} V_{OH}	– V_{DD}–0.1	0.1 –	V
Total Supply Current (C_L=130 pF — On Bus, C_L=50 pF — On Ports, No dc Loads, t_{cyc}=1.0 μs, V_{IL}=0.2 V, V_{IH}=V_{DD}–0.2 V) Run	I_{DD}	–	10	mA
Wait (Test Conditions — See Note Below)	I_{DD}	–	1.5	mA
Stop (Test Conditions — See Note Below)	I_{DD}	–	200	μA
Output High Voltage (I_{Load}=1.6 mA) A8-A12, B0-B7, DS, AS, R/$\overline{W}$	V_{OH}	4.1	–	V
(I_{Load}=0.36 mA) PA0-PA7, PB0-PB7	V_{OH}	4.1	–	V
Output Low Voltage (I_{Load}=1.6 mA) A8-A12, B0-B7, PA0-PA7, PB0-PB7, DS, AS, R/$\overline{W}$	V_{OL}	–	0.4	V
Input High Voltage PA0-PA7, PB0-PB7, B0-B7	V_{IH}	V_{DD}–2.0	–	V
$\overline{TIMER}$, $\overline{IRQ}$, $\overline{RESET}$	V_{IH}	V_{DD}–0.8	–	V
OSC1	V_{IH}	V_{DD}–1.5	–	V
Input Low Voltage (All Inputs)	V_{IL}	–	0.8	V
Frequency of Operation Crystal	f_{osc}	–	5.0	MHz
External Clock	f_{osc}	dc	5.0	MHz
Input Current $\overline{RESET}$, $\overline{IRQ}$, TIMER, OSC1	I_{in}	–	±1	μA
Hi-Z Output Leakage PA0-PA7, PB0-PB7, B0-B7	I_{TSI}	–	±10	μA
Capacitance $\overline{RESET}$, $\overline{IRQ}$, TIMER	C_{in}	–	8.0	pF
Capacitance DS, AS, R/$\overline{W}$, A8-A12, PA0-PA7, PB0-PB7, B0-B7	C_{out}	–	12.0	pF

NOTE: Test conditions for Quiescent Current Values are:
- Port A and B programmed as inputs.
- V_{IL}=0.2 V for PA0-PA7, PB0-PB7, and B0-B7.
- V_{IH}=V_{DD}–0.2 V for $\overline{RESET}$, $\overline{IRQ}$, and TIMER.
- OSC1 input is a squarewave from V_{SS}+0.2 V to V_{DD} – 0.2 V.
- OSC2 output load (including tester) is 35 pF maximum.
- Wait mode (I_{DD}) is affected linearly by this capacitance.

ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

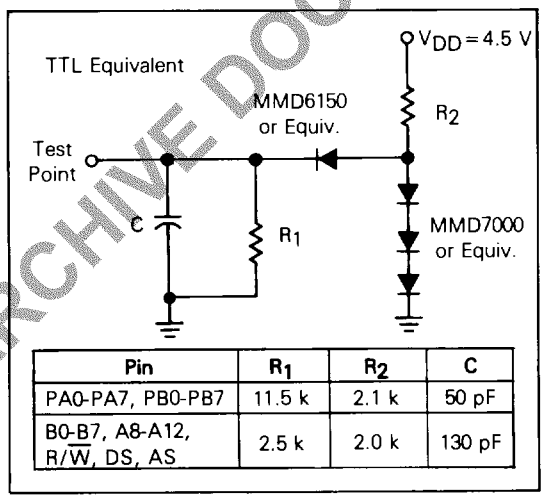
TABLE 1 — CONTROL TIMING ($V_{SS}=0$, $T_A=T_L$ to T_H)

Characteristics	Symbol	$V_{DD}=3.0$ V, $f_{osc}=1$ MHz Min	Typ	Max	$V_{DD}=5.0$ V ± 10%, $f_{osc}=5.0$ MHz Min	Typ	Max	Unit
I/O Port Timing — Input Setup Time (Figure 3)	t_{PVASL}	500	–	–	250	–	–	ns
Input Hold Time (Figure 3)	t_{ASLPX}	100	–	–	100	–	–	ns
Output Delay Time (Figure 3)	t_{ASLPV}	–	–	0	–	–	0	ns
Interrupt Setup Time (Figure 6)	t_{ILASL}	2	–	–	0.4	–	–	μs
Crystal Oscillator Startup Time (Figure 5)	t_{OXOV}	–	30	300	–	15	100	ms
Wait Recovery Startup Time (Figure 7)	t_{IVASH}	–	–	10	–	–	2	μs
Stop Recovery Startup Time (Crystal Oscillator) (Figure 8)	t_{ILASH}	–	30	300	–	15	100	ms
Required Interrupt Release (Figure 6)	t_{DSLIH}	–	–	5	–	–	1.0	μs
Timer Pulse Width (Figure 7)	t_{TH}, t_{TL}	0.5	–	–	0.5	–	–	t_{cyc}
Reset Pulse Width (Figure 5)	t_{RL}	5.5	–	–	1.5	–	–	μs
Timer Period (Figure 7)	t_{TLTL}	1.0	–	–	1.0	–	–	t_{cyc}
Interrupt Pulse Width Low (Figure 16)	t_{ILIH}	1.0	–	–	1.0	–	–	t_{cyc}
Interrupt Pulse Period (Figure 16)	t_{ILIL}	*	–	–	*	–	–	t_{cyc}
Oscillator Cycle Period (1/5 of t_{cyc})	t_{OLOL}	1000	–	–	200	–	–	ns
OSC1 Pulse Width High	t_{OH}	350	–	–	75	–	–	ns
OSC1 Pulse Width Low	t_{OL}	350	–	–	75	–	–	ns

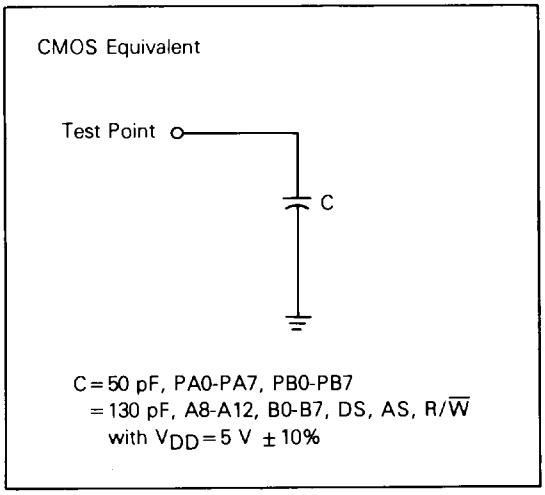
*The minimum period t_{ILIL} should not be less than the number of t_{cyc} cycles it takes to execute the interrupt service routine plus 20 t_{cyc} cycles.

ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

FIGURE 2 — EQUIVALENT TEST LOADS

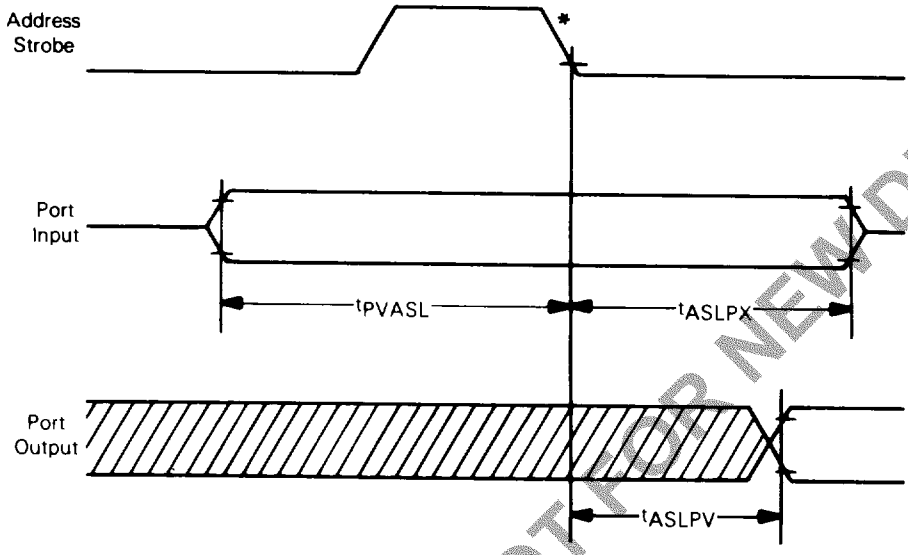


Pin	R_1	R_2	C
PA0-PA7, PB0-PB7	11.5 k	2.1 k	50 pF
B0-$\overline{B7}$, A8-A12, R/$\overline{W}$, DS, AS	2.5 k	2.0 k	130 pF



C = 50 pF, PA0-PA7, PB0-PB7
= 130 pF, A8-A12, B0-B7, DS, AS, R/$\overline{W}$
with $V_{DD}=5$ V ± 10%

FIGURE 3 — I/O PORT TIMING
(V_{Low} = 0.8 V, V_{High} = V_{DD} − 2.0 V, V_{DD} = 5.0 ± 10%
T_A = T_L to T_H, C_L on Port = 50 pF, f_{osc} = 5 MHz)



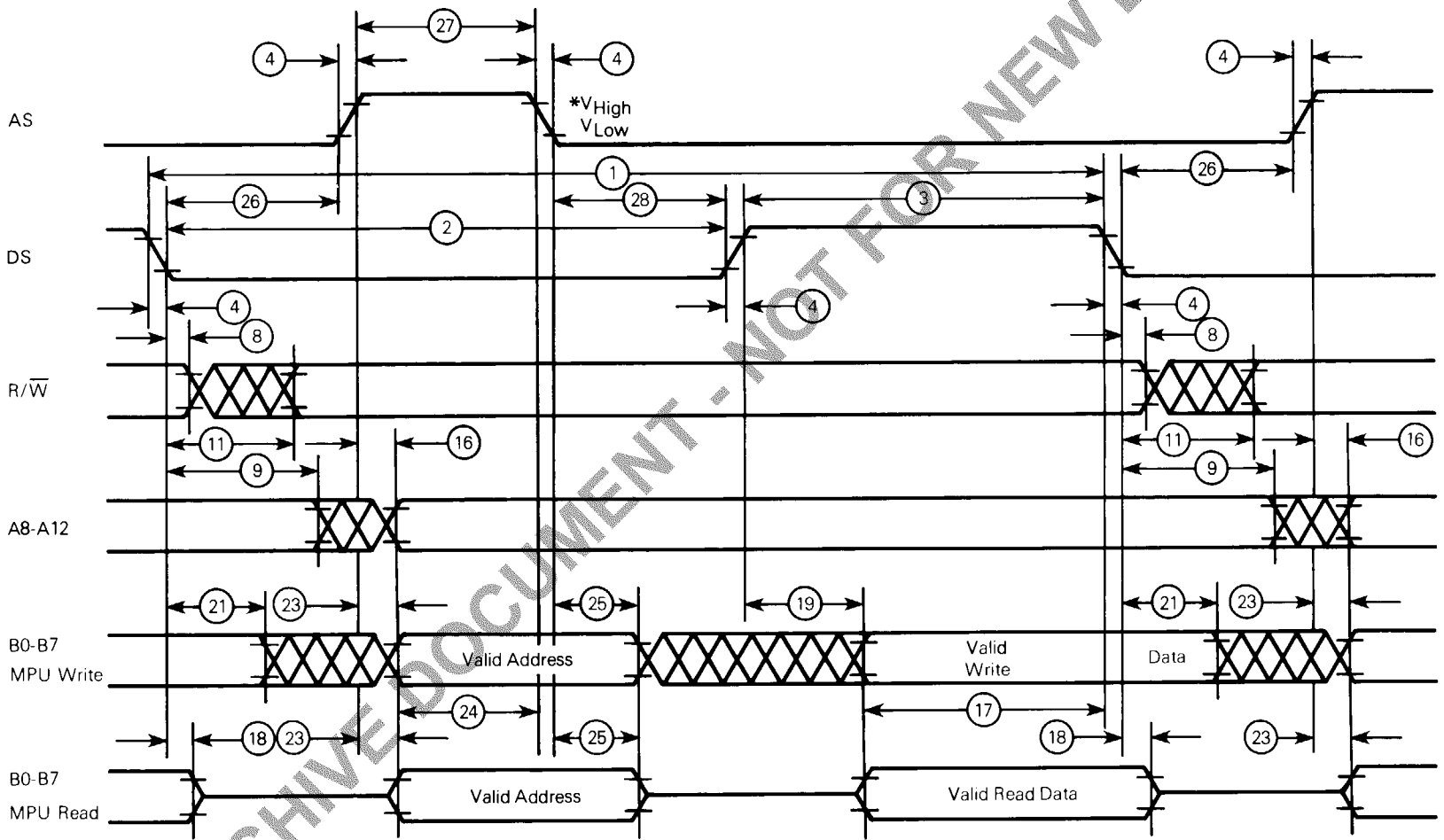
* The address strobe of the first cycle of the next instruction.

MOTOROLA DOCUMENT - NOT FOR NEW DESIGN

TABLE 2 — BUS TIMING (T_A = T_L to T_H, V_{SS} = 0 V) See Figure 4

Num	Characteristics	Symbol	f_{osc} = 1 MHz, V_{DD} = 3.0 V, 50 pF Load		f_{osc} = 5 MHz, V_{DD} = 5.0 V ± 10%, 1 TTL and 130 pF Load		Unit
			Min	Max	Min	Max	
1	Cycle Time	t_{cyc}	5000	dc	1000	dc	ns
2	Pulse Width, DS Low	PW_{EL}	2800	–	560	–	ns
3	Pulse Width, DS High	PW_{EH}	1800	–	375	–	ns
4	Clock Transition	t_r, t_f	–	100	–	30	ns
8	R/$\overline{W}$ Hold	t_{RWH}	10	–	10	–	ns
9	Non-Muxed Address Hold	t_{AH}	800	–	100	–	ns
11	R/$\overline{W}$ Delay from DS Fall	t_{AD}	–	500	–	300	ns
16	Non-Muxed Address Delay from AS Rise	t_{ADH}	0	200	0	100	ns
17	MPU Read Data Setup	t_{DSR}	200	–	115	–	ns
18	Read Data Hold	t_{DHR}	0	800	0	160	ns
19	MPU Data Delay, Write	t_{DDW}	–	0	–	120	ns
21	Write Data Hold	t_{DHW}	800	–	55	–	ns
23	Muxed Address Delay from AS Rise	t_{BHD}	0	250	0	120	ns
24	Muxed Address Valid to AS Fall	t_{ASL}	600	–	55	–	ns
25	Muxed Address Hold	t_{AHL}	250	750	60	180	ns
26	Delay DS Fall to AS Rise	t_{ASD}	800	–	160	–	ns
27	Pulse Width, AS High	PW_{ASH}	850	–	175	–	ns
28	Delay, AS Fall to DS Rise	t_{ASED}	800	–	160	–	ns

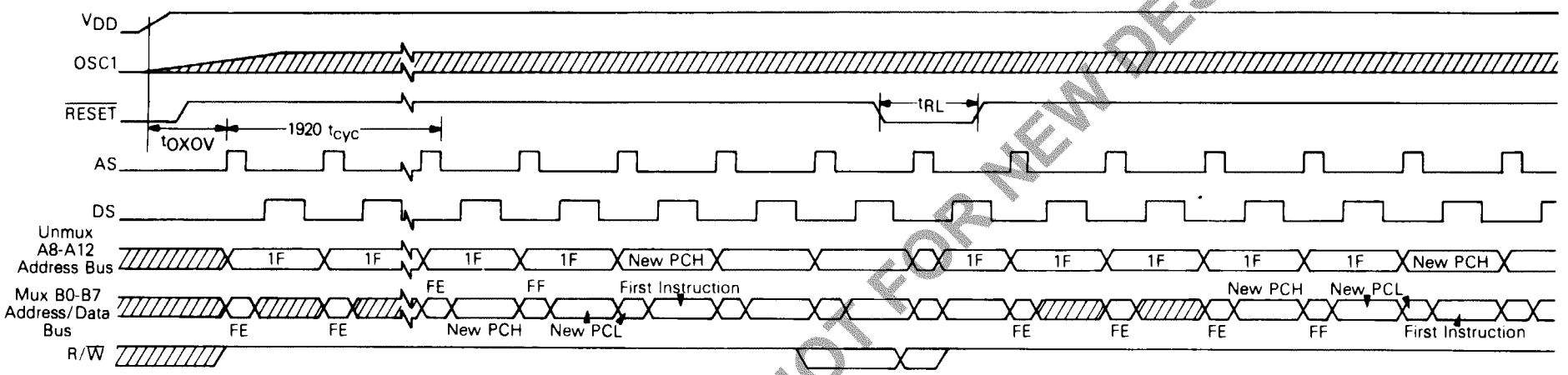
FIGURE 4 — MC146805E2 BUS TIMING



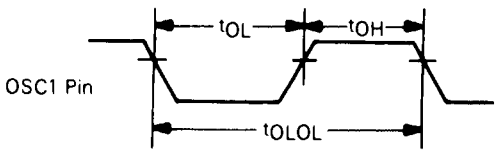
* V_{High} = 2.0 V, V_{Low} = 0.5 V for V_{DD} = 3 V for outputs only.
V_{High} = V_{DD} − 2.0 V, V_{Low} = 0.8 V for V_{DD} = 5 V ± 10% for outputs only.

ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

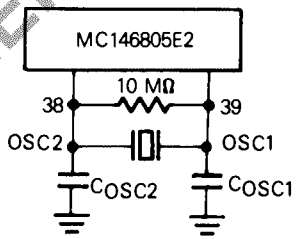
FIGURE 5 — POWER-ON RESET AND $\overline{\text{RESET}}$ TIMING



Oscillator Waveform



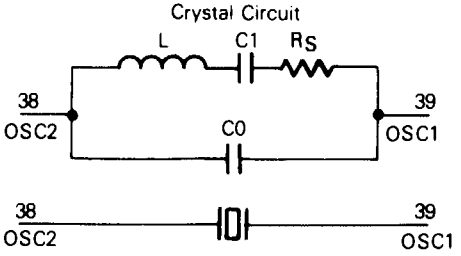
Crystal Oscillator Connections



Crystal Parameters Representative Frequencies

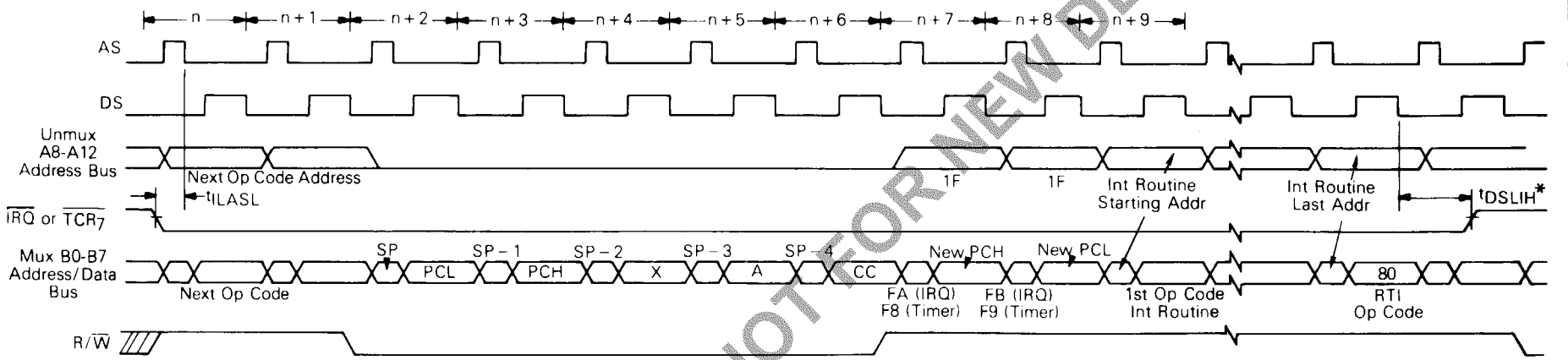
	5.0 MHz	4.0 MHz	1.0 MHz
R_S max	50Ω	75Ω	400Ω
C0	8 pF	7 pF	5 pF
C1	0.02 pF	0.012 pF	0.008 pF
Q	50 k	40 k	30 k
C_{OSC1}	15-30 pF	15-30 pF	15-40 pF
C_{OSC2}	15-25 pF	15-25 pF	15-30 pF

Crystal Circuit



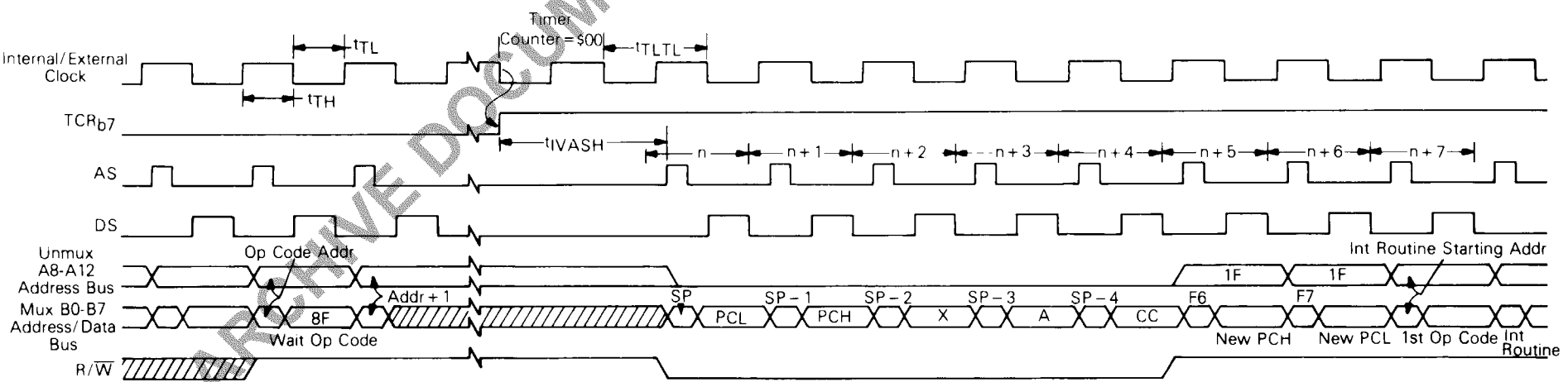
ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

FIGURE 6 — $\overline{IRQ}$ AND $\overline{TCR_7}$ INTERRUPT TIMING



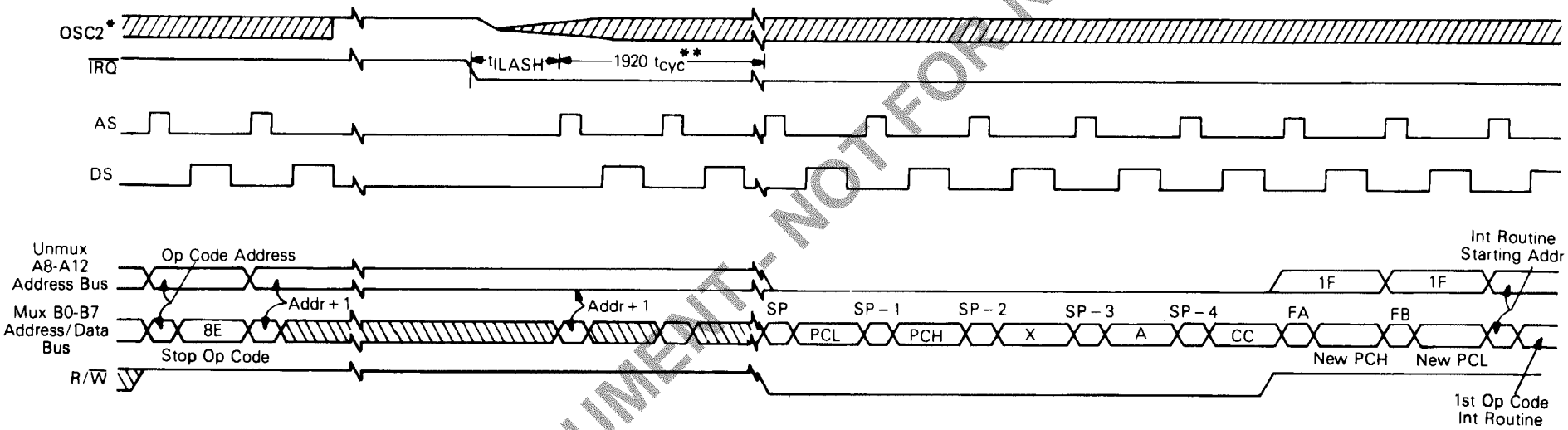
*t_{DSLIH} – The interrupting device must release the $\overline{IRQ}$ line within this time to prevent subsequent recognition of the same interrupt.

FIGURE 7 — TIMER INTERRUPT AFTER WAIT INSTRUCTION: TIMING



ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

FIGURE 8 — INTERRUPT RECOVERY FROM STOP INSTRUCTION: TIMING



* Represents the internal gating of the OSC1 input pin.

** t_{cyc} is one instruction cycle (for f_{osc} = 5 MHz, t_{cyc} = 1 μs)

ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

FUNCTIONAL PIN DESCRIPTION

V_{DD} AND V_{SS}

V_{DD} and V_{SS} provide power to the chip. V_{DD} provides power and V_{SS} is ground.

$\overline{IRQ}$ (MASKABLE INTERRUPT REQUEST)

$\overline{IRQ}$ is both a level-sensitive and edge-sensitive input which can be used to request an interrupt sequence. The MPU completes the current instruction before it responds to the request. If $\overline{IRQ}$ is low and the interrupt mask bit (I bit) in the condition code register is clear, the MPU begins an interrupt sequence at the end of the current instruction. The interrupt circuit recognizes both a "wire ORed" level as well as pulses on the $\overline{IRQ}$ line (see Interrupt section for more details). $\overline{IRQ}$ requires an external resistor to V_{DD} for "wire OR" operation.

$\overline{RESET}$

The $\overline{RESET}$ input is not required for start-up but can be used to reset the MPU internal state and provide an orderly software start-up procedure. Refer to the Reset section for a detailed description.

TIMER

The TIMER input is used for clocking the on-chip timer. Refer to Timer section for a detailed description.

AS (ADDRESS STROBE)

Address strobe (AS) is an output strobe used to indicate the presence of an address on the 8-bit multiplexed bus. The AS line is used to demultiplex the eight least significant address bits from the data bus. A latch controlled by address strobe should capture addresses on the negative edge. This output is capable of driving one standard TTL load and 130 pF and is available at $f_{osc} \div 5$ when the MPU is not in the WAIT or STOP states.

DS (DATA STROBE)

This output is used to transfer data to or from a peripheral or memory. DS occurs anytime the MPU does a data read or write. DS also occurs when the MPU does a data transfer to or from the MPU internal memory. Refer to Table 2 and Figure 4 for timing characteristics. This output is capable of driving one standard TTL load and 130 pF. DS is a continuous signal at $f_{osc} \div 5$ when the MPU is not in the WAIT or STOP state. Some bus cycles are redundant reads of opcode bytes.

R/$\overline{W}$ (READ/WRITE)

The R/$\overline{W}$ output is used to indicate the direction of data transfer for both internal memory and I/O registers, and external peripheral devices and memories. This output is used to indicate to a selected peripheral whether the MPU is going to read or write data on the next data strobe (R/$\overline{W}$ low = processor write; R/$\overline{W}$ high = processor read). The R/$\overline{W}$ output is capable of driving one standard TTL load and 130 pF. The normal standby state is read (high).

A8-A12 (HIGH ORDER ADDRESS LINES)

The A8-A12 output lines constitute the higher order nonmultiplexed addresses. Each output line is capable of driving one standard TTL load and 130 pF.

B0-B7 (ADDRESS/DATA BUS)

The B0-B7 bidirectional lines constitute the lower order addresses and data. These lines are multiplexed, with address present at address strobe time and data present at data strobe time. When in the data mode, these lines are bidirectional, transferring data to and from memory and peripheral devices as indicated by the R/$\overline{W}$ pin. As outputs in either the data or address modes, these lines are capable of driving one standard TTL load and 130 pF.

OSC1, OSC2

The MC146805E2 provides for two types of oscillator inputs – crystal circuit or external clock. The two oscillator pins are used to interface to a crystal circuit, as shown in Figure 5. If an external clock is used, it must be connected to OSC1. The input at these pins is divided by five to form the cycle rate seen on the AS and DS pins. The frequency range is specified by f_{osc}. The OSC1 to bus transitions relationships are provided in Figure 9 for system designs using oscillators slower than 5 MHz.

CRYSTAL – The circuit shown in Figure 5 is recommended when using a crystal. The internal oscillator is designed to interface with an AT-cut parallel resonant quartz crystal resonator in the frequency range specified for f_{osc} in the electrical characteristics table. An external CMOS oscillator is recommended when crystals outside the specified ranges are to be used. The crystal and components should be mounted as close as possible to the input pins to minimize output distortion and start-up stabilization time.

EXTERNAL CLOCK – An external clock should be applied to the OSC1 input with the OSC2 input not connected, as shown in Figure 10.

LI (LOAD INSTRUCTION)

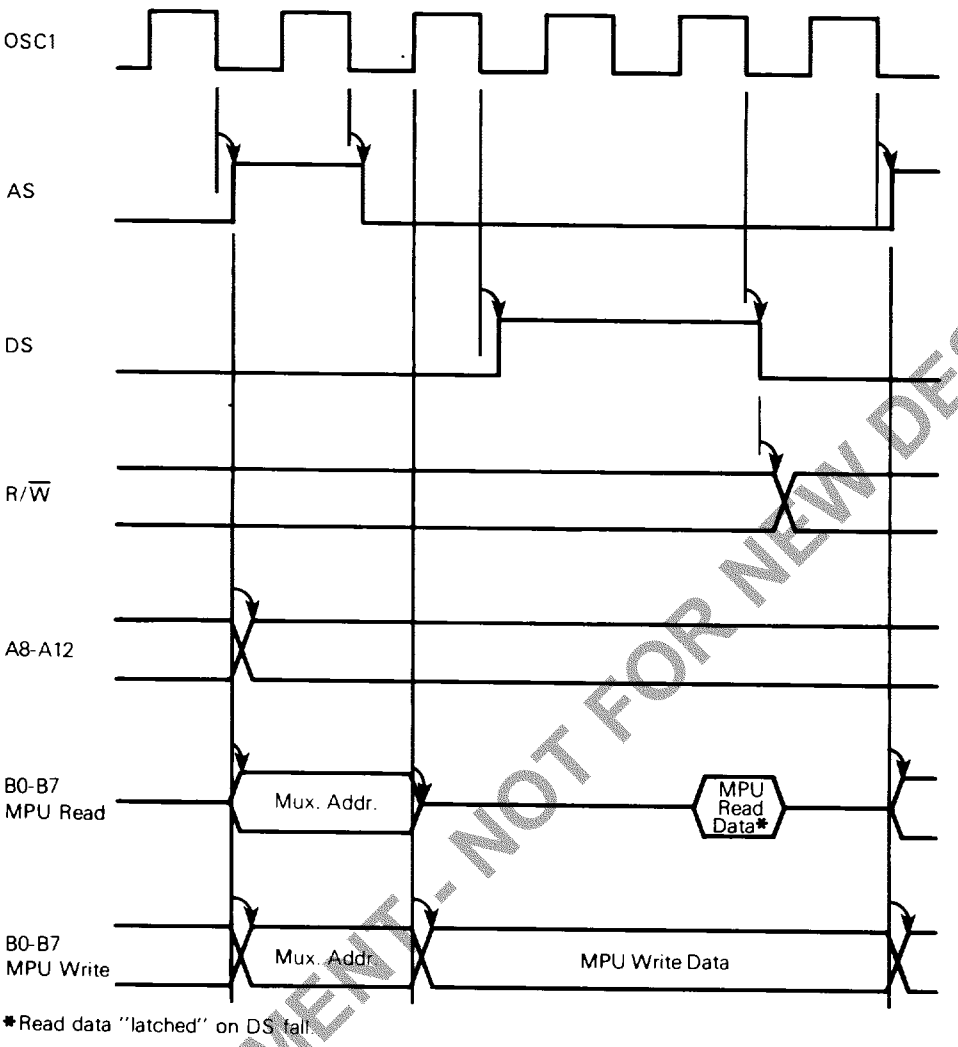
This output is used to indicate that a fetch of the next opcode is in progress. LI remains low during an external or timer interrupt. The LI output is used only for certain debugging and test systems. For normal operations this pin is not connected. The LI output is capable of driving two standard LSTTL loads and 50 pF. This signal overlaps data strobe.

PA0-PA7

These eight pins constitute input/output port A. Each line is individually programmed to be either an input or output under software control via its data direction register as shown in Figure 11(b). An I/O pin is programmed as an output when the corresponding DDR bit is set to a "1", and as an input when it is set to a "0". In the output mode the bits are latched and appear on the corresponding output pins. An MPU read of the port bits programmed as outputs reflects the last value written to that location. When programmed as an input, the input data bit(s) are not latched. An MPU read of the port bits programmed as inputs reflects the current status of the corresponding input pins. The I/O port timing is shown in Figure 3. See typical I/O port circuitry in Figure 11. During a power-on reset or external reset, all lines are configured as inputs (zero in data direction register). The output port register is not initialized by reset. The TTL compatible three-state output buffers are capable of driving one standard TTL load and 50 pF. The DDR is a read/write register.

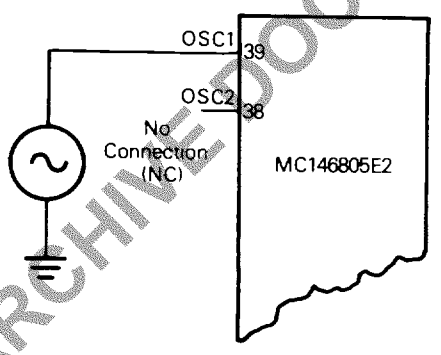
ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

FIGURE 9 — OSC1 TO BUS TRANSITIONS



*Read data "latched" on DS fall.

FIGURE 10 — EXTERNAL CLOCK CONNECTION



ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

PB0-PB7

These eight pins interface with input/output port B. Refer to PA0-PA7 description for details of operation.

MEMORY ADDRESSING

The MC146805E2 is capable of addressing 8192 bytes of memory and I/O registers. The address space is divided into internal memory space and external memory space, as shown in Figure 12.

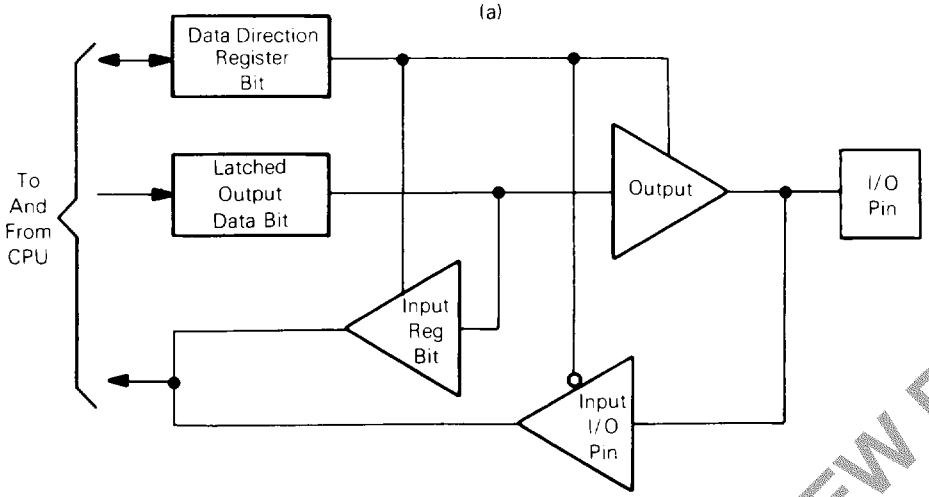
The internal memory space is located within the first 128 bytes of memory (first half of page zero) and is comprised of the I/O port locations, timer locations, and 112 bytes of RAM. The MPU can read from or write to any of these locations. A program write to on-chip locations is repeated on the external bus to permit off-chip memory to duplicate the content of on-chip memory. Program reads to on-chip locations also appear on the external bus, but the MPU accepts data only from the addressed on-chip location. Any read data appearing on the input bus is ignored.

The stack pointer is used to address data stored on the stack. Data is stored on the stack during interrupts and subroutine calls. At power-up, the stack pointer is set to $007F and it is decremented as data is pushed onto the stack. When data is removed from the stack, the stack pointer is incremented. A maximum of 64 bytes of RAM is available for stack usage. Since most programs use only a small part of the allotted stack locations for interrupts and/or subroutine stacking purposes, the unused bytes are usable for program data storage.

All memory locations above location $007F are part of the external memory map. In addition, ten locations in the I/O portion of the lower 128 bytes of memory space, as shown in

FIGURE 11 — TYPICAL PORT I/O CIRCUITRY

(a)



(b)

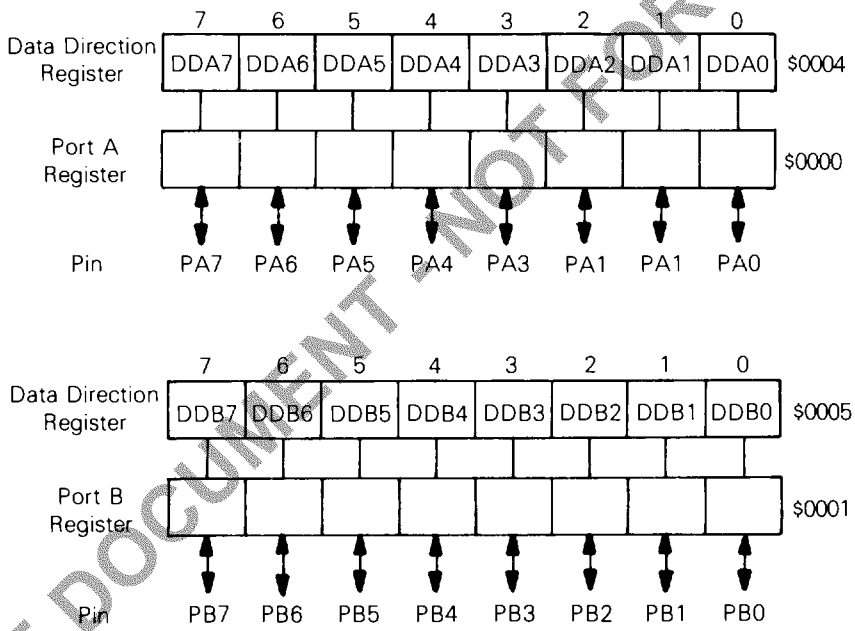


TABLE 3 — I/O PIN FUNCTIONS

R/$\overline{W}$	DDR	I/O Pin Functions
0	0	The I/O pin is in input mode. Data is written into the output data latch.
0	1	Data is written into the output data latch and output to the I/O pin.
1	0	The state of the I/O pin is read.
1	1	The I/O pin is in an output mode. The output data latch is read.

ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

Figure 12, are part of the external memory map. All of the external memory space is user definable except the highest 10 locations. Locations $1FF6 to $1FFF of the external address space are reserved for interrupt and reset vectors (see Figure 12).

REGISTERS

The MC146805E2 contains five registers as shown in the programming model in Figure 13. The interrupt stacking order is shown in Figure 14.

ACCUMULATOR (A)

This accumulator is an 8-bit general purpose register used to hold operands and results of arithmetic calculations and data manipulations.

INDEX REGISTER (X)

The X register is an 8-bit register which is used during the indexed modes of addressing. It provides an 8-bit value which is used to create an effective address. The index register is also used for data manipulations with the read-modify-write type of instructions and as a temporary storage register when not performing addressing operations.

PROGRAM COUNTER (PC)

The program counter is a 13-bit register that contains the address of the next instruction to be executed by the processor.

ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

FIGURE 12 — MPU ADDRESS MAP

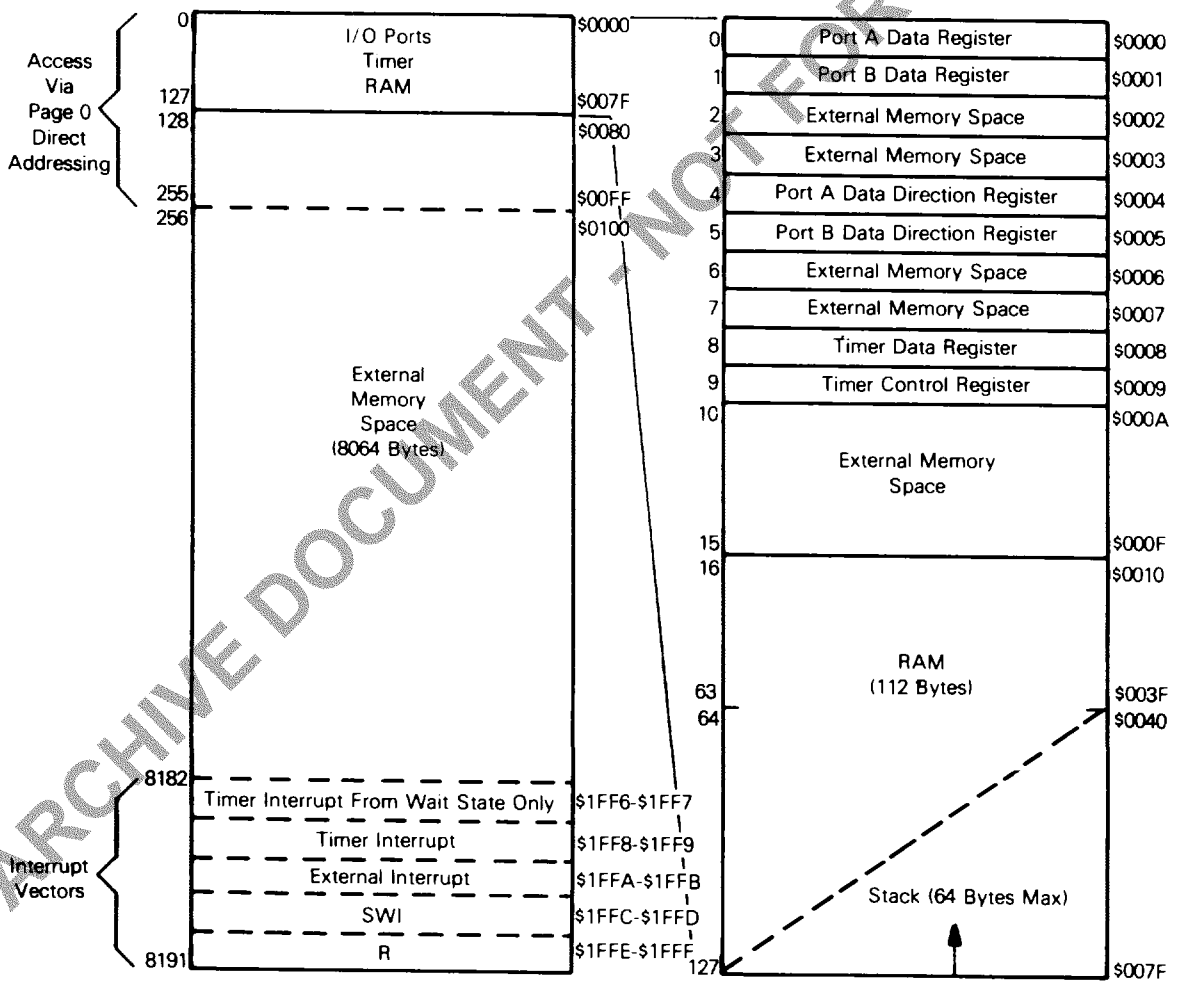


FIGURE 13 — PROGRAMMING MODEL

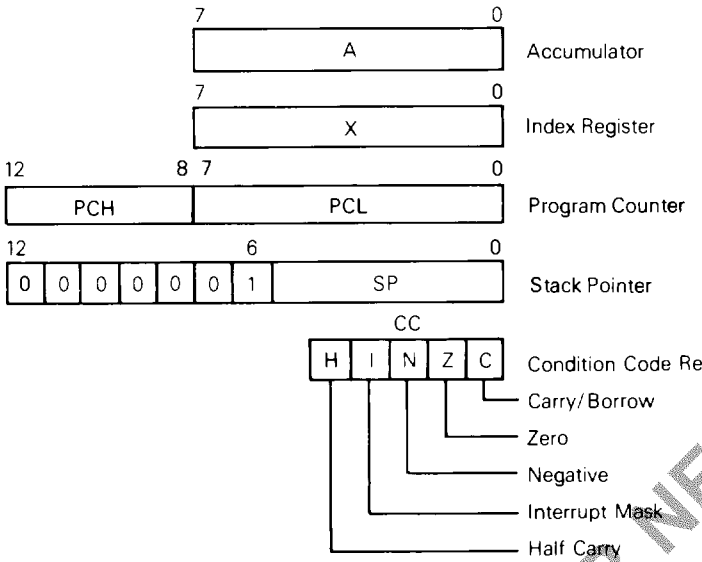
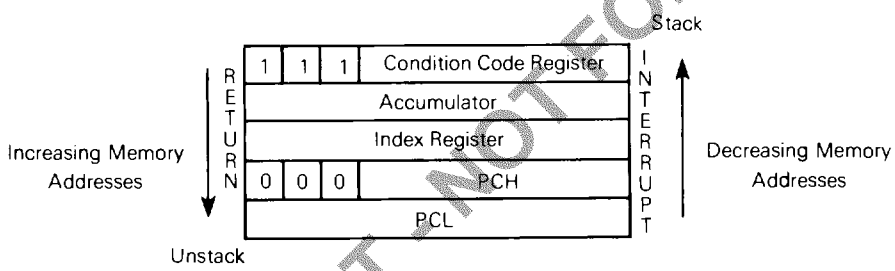


FIGURE 14 — STACKING ORDER



NOTE: Since the stack pointer decrements during pushes, the PCL is stacked first, followed by PCH, etc. Pulling from the stack is in the reverse order.

ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

STACK POINTER (SP)

The stack pointer is a 13-bit register containing the address of the next free location on the stack. When accessing memory, the seven most significant bits are permanently set to 0000001. They are appended to the six least significant register bits to produce an address within the range of $007F to $0040. The stack area of RAM is used to store the return address on subroutine calls and the machine state during interrupts. During external or power-on reset, and during a "reset stack pointer" instruction, the stack pointer is set to its upper limit ($007F). Nested interrupts and/or subroutines may use up to 64 (decimal) locations, beyond which the stack pointer "wraps around" and points to its upper limit, thereby losing the previously stored information. A subroutine call occupies two RAM bytes on the stack, while an interrupt uses five bytes.

CONDITION CODE REGISTER (CC)

The condition code register is a 5-bit register in which each bit is used to indicate the results of the instruction just executed. These bits can be individually tested by a program and specific action taken as a result of their state. Each of the five bits is explained below.

HALF CARRY BIT (H) — The H bit is set to a one when a carry occurs between bits 3 and 4 of the ALU during an ADD or ADC instruction. The H bit is useful in binary coded decimal addition subroutines.

INTERRUPT MASK BIT (I) — When the I bit is set, both the external interrupt and the timer interrupt are disabled. Clearing this bit enables the above interrupts. If an interrupt occurs while the I bit is set, the interrupt is latched and will be processed when the I bit is next cleared.

NEGATIVE BIT (N) — When set, this bit indicates that the result of the last arithmetic, logical, or data manipulation was negative (bit 7 in the result is a logical one).

ZERO BIT (Z) — When set, this bit indicates that the result of the last arithmetic, logical, or data manipulation was zero.

CARRY BIT (C) — The C bit is set when a carry or a borrow out of the ALU occurs during an arithmetic instruction. The C bit is also modified during bit test, shift, rotate, and branch types of instruction.

RESETS

The MC146805E2 has two reset modes: an active low external reset pin ($\overline{\text{RESET}}$) and a power-on reset function; refer to Figure 5.

$\overline{\text{RESET}}$ (PIN #1)

The $\overline{\text{RESET}}$ input pin is used to reset the MPU and provide an orderly software start-up procedure. When using the external reset mode, the $\overline{\text{RESET}}$ pin must stay low for a minimum of one t_{RL}. The $\overline{\text{RESET}}$ pin is provided with a Schmitt trigger to improve its noise immunity capability.

POWER-ON RESET

The power-on reset occurs when a positive transition is detected on V_{DD}. The power-on reset is used strictly for power turn-on conditions and should not be used to detect any drops in the power supply voltage. There is no provision for a power-down reset. The power-on circuitry provides for a 1920 t_{cyc} delay from the time of the first oscillator operation. If the external reset pin is low at the end of the 1920 t_{cyc} time out, the processor remains in the reset condition.

Either of the two types of reset conditions causes the following to occur:

- Timer control register interrupt request bit (bit 7) is cleared to a "0".
- Timer control register interrupt mask bit (bit 6) is set to a "1".
- All data direction register bits are cleared to a "0" (inputs).
- Stack pointer is set to $007F.
- The address bus is forced to the reset vector ($1FFE, $1FFF).
- Condition code register interrupt mask bit (I) is set to a "1".
- STOP and WAIT latches are reset.
- External interrupt latch is reset.

All other functions, such as other registers (including output ports), the timer, etc., are not cleared by the reset conditions.

INTERRUPTS

The MC146805E2 may be interrupted by one of three different methods: either one of two maskable hardware interrupts (external input or timer) or a non-maskable software interrupt (SWI). Systems often require that normal processing be interrupted so that some external event may be serviced.

Interrupts cause the processor registers to be saved on the stack and the interrupt mask set to prevent additional interrupts. The RTI instruction causes the register contents to be recovered from the stack and a return to normal processing. The stacking order is shown in Figure 14.

Unlike $\overline{\text{RESET}}$, hardware interrupts do not cause the current instruction excutiorr to be halted, but are considered pending until the current instruction execution is complete.

When the current instruction is complete, the processor checks all pending hardware interrupts and if unmasked, proceeds with interrupt processing; otherwise, the next instruction is fetched and executed. Note that masked interrupts are latched for later interrupt service.

If both an external interrupt and a timer interrupt are pending at the end of an instruction execution, the external interrupt is serviced first. The SWI is executed as any other instruction. Refer to Figure 15 for the interrupt and instruction processing sequence.

TIMER INTERRUPT

If the timer mask bit (TCR6) is cleared, then each time the timer decrements to zero (transitions from $01 to $00) an interrupt request is generated. The actual processor interrupt is generated only if the interrupt mask bit of the condition code register is also cleared. When the interrupt is recognized, the current state of the machine is pushed onto the stack and the I bit in the condition code register is set. This masks further interrupts until the present one is serviced. The processor now vectors to the timer interrupt service routine. The address for this service routine is specified by the contents of $1FF8 and $1FF9 unless the processor is in a WAIT mode, in which case users of mask versions BP4XX-XX and AW9XXXX should refer to the appendix for additional information regarding exceptions to this function. The contents of $1FF6 and $1FF7 specify the service routine. Also, software must be used to clear the timer interrupt request bit (TCR7). At the end of the timer interrupt service routine, the software normally executes an RTI instruction which restores the machine state and starts executing the interrupted program.

EXTERNAL INTERRUPT

If the interrupt mask bit of the condition code register is cleared and the external interrupt pin $\overline{\text{IRQ}}$ is "low," then the external interrupt occurs. The action of the external interrupt is identical to the timer interrupt with the exception that the service routine address is specified by the contents of $1FFA and $1FFB. The interrupt logic recognizes both a "wire ORed" level and pulses on the external interrupt line. Figure 16 shows both a functional diagram and timing for the interrupt line. The timing diagram shows two different treatments of the interrupt line ($\overline{\text{IRQ}}$) to the processor. The first configuration shows many interrupt lines "wire ORed" to form the interrupts at the processor. Thus, if after servicing an interrupt the $\overline{\text{IRQ}}$ remains low, then the next interrupt is recognized. The second method is single pulses on the interrupt line spaced far enough apart to be serviced. Users of mask versions BP4XXXX and AW9XXXX should refer to the appendix regarding exceptions to this function. The minimum time between pulses is a function of the length of the interrupt service routine. Once a pulse ocurs, the next pulse should not occur until the MPU software has exited the routine (an RTI occurs). This time (t_{ILIL}) is obtained by adding 20 instruction cycles (one cycle $t_{cyc} = 5/f_{osc}$) to the total number of cycles it takes to complete the service routine including the RTI instruction; refer to Figure 6.

SOFTWARE INTERRUPT (SWI)

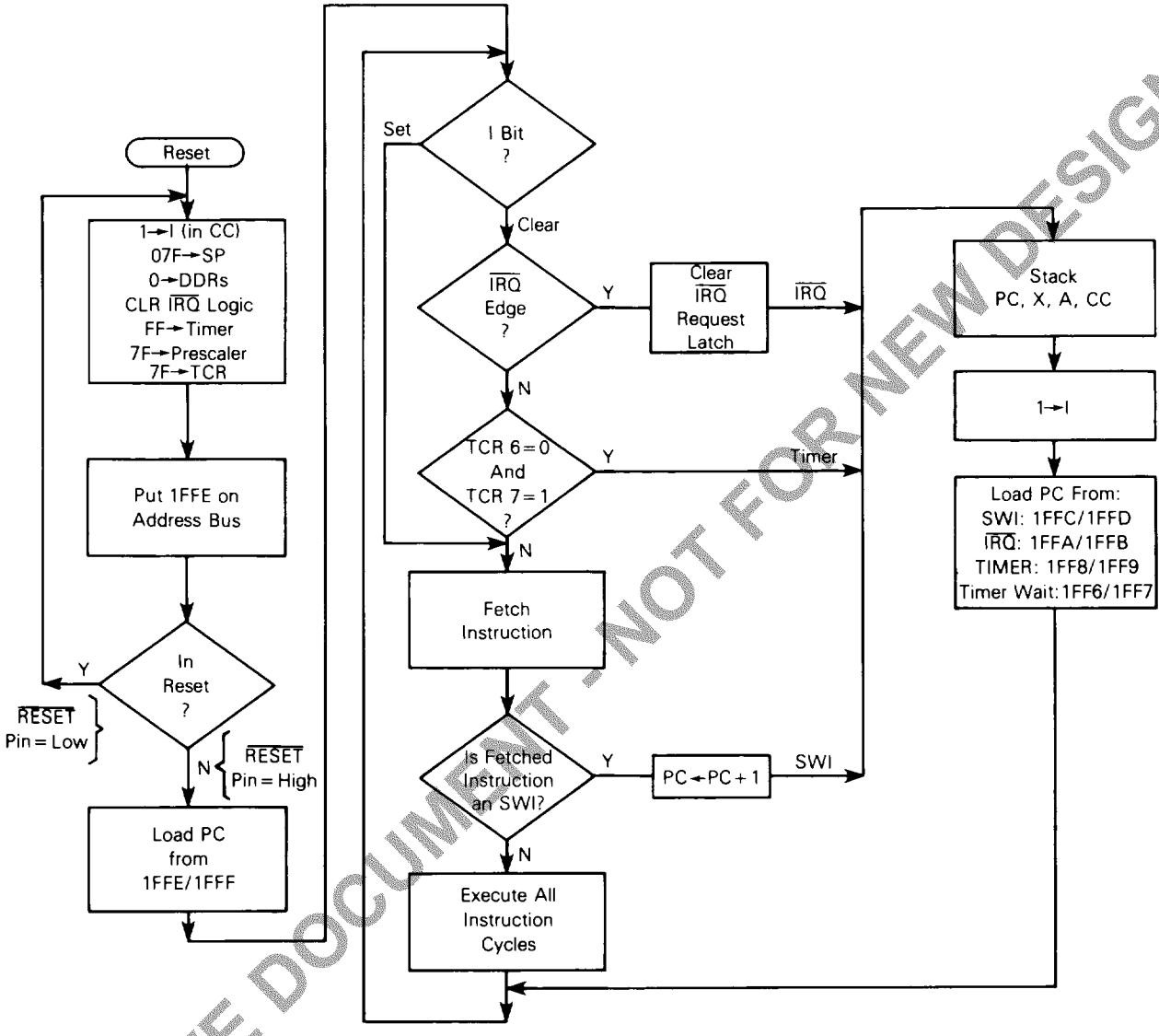
The software interrupt is an executable instruction. The action of the SWI instruction is similar to the hardware interrupts. The SWI is executed regardless of the state of the interrupt mask in the condition code register. The service routine address is specified by the contents of memory locations $1FFC and $1FFD. See Figure 15 for interrupt and instruction processing flowchart.

STOP

The STOP instruction places the MC146805E2 in a low power consumption mode. In the STOP function the internal oscillator is turned off, causing all internal processing and the timer to be halted; refer to Figure 17. The DS and AS lines go to a low state and the R/$\overline{\text{W}}$ line goes to a high state.

ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

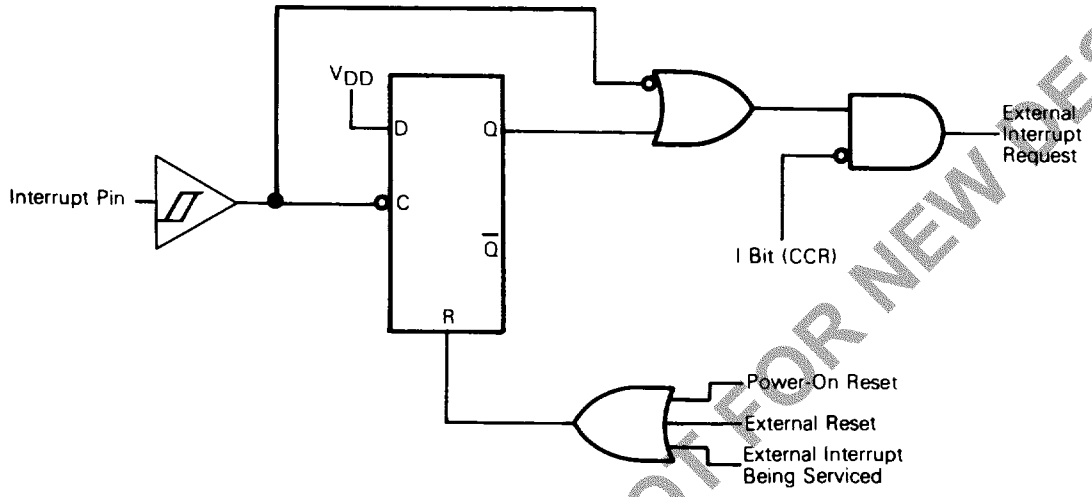
FIGURE 15 — $\overline{\text{RESET}}$ AND INTERRUPT PROCESSING FLOWCHART



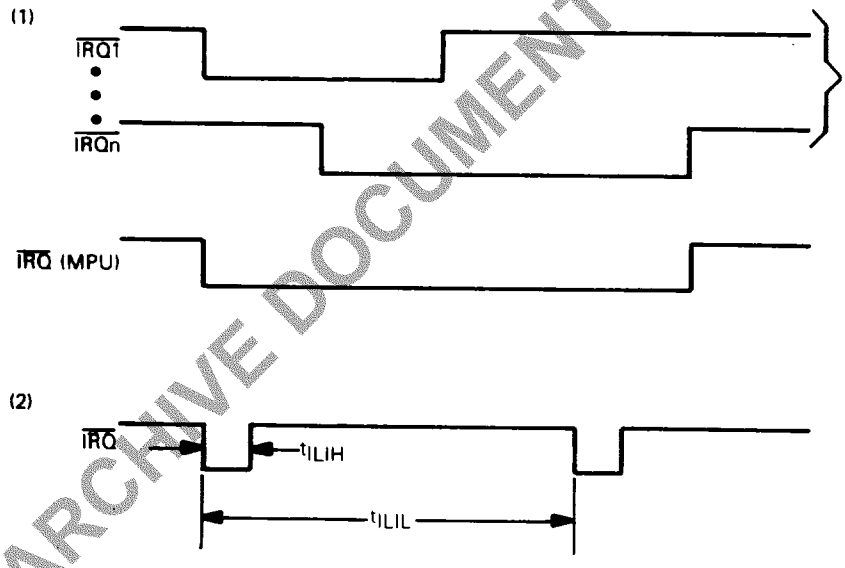
ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

FIGURE 16 — EXTERNAL INTERRUPT

(a) Interrupt Functional Diagram



(b) Interrupt Mode Diagram



Wire ORed Condition

If after servicing an interrupt the $\overline{IRQ}$ remains low, then the next interrupt is recognized.

Pulse Condition

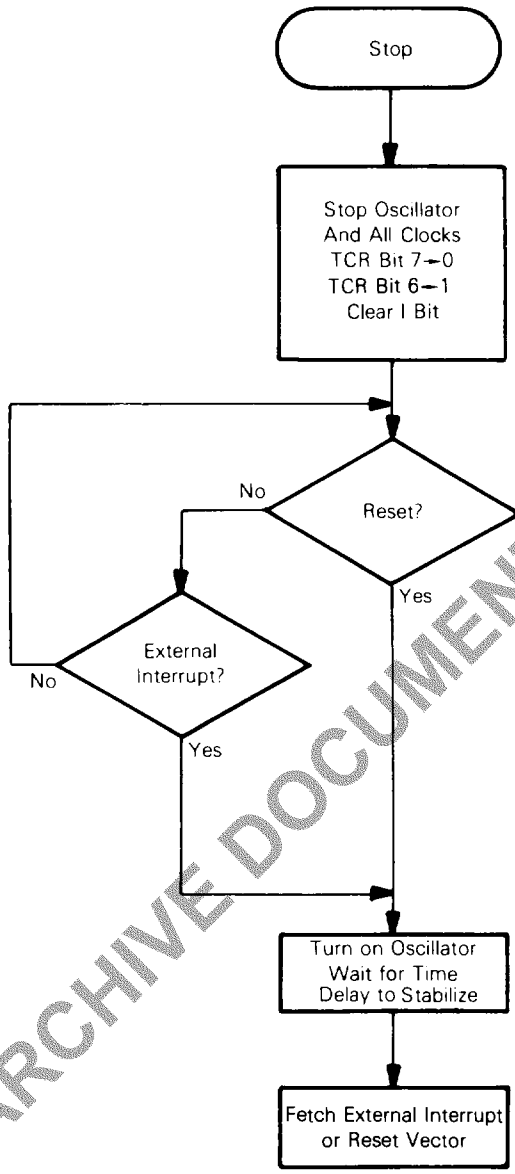
The minimum pulse width (t_{ILIH}) is one t_{cyc}. The period t_{ILIL} should not be less than the number of t_{cyc} cycles it takes to execute the interrupt service routine plus 20 t_{cyc} cycles.

ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

The multiplexed address/data bus goes to the data input state (as shown in Figure 8). The high order address lines remain at the address of the next instruction. The MPU remains in the STOP mode until an external interrupt or reset occurs.

During the STOP mode, timer control register (TCR) bits 6 and 7 are altered to remove any pending timer interrupt requests and to disable any further timer interrupts. External interrupts are enabled in the condition code register. All other registers and memory remain unaltered. All I/O lines remain unchanged.

FIGURE 17 — STOP FUNCTION FLOWCHART



WAIT

The WAIT instruction places the MC146805E2 in a low power consumption mode, but the WAIT mode consumes somewhat more power than the STOP mode; refer to Table 1. In the WAIT function, the internal clock is disabled from all internal circuitry except the timer circuit; refer to Figure 18. Thus, all internal processing is halted except the timer which is allowed to count in a normal sequence. The R/$\overline{W}$ line goes to a high state, the multiplexed address/data bus goes to the data input state, and the DS and AS lines go to the low state (as shown in Figure 7). The high order address lines remain at the address of the next instruction. The MPU remains in this state until an external interrupt, timer interrupt, or a reset occurs.

During the WAIT mode, the I bit in the condition code register is cleared to enable interrupts. All other registers, memory, and I/O lines remain in their last state. The timer may be enabled to allow a periodic exit from the WAIT mode. If an external and a timer interrupt occur at the same time, the external interrupt is serviced first; then, if the timer interrupt request is not cleared in the external interrupt routine, the normal timer interrupt (not the timer WAIT interrupt) is serviced since the MPU is no longer in the WAIT mode.

TIMER

The MPU timer contains a single 8-bit software programmable counter (timer data register) with 7-bit software selectable prescaler. Figure 19 shows a block diagram of the timer. The counter may be preset under program control and decrements towards zero. When the counter decrements to zero, the timer interrupt request bit, i.e., bit 7 of the timer control register (TCR), is set. Then if the timer interrupt is not masked, i.e., bit 6 of the TCR and the I bit in the condition code register are both cleared, the processor receives an interrupt. After completion of the current instruction, the processor proceeds to store the appropriate registers on the stack, and then fetches the timer interrupt vector from locations $1FF8 and $1FF9 in order to begin servicing the interrupt. If the MPU is interrupted by the timer while in the WAIT mode, the interrupt vector fetch would be from locations $1FF6 and $1FF7.

The counter continues to count after it reaches zero, allowing the software to determine the number of internal or external input clocks since the timer interrupt request bit was set. The counter may be read at any time by the processor without disturbing the count. The content of the counter becomes stable prior to the read portion of a cycle and does not change during the read. The timer interrupt request bit remains set until cleared by the software. If a read occurs before the timer interrupt is serviced, the interrupt is lost. TCR7 may also be used as a scanned status bit in a non-interrupt mode of operation (TCR6 = 1).

The prescaler is a 7-bit divider which is used to extend the maximum length of the timer. Bit 0, bit 1, and bit 2 of the TCR are programmed to choose the appropriate prescaler output which is used as the counter input. The processor cannot write into or read from the prescaler; however, its contents are cleared to all "0s" by the write operation into TCR when bit 3 of the written data equals 1, which allows for truncation-free counting.

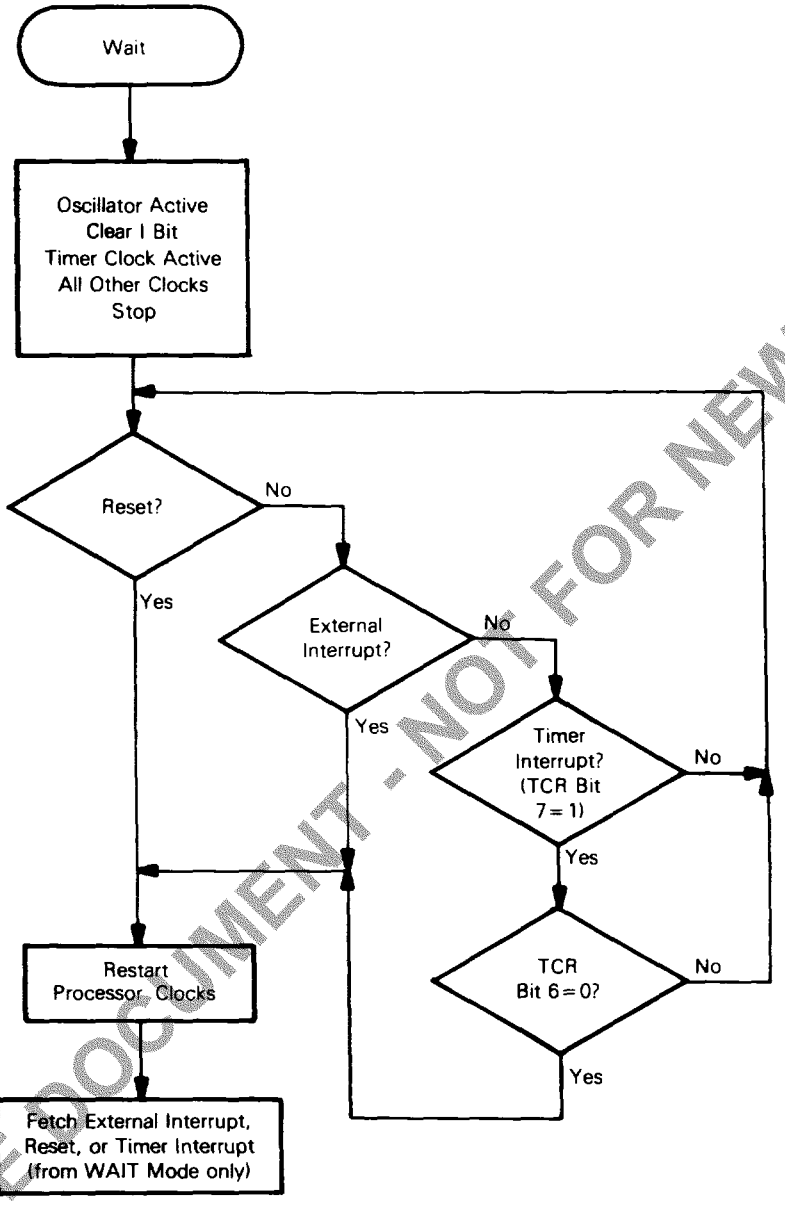
The timer input can be configured for three different operating modes, plus a disable mode, depending on the value written to the TCR4, TCR5 control bits. Refer to the Timer Control Register section.

TIMER INPUT MODE 1

If TCR4 and TCR5 are both programmed to a "0", the input to the timer is from an internal clock and the external TIMER input is disabled. The internal clock mode can be

ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

FIGURE 18 — WAIT FUNCTION FLOWCHART



ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

used for periodic interrupt generation, as well as a reference in frequency and event measurement. The internal clock is the instruction cycle clock and is coincident with address strobe (AS) except during a WAIT instruction. During a WAIT instruction the AS pin goes to a low state but the internal clock to the timer continues to run at its normal rate.

TIMER INPUT MODE 2

With TCR4=1 and TCR5=0, the internal clock and the TIMER input pin are ANDed to form the timer input signal. This mode can be used to measure external pulse widths. The external timer input pulse simply turns on the internal clock for the duration of the pulse. The resolution of the count in this mode is ±1 clock and therefore accuracy improves with longer input pulse widths.

TIMER INPUT MODE 3

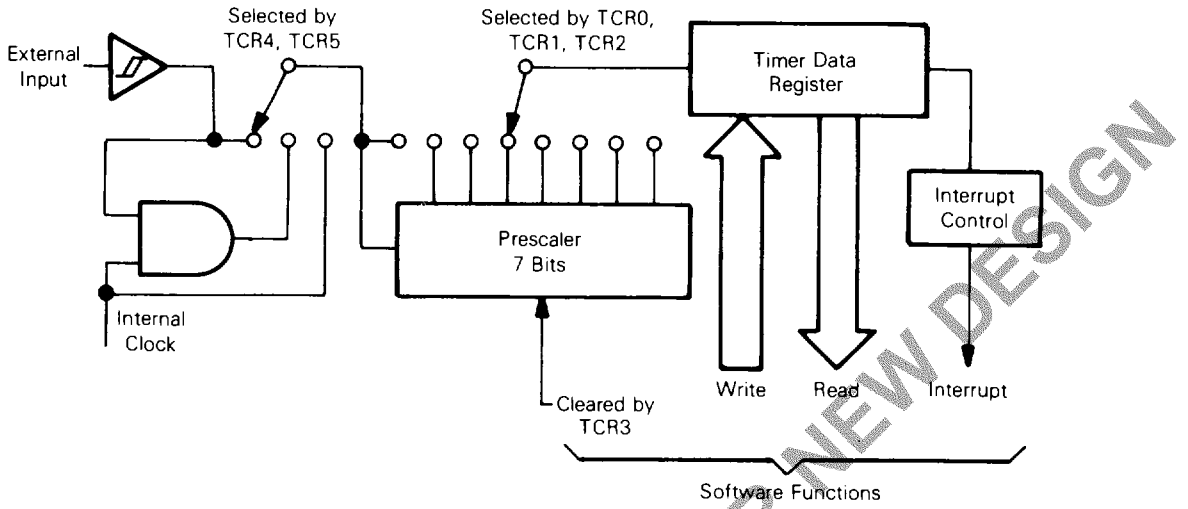
If TCR4=0 and TCR5=1, then all inputs to the timer are disabled.

TIMER INPUT MODE 4

If TCR4=1 and TCR5=1, the internal clock input to the timer is disabled and the TIMER input pin becomes the input to the timer. The external TIMER pin can, in this mode, be used to count external events as well as external frequencies for generating periodic interrupts.

Figure 19 shows a block diagram of the timer subsystem.

FIGURE 19 — TIMER BLOCK DIAGRAM



NOTES:
1. Prescaler and timer data register are clocked on the falling edge of the internal clock (AS) or external input.
2. Timer data register is written to during data strobe (DS) and counts down continuously.

TIMER CONTROL REGISTER (TCR)

7	6	5	4	3	2	1	0
TCR7	TCR6	TCR5	TCR4	TCR3	TCR2	TCR1	TCR0

All bits in this register except bit 3 are read/write bits.

TCR5	TCR4	
0	0	Internal clock (AS) to timer
0	1	AND of internal clock (AS) and TIMER pin to timer
1	0	Inputs to timer disabled
1	1	TIMER pin to timer

TCR7 — Timer interrupt request bit: bit used to indicate the timer interrupt when it is logic "1".
- 1 — Set whenever the counter decrements to zero, or under program control.
- 0 — Cleared on external reset, power-on reset, STOP instruction, or program control.

TCR6 — Timer interrupt mask bit: when this bit is a logic "1" it inhibits the timer interrupt to the processor.
- 1 — Set on external reset, power-on reset, STOP instruction, or program control.
- 0 — Cleared under program control.

TCR5 — External or internal bit: selects the input clock source to be either the external TIMER pin or the internal clock (unaffected by $\overline{\text{RESET}}$).
- 1 — Select external clock source.
- 0 — Select internal clock source (AS).

TCR4 — External enable bit: control bit used to enable the external TIMER pin (unaffected by $\overline{\text{RESET}}$).
- 1 — Enable external TIMER pin.
- 0 — Disable external TIMER pin.

TCR3 — Timer Prescaler Reset bit: writing a "1" to this bit resets the prescaler to zero. A read of this location always indicates a "0" (unaffected by $\overline{\text{RESET}}$).

TCR2, TCR1, TCR0 — Prescaler address bits: decoded to select one of eight outputs of the prescaler (unaffected by $\overline{\text{RESET}}$).

Prescaler

TCR2	TCR1	TCR0	Result
0	0	0	÷1
0	0	1	÷2
0	1	0	÷4
0	1	1	÷8
1	0	0	÷16
1	0	1	÷32
1	1	0	÷64
1	1	1	÷128

INSTRUCTION SET

The MPU has a set of 61 basic instructions. They can be divided into five different types: register/memory, read-modify-write, branch, bit manipulation, and control. The following paragraphs briefly explain each type. All the instructions within a given type are presented in individual tables.

ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

REGISTER/MEMORY INSTRUCTIONS

Most of these instructions use two operands. One operand is either the accumulator or the index register. The other operand is obtained from memory using one of the addressing modes. The jump unconditional (JMP) and jump to subroutine (JSR) instructions have no register operand. Refer to Table 4.

READ-MODIFY-WRITE INSTRUCTIONS

These instructions read a memory location or a register, modify or test its contents, and write the modified value back to memory or to the register. The test for negative or zero (TST) instruction is an exception to the read-modify-write sequence since it does not modify the value. Refer to Table 5.

BRANCH INSTRUCTIONS

This set of instructions branches if a particular condition is met, otherwise no operation is performed. Branch instructions are two byte instructions. Refer to Table 6.

BIT MANIPULATION INSTRUCTIONS

The MPU is capable of setting or clearing any bit which resides in the first 256 bytes of the memory space, where all port registers, port DDRs, timer, timer control, and on-chip RAM reside. An additional feature allows the software to test and branch on the state of any bit within these 256 locations. The bit set, bit clear and bit test, and branch functions are all implemented with a single instruction. For the test and branch instructions, the value of the bit tested is also placed in the carry bit of the condition code register. Refer to Table 7.

CONTROL INSTRUCTIONS

These instructions are register reference instructions and are used to control processor operation during program execution. Refer to Table 8.

ALPHABETICAL LISTING

The complete instruction set is given in alphabetical order in Table 9.

OPCODE MAP SUMMARY

Table 10 is an opcode map for the instructions used on the MCU.

ADDRESSING MODES

The MPU uses ten different addressing modes to provide the programmer with an opportunity to optimize the code to all situations. The various indexed addressing modes make it possible to locate data tables, code conversion tables, and scaling tables anywhere in the memory space. Short indexed accesses are single byte instructions, while the longest instructions (three bytes) permit accessing tables throughout memory. Short and long absolute addressing is also included. Two byte direct addressing instructions access all data bytes in most applications. Extended addressing permits jump instructions to reach all memory. Table 9 shows the addressing modes for each instruction, with the effects each instruction has on the condition code register. An opcode map is shown in Table 10.

The term "effective address" or EA is used in describing the various addressing modes, and is defined as the address to or from which the argument for an instruction is fetched or stored. The ten addressing modes of the processor are described below. Parentheses are used to indicate "contents of," an arrow indicates "is replaced by," and a colon indicates concatenation of two bytes.

INHERENT

In inherent instructions all the information necessary to execute the instruction is contained in the opcode. Operations specifying only the index register or accumulator, and no other arguments, are included in this mode.

IMMEDIATE

In immediate addressing, the operand is contained in the byte immediatley following the opcode. Immediate addressing is used to access constants which do not change during program execution (e.g., a constant used to initialize a loop counter).

$$EA = PC + 1;\ PC \leftarrow PC + 2$$

DIRECT

In the direct addressing mode, the effective address of the argument is contained in a single byte following the opcode byte. Direct addressing allows the user to directly address the lowest 256 bytes in memory with a single two byte instruction. This includes all on-chip RAM and I/O registers and up to 128 bytes of off-chip ROM. Direct addressing is efficient in both memory and speed.

$$EA = (PC + 1);\ PC \leftarrow PC + 2$$

Address Bus High $\leftarrow$ 0; Address Bus Low $\leftarrow$ (PC + 1)

EXTENDED

In the extended addressing mode, the effective address of the argument is contained in the two bytes following the opcode. Instructions with extended addressing modes are capable of referencing arguments anywhere in memory with a single three byte instruction. When using the Motorola assembler, the user need not specify whether an instruction uses direct or extended addressing. The assembler automatically selects the most efficient addressing mode.

$$EA = (PC + 1):(PC + 2);\ PC \leftarrow PC + 3$$

Address Bus High $\leftarrow$ (PC + 1); Address Bus Low $\leftarrow$ (PC + 2)

INDEXED, NO OFFSET

In the indexed, no offset addressing mode, the effective address of the argument is contained in the 8-bit index register. Thus, this addressing mode can access the first 256 memory locations. These instructions are only one byte long. This mode is used to move a pointer through a table or to address a frequently referenced RAM or I/O location.

$$EA = X;\ PC \leftarrow PC + 1$$

Address Bus High $\leftarrow$ 0, Address Bus Low $\leftarrow$ X

INDEXED, 8-BIT OFFSET

Here the EA is obtained by adding the contents of the byte following the opcode to that of the index register; therefore, the operand is located anywhere within the lowest 511 memory locations. For example, this mode of addressing is useful for selecting the m-th element in an n element table. All instructions are two bytes. The contents of the index register (X) is not changed. The contents of (PC + 1) is an unsigned 8-bit integer. One byte offset indexing permits look-up tables to be easily accessed in either RAM or ROM.

$$EA = X + (PC + 1);\ PC \leftarrow PC + 2$$

Address Bus High $\leftarrow$ K; Address Bus Low $\leftarrow$ X + (PC + 1)

where: K = The carry from the addition of X + (PC + 1)

ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

INDEXED, 16-BIT OFFSET

In the indexed, 16-bit offset addressing mode the effective address is the sum of the contents of the unsiged 8-bit index register and the two unsigned bytes following the opcode. This addressing mode can be used in a manner similar to indexed 8-bit offset, except that this three byte instruction allows tables to be anywhere in memory (e.g., jump tables in ROM). As with direct and extended, the M6805 assembler determines the most efficient form of indexed offset — 8 or 16 bit. The content of the index register is not changed.

EA = X + [(PC + 1):(PC + 2)]; PC ← PC + 3
Address Bus High ← (PC + 1) + K
Address Bus Low ← K + (PC + 2)
where: K = The carry from the addition of X + (PC + 2)

RELATIVE

Relative addressing is used only in branch instructions. In relative addressing the content of the 8-bit signed byte following the opcode (the offset) is added to the PC if and only if the branch condition is true. Otherwise, control proceeds to the next instruction. The span of relative addressing is limited to the range of −126 to +129 bytes from the branch instruction opcode location. The Motorola assembler calculates the proper offset and checks to see if it is within the span of the branch.

EA = PC + 2 + (PC + 1); PC ← EA if branch is taken;
otherwise, PC ← PC + 2

BIT SET/CLEAR

Direct addressing and bit addressing are combined in instructions which set and clear individual memory and I/O bits. In the bit set and clear instructions, the byte is specified as a direct address in the location following the opcode. The first 256 addressable locations are thus accessed. The bit to be modified within that byte is specified with three bits of the opcode. The bit set and clear instructions occupy two bytes, one for the opcode (including the bit number) and the second to address the byte which contains the bit of interest.

EA = (PC + 1); PC ← PC + 2
Address Bus High ← 0; Address Bus Low ← (PC + 1)

BIT TEST AND BRANCH

Bit test and branch is a combination of direct addressing, bit addressing, and relative addressing. The bit address and condition (set or clear) to be tested are part of the opcode. The address of the byte to be tested is in the single byte immediately following the opcode byte (EA1). The signed relative 8-bit offset is in the third byte (EA2) and is added to the PC if the specified bit is set or clear in the specified memory location. This single three byte instruction allows the program to branch based on the condition of any bit in the first 256 locations of memory.

EA1 = (PC + 1)
Address Bus High ← 0; Address Bus Low ← (PC + 1)
EA2 = PC + 3 + (PC + 2); PC ← EA2 if branch taken;
otherwise, PC ← PC + 3

SYSTEM CONFIGURATION

Figures 20 through 25 show in general terms how the MC146805E2 bus structure may be utilized. Specified interface details vary with the various peripheral and memory devices employed.

Table 11 provides a detailed description of the information present on the bus, read/write (R/$\overline{W}$) pin and the load instruction (LI) pin during each cycle for each instruction.

This information is useful in comparing actual with expected results during debug of both software and hardware as the control program is executed. The information is categorized in groups according to addressing mode and number of cycles per instruction.

ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

TABLE 4 — REGISTER/MEMORY INSTRUCTIONS

		Addressing Modes																	
		Immediate			Direct			Extended			Indexed (No Offset)			Indexed (8-Bit Offset)			Indexed (16-Bit Offset)		
Function	Mnemonic	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles
Load A from Memory	LDA	A6	2	2	B6	2	3	C6	3	4	F6	1	3	E6	2	4	D6	3	5
Load X from Memory	LDX	AE	2	2	BE	2	3	CE	3	4	FE	1	3	EE	2	4	DE	3	5
Store A in Memory	STA	—	—	—	B7	2	4	C7	3	5	F7	1	4	E7	2	5	D7	3	6
Store X in Memory	STX	—	—	—	BF	2	4	CF	3	5	FF	1	4	EF	2	5	DF	3	6
Add Memory to A	ADD	AB	2	2	BB	2	3	CB	3	4	FB	1	3	EB	2	4	DB	3	5
Add Memory and Carry to A	ADC	A9	2	2	B9	2	3	C9	3	4	F9	1	3	E9	2	4	D9	3	5
Subtract Memory	SUB	A0	2	2	B0	2	3	C0	3	4	F0	1	3	E0	2	4	D0	3	5
Subtract Memory from A with Borrow	SBC	A2	2	2	B2	2	3	C2	3	4	F2	1	3	E2	2	4	D2	3	5
AND Memory to A	AND	A4	2	2	B4	2	3	C4	3	4	F4	1	3	E4	2	4	D4	3	5
OR Memory with A	ORA	AA	2	2	BA	2	3	CA	3	4	FA	1	3	EA	2	4	DA	3	5
Exclusive OR Memory with A	EOR	A8	2	2	B8	2	3	C8	3	4	F8	1	3	E8	2	4	D8	3	5
Arithmetic Compare A with Memory	CMP	A1	2	2	B1	2	3	C1	3	4	F1	1	3	E1	2	4	D1	3	5
Arithmetic Compare X with Memory	CPX	A3	2	2	B3	2	3	C3	3	4	F3	1	3	E3	2	4	D3	3	5
Bit Test Memory with A (Logical Compare)	BIT	A5	2	2	B5	2	3	C5	3	4	F5	1	3	E5	2	4	D5	3	5
Jump Unconditional	JMP	—	—	—	BC	2	2	CC	3	3	FC	1	2	EC	2	3	DC	3	4
Jump to Subroutine	JSR	—	—	—	BD	2	5	CD	3	6	FD	1	5	ED	2	6	DD	3	7

TABLE 5 — READ-MODIFY-WRITE INSTRUCTIONS

		Addressing Modes														
		Inherent (A)			Inherent (X)			Direct			Indexed (No Offset)			Indexed (8-Bit Offset)		
Function	Mnemonic	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles
Increment	INC	4C	1	3	5C	1	3	3C	2	5	7C	1	5	6C	2	6
Decrement	DEC	4A	1	3	5A	1	3	3A	2	5	7A	1	5	6A	2	6
Clear	CLR	4F	1	3	5F	1	3	3F	2	5	7F	1	5	6F	2	6
Complement	COM	43	1	3	53	1	3	33	2	5	73	1	5	63	2	6
Negate (2's Complement)	NEG	40	1	3	50	1	3	30	2	5	70	1	5	60	2	6
Rotate Left Thru Carry	ROL	49	1	3	59	1	3	39	2	5	79	1	5	69	2	6
Rotate Right Thru Carry	ROR	46	1	3	56	1	3	36	2	5	76	1	5	66	2	6
Logical Shift Left	LSL	48	1	3	58	1	3	38	2	5	78	1	5	68	2	6
Logical Shift Right	LSR	44	1	3	54	1	3	34	2	5	74	1	5	64	2	6
Arithmetic Shift Right	ASR	47	1	3	57	1	3	37	2	5	77	1	5	67	2	6
Test for Negative or Zero	TST	4D	1	3	5D	1	3	3D	2	4	7D	1	4	6D	2	5

ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

TABLE 6 — BRANCH INSTRUCTIONS

Function	Mnemonic	Relative Addressing Mode		
		Op Code	# Bytes	# Cycles
Branch Always	BRA	20	2	3
Branch Never	BRN	21	2	3
Branch IFF Higher	BHI	22	2	3
Branch IFF Lower or Same	BLS	23	2	3
Branch IFF Carry Clear	BCC	24	2	3
(Branch IFF Higher or Same)	(BHS)	24	2	3
Branch IFF Carry Set	BCS	25	2	3
(Branch IFF Lower)	(BLO)	25	2	3
Branch IFF Not Equal	BNE	26	2	3
Branch IFF Equal	BEQ	27	2	3
Branch IFF Half Carry Clear	BHCC	28	2	3
Branch IFF Half Carry Set	BHCS	29	2	3
Branch IFF Plus	BPL	2A	2	3
Branch IFF Minus	BMI	2B	2	3
Branch IFF Interrupt Mask Bit is Clear	BMC	2C	2	3
Branch IFF Interrupt Mask Bit is Set	BMS	2D	2	3
Branch IFF Interrupt Line is Low	BIL	2E	2	3
Branch IFF Interrupt Line is High	BIH	2F	2	3
Branch to Subroutine	BSR	AD	2	6

TABLE 7 — BIT MANIPULATION INSTRUCTIONS

Function	Mnemonic	Addressing Modes					
		Bit Set/Clear			Bit Test and Branch		
		Op Code	# Bytes	# Cycles	Op Code	# Bytes	# Cycles
Branch IFF Bit n is Set	BRSET n (n=0...7)	—	—	—	2•n	3	5
Branch IFF Bit n is Clear	BRCLR n (n=0...7)	—	—	—	01+2•n	3	5
Set Bit n	BSET n (n=0...7)	10+2•n	2	5	—	—	—
Clear Bit n	BCLR n (n=0...7)	11+2•n	2	5	—	—	—

TABLE 8 — CONTROL INSTRUCTIONS

Function	Mnemonic	Inherent		
		Op Code	# Bytes	# Cycles
Transfer A to X	TAX	97	1	2
Transfer X to A	TXA	9F	1	2
Set Carry Bit	SEC	99	1	2
Clear Carry Bit	CLC	98	1	2
Set Interrupt Mask Bit	SEI	9B	1	2
Clear Interrupt Mask Bit	CLI	9A	1	2
Software Interrupt	SWI	83	1	10
Return from Subroutine	RTS	81	1	6
Return from Interrupt	RTI	80	1	9
Reset Stack Pointer	RSP	9C	1	2
No-Operation	NOP	9D	1	2
Stop	STOP	8E	1	2
Wait	WAIT	8F	1	2

ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

TABLE 9 — INSTRUCTION SET

Mnemonic	Addressing Modes										Condition Codes				
	Inherent	Immediate	Direct	Extended	Relative	Indexed (No Offset)	Indexed (8 Bits)	Indexed (16 Bits)	Bit Set/ Clear	Bit Test & Branch	H	I	N	Z	C
ADC		X	X	X		X	X	X			↕	•	↕	↕	↕
ADD		X	X	X		X	X	X			↕	•	↕	↕	↕
AND		X	X	X		X	X	X			•	•	↕	↕	•
ASL	X		X			X	X				•	•	↕	↕	↕
ASR	X		X			X	X				•	•	↕	↕	↕
BCC					X						•	•	•	•	•
BCLR									X		•	•	•	•	•
BCS					X						•	•	•	•	•
BEQ					X						•	•	•	•	•
BHCC					X						•	•	•	•	•
BHCS					X						•	•	•	•	•
BHI					X						•	•	•	•	•
BHS					X						•	•	•	•	•
BIH					X						•	•	•	•	•
BIL					X						•	•	•	•	•
BIT		X	X	X		X	X	X			•	•	↕	↕	•
BLO					X						•	•	•	•	•
BLS					X						•	•	•	•	•
BMC					X						•	•	•	•	•
BMI					X						•	•	•	•	•
BMS					X						•	•	•	•	•
BNE					X						•	•	•	•	•
BPL					X						•	•	•	•	•
BRA					X						•	•	•	•	•
BRN					X						•	•	•	•	•
BRCLR										X	•	•	•	↕	↕
BRSET										X	•	•	•	↕	↕
BSET									X		•	•	•	•	•
BSR					X						•	•	•	•	•
CLC	X										•	•	•	•	0
CLI	X										•	0	•	•	•
CLR	X		X			X	X				•	•	0	1	•
CMP		X	X	X		X	X	X			•	•	↕	↕	↕
COM	X		X			X	X				•	•	↕	↕	1
CPX		X	X	X		X	X	X			•	•	↕	↕	↕
DEC	X		X			X	X				•	•	↕	↕	•
EOR		X	X	X		X	X	X			•	•	↕	↕	•
INC	X		X			X	X				•	•	↕	↕	•
JMP			X	X		X	X	X			•	•	•	•	•
JSR			X	X		X	X	X			•	•	•	•	•
LDA		X	X	X		X	X	X			•	•	↕	↕	•
LDX		X	X	X		X	X	X			•	•	↕	↕	•
LSL	X		X			X	X				•	•	↕	↕	↕
LSR	X		X			X	X				•	•	0	↕	↕
NEG	X		X			X	X				•	•	↕	↕	↕
NOP	X										•	•	•	•	•
ORA		X	X	X		X	X	X			•	•	↕	↕	•
ROL	X		X			X	X				•	•	↕	↕	↕
ROR	X		X			X	X				•	•	↕	↕	↕
RSP	X										•	•	•	•	•
RTI	X										?	?	?	?	?
RTS	X										•	•	•	•	•
SBC		X	X	X		X	X	X			•	•	↕	↕	↕
SEC	X										•	•	•	•	1
SEI	X										•	1	•	•	•
STA			X	X		X	X	X			•	•	↕	↕	•
STOP	X										•	0	•	•	•
STX			X	X		X	X	X			•	•	↕	↕	•
SUB		X	X	X		X	X	X			•	•	↕	↕	↕
SWI	X										•	1	•	•	•
TAX	X										•	•	•	•	•
TST	X		X			X	X				•	•	↕	↕	•
TXA	X										•	•	•	•	•
WAIT	X										•	0	•	•	•

Condition Code Symbols

- H Half Carry (From Bit 3)
- I Interrupt Mask
- N Negative (Sign Bit)
- Z Zero
- C Carry/Borrow
- ↕ Test and Set if True, Cleared Otherwise.
- • Not Affected
- ? Load CC Register From Stack
- 0 Cleared
- 1 Set

ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

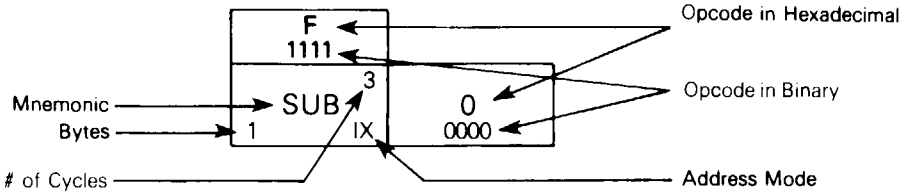
TABLE 10 — MC146805 CMOS INSTRUCTION SET OPCODE MAP

	Bit Manipulation		Branch	Read-Modify-Write					Control		Register/Memory					
	BTB	BSC	REL	DIR	INH	INH	IX1	IX	INH	INH	IMM	DIR	EXT	IX2	IX1	IX
Hi / Low	0 0000	1 0001	2 0010	3 0011	4 0100	5 0101	6 0110	7 0111	8 1000	9 1001	A 1010	B 1011	C 1100	D 1101	E 1110	F 1111
0 0000	BRSET0 5 3 BTB	BSET0 5 2 BSC	BRA 3 2 REL	NEG 5 2 DIR	NEG 3 1 INH	NEG 3 1 INH	NEG 6 2 IX1	NEG 5 1 IX	RTI 9 1 INH		SUB 2 2 IMM	SUB 3 2 DIR	SUB 4 3 EXT	SUB 5 3 IX2	SUB 4 2 IX1	SUB 3 1 IX
1 0001	BRCLR0 5 3 BTB	BCLR0 5 2 BSC	BRN 3 2 REL						RTS 6 1 INH		CMP 2 2 IMM	CMP 3 2 DIR	CMP 4 3 EXT	CMP 5 3 IX2	CMP 4 2 IX1	CMP 3 1 IX
2 0010	BRSET1 5 3 BTB	BSET1 5 2 BSC	BHI 3 2 REL								SBC 2 2 IMM	SBC 3 2 DIR	SBC 4 3 EXT	SBC 5 3 IX2	SBC 4 2 IX1	SBC 3 1 IX
3 0011	BRCLR1 5 3 BTB	BCLR1 5 2 BSC	BLS 3 2 REL	COM 5 2 DIR	COMA 3 1 INH	COMX 3 1 INH	COM 6 2 IX1	COM 5 1 IX	SWI 10 1 INH		CPX 2 2 IMM	CPX 3 2 DIR	CPX 4 3 EXT	CPX 5 3 IX2	CPX 4 2 IX1	CPX 3 1 IX
4 0100	BRSET2 5 3 BTB	BSET2 5 2 BSC	BCC 3 2 REL	LSR 5 2 DIR	LSRA 3 1 INH	LSRX 3 1 INH	LSR 6 2 IX1	LSR 5 1 IX			AND 2 2 IMM	AND 3 2 DIR	AND 4 3 EXT	AND 5 3 IX2	AND 4 2 IX1	AND 3 1 IX
5 0101	BRCLR2 5 3 BTB	BCLR2 5 2 BSC	BCS 3 2 REL								BIT 2 2 IMM	BIT 3 2 DIR	BIT 4 3 EXT	BIT 5 3 IX2	BIT 4 2 IX1	BIT 3 1 IX
6 0110	BRSET3 5 3 BTB	BSET3 5 2 BSC	BNE 3 2 REL	ROR 5 2 DIR	RORA 3 1 INH	RORX 3 1 INH	ROR 6 2 IX1	ROR 5 1 IX			LDA 2 2 IMM	LDA 3 2 DIR	LDA 4 3 EXT	LDA 5 3 IX2	LDA 4 2 IX1	LDA 3 1 IX
7 0111	BRCLR3 5 3 BTB	BCLR3 5 2 BSC	BEQ 3 2 REL	ASR 5 2 DIR	ASRA 3 1 INH	ASRX 3 1 INH	ASR 6 2 IX1	ASR 5 1 IX		TAX 2 1 INH		STA 4 2 DIR	STA 5 3 EXT	STA 6 3 IX2	STA 5 2 IX1	STA 4 1 IX
8 1000	BRSET4 5 3 BTB	BSET4 5 2 BSC	BHCC 3 2 REL	LSL 5 2 DIR	LSLA 3 1 INH	LSLX 3 1 INH	LSL 6 2 IX1	LSL 5 1 IX		CLC 2 1 INH	EOR 2 2 IMM	EOR 3 2 DIR	EOR 4 3 EXT	EOR 5 3 IX2	EOR 4 2 IX1	EOR 3 1 IX
9 1001	BRCLR4 5 3 BTB	BCLR4 5 2 BSC	BHCS 3 2 REL	ROL 5 2 DIR	ROLA 3 1 INH	ROLX 3 1 INH	ROL 6 2 IX1	ROL 5 1 IX		SEC 2 1 INH	ADC 2 2 IMM	ADC 3 2 DIR	ADC 4 3 EXT	ADC 5 3 IX2	ADC 4 2 IX1	ADC 3 1 IX
A 1010	BRSET5 5 3 BTB	BSET5 5 2 BSC	BPL 3 2 REL	DEC 5 2 DIR	DECA 3 1 INH	DECX 3 1 INH	DEC 6 2 IX1	DEC 5 1 IX		CLI 2 1 INH	ORA 2 2 IMM	ORA 3 2 DIR	ORA 4 3 EXT	ORA 5 3 IX2	ORA 4 2 IX1	ORA 3 1 IX
B 1011	BRCLR5 5 3 BTB	BCLR5 5 2 BSC	BMI 3 2 REL							SEI 2 1 INH	ADD 2 2 IMM	ADD 3 2 DIR	ADD 4 3 EXT	ADD 5 3 IX2	ADD 4 2 IX1	ADD 3 1 IX
C 1100	BRSET6 5 3 BTB	BSET6 5 2 BSC	BMC 3 2 REL	INC 5 2 DIR	INCA 3 1 INH	INCX 3 1 INH	INC 6 2 IX1	INC 5 1 IX		RSP 2 1 INH		JMP 2 2 DIR	JMP 3 3 EXT	JMP 4 3 IX2	JMP 3 2 IX1	JMP 2 1 IX
D 1101	BRCLR6 5 3 BTB	BCLR6 5 2 BSC	BMS 3 2 REL	TST 4 2 DIR	TSTA 3 1 INH	TSTX 3 1 INH	TST 5 2 IX1	TST 4 1 IX		NOP 2 1 INH	BSR 6 2 REL	JSR 5 2 DIR	JSR 6 3 EXT	JSR 7 3 IX2	JSR 6 2 IX1	JSR 5 1 IX
E 1110	BRSET7 5 3 BTB	BSET7 5 2 BSC	BIL 3 2 REL						STOP 2 1 INH		LDX 2 2 IMM	LDX 3 2 DIR	LDX 4 3 EXT	LDX 5 3 IX2	LDX 4 2 IX1	LDX 3 1 IX
F 1111	BRCLR7 5 3 BTB	BCLR7 5 2 BSC	BIH 3 2 REL	CLR 5 2 DIR	CLRA 3 1 INH	CLRX 3 1 INH	CLR 6 2 IX1	CLR 5 1 IX	WAIT 2 1 INH	TXA 2 1 INH		STX 4 2 DIR	STX 5 3 EXT	STX 6 3 IX2	STX 5 2 IX1	STX 4 1 IX

Abbreviations for Address Modes

INH	Inherent
A	Accumulator
X	Index Register
IMM	Immediate
DIR	Direct
EXT	Extended
REL	Relative
BSC	Bit Set/Clear
BTB	Bit Test and Branch
IX	Indexed (No Offset)
IX1	Indexed, 1 Byte (8-Bit) Offset
IX2	Indexed, 2 Byte (16-Bit) Offset

LEGEND



ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

FIGURE 20 — CONNECTION TO CMOS PERIPHERALS

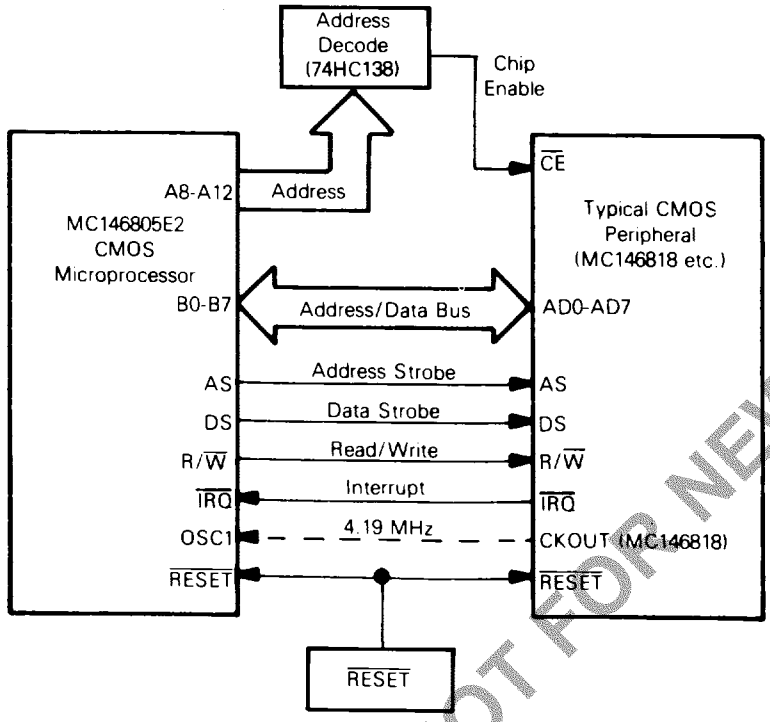
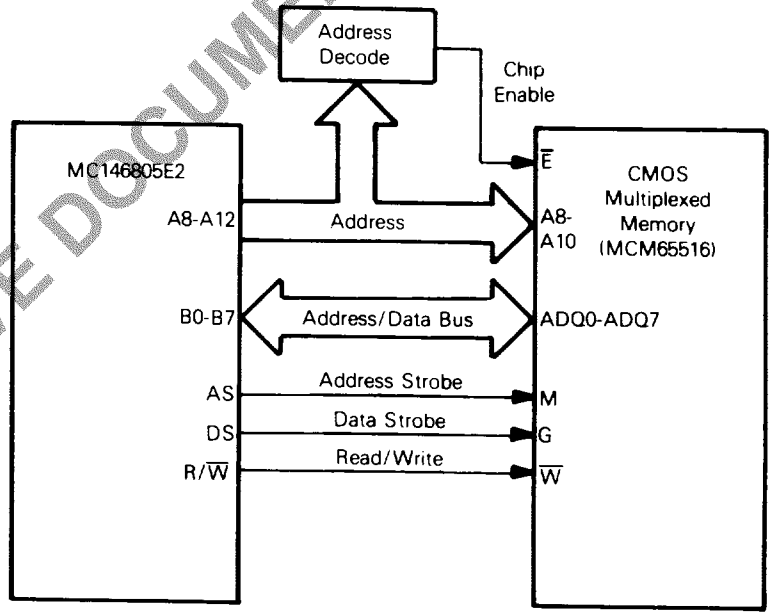
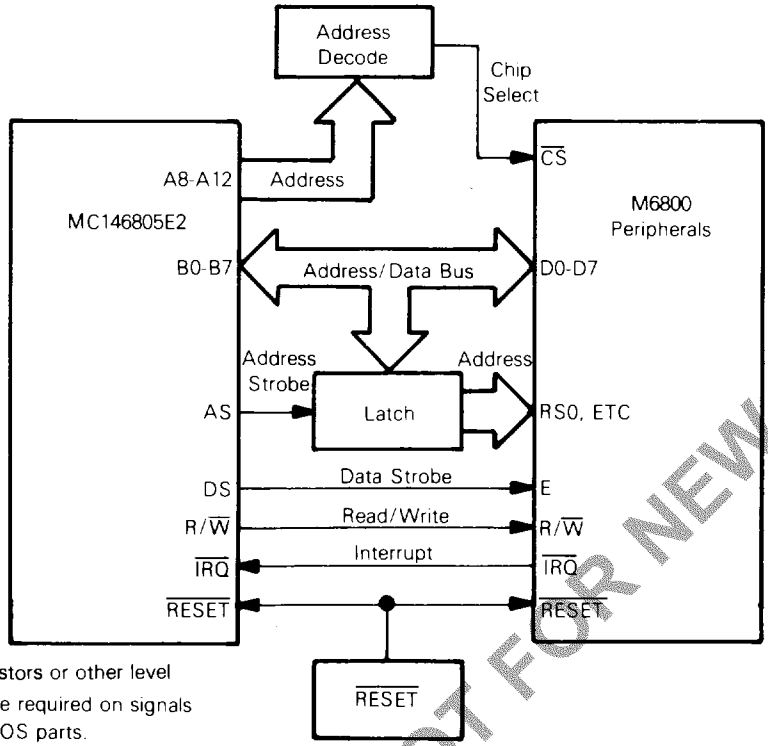


FIGURE 21 — CONNECTION TO CMOS MULTIPLEXED MEMORIES



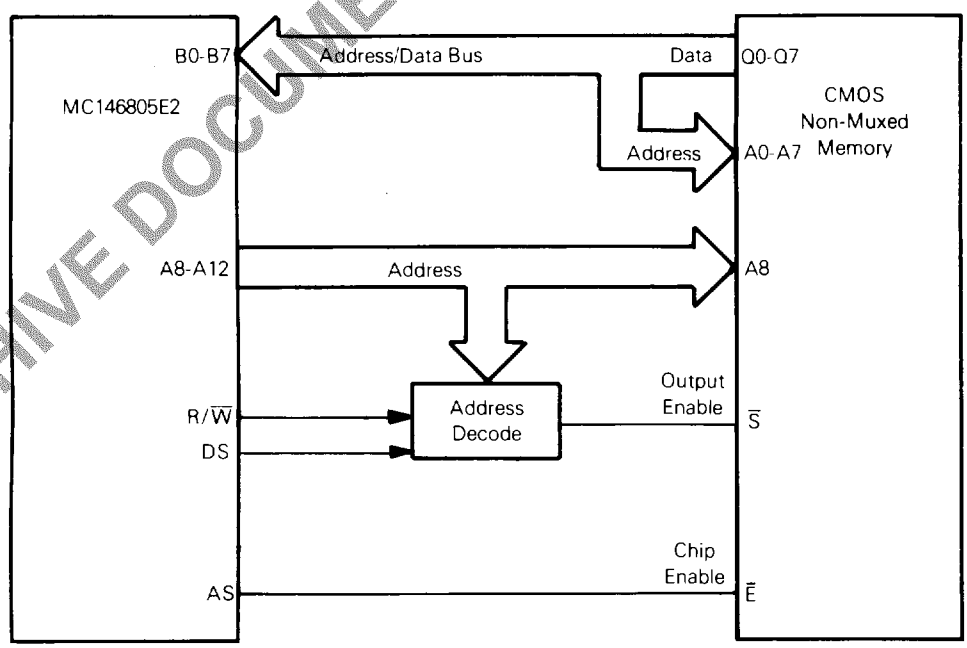
ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

FIGURE 22 — CONNECTION TO M6800 PERIPHERALS



NOTE: In some cases, pullup resistors or other level shifting techniques may be required on signals going from NMOS to CMOS parts.

FIGURE 23 — CONNECTION TO LATCHED NON-MULTIPLEXED CMOS ROM AND EPROM



ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

FIGURE 24 — CONNECTION TO STATIC CMOS RAMS

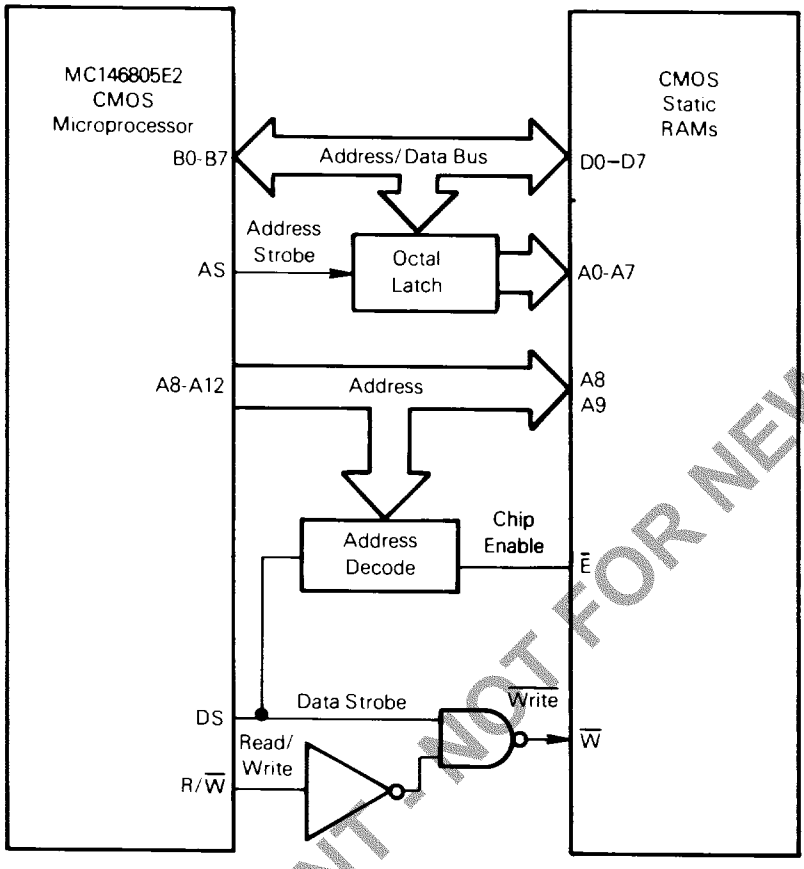
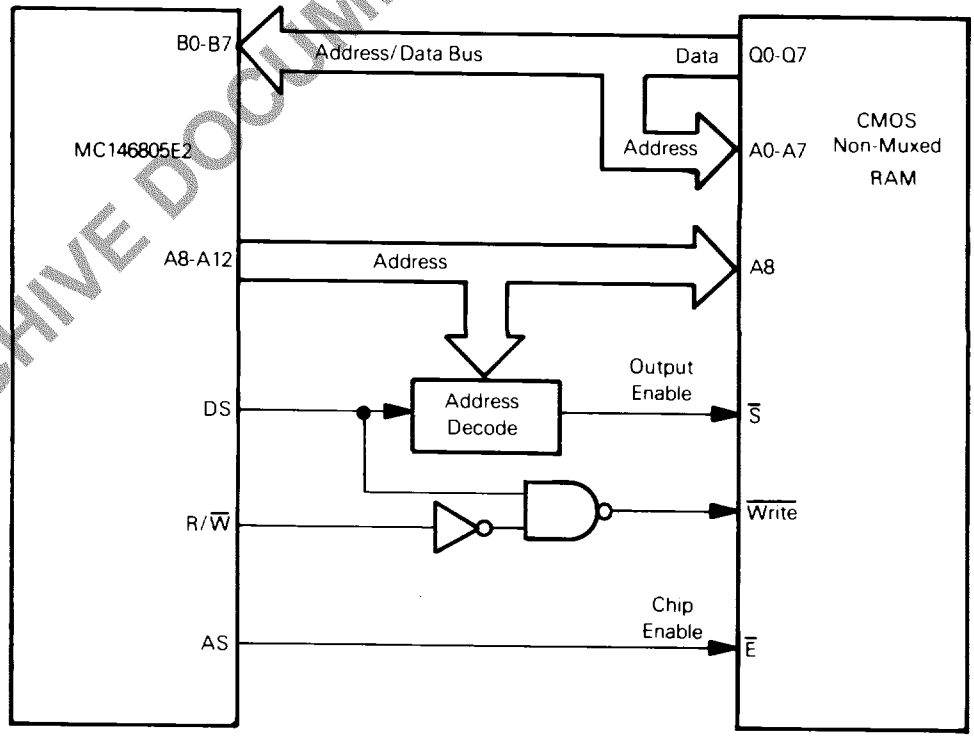


FIGURE 25 — CONNECTION TO LATCHED NON-MULTIPLEXED CMOS RAM



ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

TABLE 11 — SUMMARY OF CYCLE-BY-CYCLE OPERATION

Address Mode / Instructions	Cycles	Cycle #	Address Bus	R/W Pin	LI Pin	Data Bus
Inherent						
LSR LSL ASR NEG CLR ROL COM ROR DEC INC TST	3	1 2 3	Op Code Address Op Code Address +1 Op Code Address +1	1 1 1	1 0 0	Op Code Op Code Next Instruction Op Code Next Instruction
TAX CLC SEC STOP CLI SEI RSP WAIT NOP TXA	2	1 2	Op Code Address Op Code Address +1	1 1	1 0	Op Code Op Code Next Instruction
RTS	6	1 2 3 4 5 6	Op Code Address Op Code Address +1 Stack Pointer Stack Pointer +1 Stack Pointer +2 New Op Code Address	1 1 1 1 1 1	1 0 0 0 0 0	Op Code Op Code Next Instruction Irrelevant Data Irrelevant Data Irrelevant Data New Op Code
SWI	10	1 2 3 4 5 6 7 8 9 10	Op Code Address Op Code Address +1 Stack Pointer Stack Pointer −1 Stack Pointer −2 Stack Pointer −3 Stack Pointer −4 Vector Address 1FFC (Hex) Vector Address 1FFD (Hex) Interrupt Routine Starting Address	1 1 0 0 0 0 0 1 1 1	1 0 0 0 0 0 0 0 0 0	Op Code Op Code Next Instruction Return Address (LO Byte) Return Address (HI Byte) Contents of Index Register Contents of Accumulator Contents of CC Register Address of Int. Routine (HI Byte) Address of Int. Routine (LO Byte) Interrupt Routine First Opcode
RTI	9	1 2 3 4 5 6 7 8 9	Op Code Address Op Code Address +1 Stack Pointer Stack Pointer +1 Stack Pointer +2 Stack Pointer +3 Stack Pointer +4 Stack Pointer +5 New Op Code Address	1 1 1 1 1 1 1 1 1	1 0 0 0 0 0 0 0 0	Op Code Op Code Next Instruction Irrelevant Data Irrelevant Data Irrelevant Data Irrelevant Data Irrelevant Data Irrelevant Data New Op Code
Immediate						
ADC EOR CPX ADD LDA LDX AND ORA BIT SBC CMP SUB	2	1 2	Op Code Address Op Code Address +1	1 1	1 0	Op Code Operand Data
Bit Set/Clear						
BSET n BCLR n	5	1 2 3 4 5	Op Code Address Op Code Address +1 Address of Operand Address of Operand Address of Operand	1 1 1 1 0	1 0 0 0 0	Op Code Address of Operand Operand Data Operand Data Manipulated Data
Bit Test and Branch						
BRSET n BRCLR n	5	1 2 3 4 5	Op Code Address Op Code Address +1 Address of Operand Op Code Address +2 Op Code Address +2	1 1 1 1 1	1 0 0 0 0	Op Code Address of Operand Operand Data Branch Offset Branch Offset
Relative						
BCC BHI BNE BEQ BCS BPL BHCC BLS BIL BMC BRN BHCS BIH BMI BMS BRA	3	1 2 3	Op Code Address Op Code Address +1 Op Code Address +1	1 1 1	1 0 0	Op Code Branch Offset Branch Offset
BSR	6	1 2 3 4 5 6	Op Code Address Op Code Address +1 Op Code Address +1 Subroutine Starting Address Stack Pointer Stack Pointer −1	1 1 1 1 0 0	1 0 0 0 0 0	Op Code Branch Offset Branch Offset First Subroutine Op Code Return Address (LO Byte) Return Address (HI Byte)

ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

TABLE 11 — SUMMARY OF CYCLE-BY-CYCLE OPERATION (CONTINUED)

Address Mode / Instructions	Cycles	Cycle #	Address Bus	R/W Pin	LI Pin	Data Bus
Direct						
JMP	2	1	Op Code Address	1	1	Op Code
		2	Op Code Address + 1	1	0	Jump Address
ADC EOR CPX ADD LDA LDX AND ORA BIT SBC CMP SUB	3	1	Op Code Address	1	1	Op Code
		2	Op Code Address + 1	1	0	Address of Operand
		3	Address of Operand	1	0	Operand Data
TST	4	1	Op Code Address	1	1	Op Code
		2	Op Code Address + 1	1	0	Address of Operand
		3	Address of Operand	1	0	Operand Data
		4	Op Code Address + 2	1	0	Op Code Next Instruction
STA STX	4	1	Op Code Address	1	1	Op Code
		2	Op Code Adrress + 1	1	0	Address of Operand
		3	Op Code Address + 1	1	0	Address of Operand
		4	Address of Operand	0	0	Operand Data
LSL LSR DEC ASR NEG INC CLR ROL COM ROR	5	1	Op Code Address	1	1	Op Code
		2	Op Code Address + 1	1	0	Address of Operand
		3	Operand Address	1	0	Current Operand Data
		4	Operand Address	1	0	Current Operand Data
		5	Operand Address	0	0	New Operand Data
JSR	5	1	Op Code Address	1	1	Op Code
		2	Op Code Address + 1	1	0	Subroutine Address (LO Byte)
		3	Subroutine Starting Address	1	0	1st Subroutine Op Code
		4	Stack Pointer	0	0	Return Address (LO Byte)
		5	Stack Pointer − 1	0	0	Return Address (HI Byte)
Extended						
JMP	3	1	Op Code Address	1	1	Op Code
		2	Op Code Address + 1	1	0	Jump Address (HI Byte)
		3	Op Code Address + 2	1	0	Jump Address (LO Byte)
ADC BIT ORA ADD CMP LDX AND EOR SBC CPX LDA SUB	4	1	Op Code Address	1	1	Op Code
		2	Op Code Address + 1	1	0	Address Operand (HI Byte)
		3	Op Code Address + 2	1	0	Address Operand (LO Byte)
		4	Address of Operand	1	0	Operand Data
STA STX	5	1	Op Code Address	1	1	Op Code
		2	Op Code Address + 1	1	0	Address of Operand (HI Byte)
		3	Op Code Address + 2	1	0	Address of Operand (LO Byte)
		4	Op Code Address + 2	1	0	Address of Operand (LO Byte)
		5	Address of Operand	0	0	Operand Data
JSR	6	1	Op Code Address	1	1	Op Code
		2	Op Code Address + 1	1	0	Address of Subroutine (HI Byte)
		3	Op Code Address + 2	1	0	Address of Subroutine (LO Byte)
		4	Subroutine Starting Address	1	0	1st Subroutine Op Code
		5	Stack Pointer	0	0	Return Address (LO Byte)
		6	Stack Pointer − 1	0	0	Return Address (HI Byte)
Indexed, No-Offset						
JMP	2	1	Op Code Address	1	1	Op Code
		2	Op Code Address + 1	1	0	Op Code Next Instruction
ADC EOR CPX ADD LDA LDX AND ORA BIT SBC CMP SUB	3	1	Op Code Address	1	1	Op Code
		2	Op Code Address + 1	1	0	Op Code Next Instruction
		3	Index Register	1	0	Operand Data
TST	4	1	Op Code Address	1	1	Op Code
		2	Op Code Address + 1	1	0	Op Code Next Instruction
		3	Index Register	1	0	Operand Data
		4	Op Code Address + 1	1	0	Op Code Next Instruction
STA STX	4	1	Op Code Address	1	1	Op Code
		2	Op Code Address + 1	1	0	Op Code Next Instruction
		3	Op Code Address + 1	1	0	Op Code Next Instruction
		4	Index Register	0	0	Operand Data
LSL LSR DEC ASR NEG INC CLR ROL COM ROR	5	1	Op Code Address	1	1	Op Code
		2	Op Code Address + 1	1	0	Op Code Next Instruction
		3	Index Register	1	0	Current Operand Data
		4	Index Register	1	0	Current Operand Data
		5	Index Register	0	0	New Operand Data
JSR	5	1	Op Code Address	1	1	Op Code
		2	Op Code Address + 1	1	0	Op Code Next Instruction
		3	Index Register	1	0	1st Subroutine Op Code
		4	Stack Pointer	0	0	Return Address (LO Byte)
		5	Stack Pointer − 1	0	0	Return Address (HI Byte)

ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

TABLE 11 — SUMMARY OF CYCLE-BY-CYCLE OPERATION (CONTINUED)

Address Mode / Instructions	Cycles	Cycle #	Address Bus	R/$\overline{W}$ Pin	LI Pin	Data Bus
Indexed 8-Bit Offset						
JMP	3	1	Op Code Address	1	1	Op Code
		2	Op Code Address + 1	1	0	Offset
		3	Op Code Address + 1	1	0	Offset
ADC EOR CPX ADD LDA LDX AND ORA CMP SUB BIT SBC	4	1	Op Code Address	1	1	Op Code
		2	Op Code Address + 1	1	0	Offset
		3	Op Code Address + 1	1	0	Offset
		4	Index Register + Offset	1	0	Operand Data
STA STX	5	1	Op Code Address	1	1	Op Code
		2	Op Code Address + 1	1	0	Offset
		3	Op Code Address + 1	1	0	Offset
		4	Op Code Address + 1	1	0	Offset
		5	Index Register + Offset	0	0	Operand Data
TST	5	1	Op Code Address	1	1	Op Code
		2	Op Code Address + 1	1	0	Offset
		3	Op Code Address + 1	1	0	Offset
		4	Index Register + Offset	1	0	Operand Data
		5	Op Code Address + 2	1	0	Op Code Next Instruction
LSL LSR ASR NEG CLR ROL COM ROR DEC INC	6	1	Op Code Address	1	1	Op Code
		2	Op Code Address + 1	1	0	Offset
		3	Op Code Address + 1	1	0	Offset
		4	Index Register + Offset	1	0	Current Operand Data
		5	Index Register + Offset	1	0	Current Operand Data
		6	Index Register + Offset	0	0	New Operand Data
JSR	6	1	Op Code Address	1	1	Op Code
		2	Op Code Address + 1	1	0	Offset
		3	Op Code Address + 1	1	0	Offset
		4	Index Register + Offset	1	0	1st Subroutine Op Code
		5	Stack Pointer	0	0	Return Address LO Byte
		6	Stack Pointer − 1	0	0	Return Address HI Byte
Indexed, 16-Bit Offset						
JMP	4	1	Op Code Address	1	1	Op Code
		2	Op Code Address + 1	1	0	Offset (HI Byte)
		3	Op Code Address + 2	1	0	Offset (LO Byte)
		4	Op Code Address + 2	1	0	Offset (LO Byte)
ADC CMP SUB ADD EOR SBC AND ORA CPX LDA BIT LDX	5	1	Op Code Address	1	1	Op Code
		2	Op Code Address + 1	1	0	Offset (HI Byte)
		3	Op Code Address + 2	1	0	Offset (LO Byte)
		4	Op Code Address + 2	1	0	Offset (LO Byte)
		5	Index Register + Offset	1	0	Operand Data
STA STX	6	1	Op Code Address	1	1	Op Code
		2	Op Code Address + 1	1	0	Offset (HI Byte)
		3	Op Code Address + 2	1	0	Offset (LO Byte)
		4	Op Code Address + 2	1	0	Offset (LO Byte)
		5	Op Code Address + 2	1	0	Offset (LO Byte)
		6	Index Register + Offset	0	0	Operand Data
JSR	7	1	Op Code Address	1	1	Op Code
		2	Op Code Address + 1	1	0	Offset (HI Byte)
		3	Op Code Address + 2	1	0	Offset (LO Byte)
		4	Op Code Address + 2	1	0	Offset (LO Byte)
		5	Index Register + Offset	1	0	1st Subroutine Op Code
		6	Stack Pointer	0	0	Return Address (LO Byte)
		7	Stack Pointer − 1	0	0	Return Address (HO Byte)

ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

Motorola reserves the right to make changes to any products herein to improve reliability, function or design. Motorola does not assume any liability arising out of the application or use of any product or circuit described herein; neither does it convey any license under its patent rights nor the rights of others.

TABLE 11 — SUMMARY OF CYCLE-BY-CYCLE OPERATION (CONTINUED)

Instructions	Cycles	Cycle #	Address Bus	$\overline{RESET}$ Pin	R/$\overline{W}$ Pin	LI Pin	Data Bus
Other Functions							
Hardware $\overline{RESET}$	5		$1FFE	0	1	0	Irrelevant Data
			$1FFE	0	1	0	Irrelevant Data
		1	$1FFE	1	1	0	Irrelevant Data
		2	$1FFE	1	1	0	Irrelevant Data
		3	$1FFE	1	1	0	Vector High
		4	$1FFF	1	1	0	Vector Low
		5	Reset Vector	1	1	0	Op Code
Power on Reset	1922	1	$1FFE	1	1	0	Irrelevant Data
		•	•	•	•	•	•
		•	•	•	•	•	•
		•	•	•	•	•	•
		1919	$1FFE	1	1	0	Irrelevant Data
		1920	$1FFE	1	1	0	Vector High
		1921	$1FFF	1	1	0	Vector Low
		1922	Reset Vector	1	1	0	Op Code
Instruction	**Cycles**	**Cycles #**	**Address Bus**	$\overline{IRQ}$ Pin	R/$\overline{W}$ Pin	LI Pin	**Data Bus**
$\overline{IRQ}$ Interrupt (Timer Vector $1FF8, $1FF9)	10		Last Cycle of Previous Instruction	0	X	0	X
		1	Next Op Code Address	0	1	0	Irrelevant Data
		2	Next Op Code Address	X	1	0	Irrelevant Data
		3	SP	X	0	0	Return Address (LO Byte)
		4	SP – 1	X	0	0	Return Address (HI Byte)
		5	SP – 2	X	0	0	Contents Index Reg
		6	SP – 3	X	0	0	Contents Accumulator
		7	SP – 4	X	0	0	Contents CC Register
		8	$1FFA	X	1	0	Vector High
		9	$1FFB	X	1	0	Vector Low
		10	$\overline{IRQ}$ Vector	X	1	0	Int Routine First

ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

APPENDIX

MC146805E2 INTERRUPT CLARIFICATION

Under certain circumstances, the MC146805E2 (BP4XXXX and AW9XXXX) 8-bit Microprocessor Unit $\overline{IRQ}$ interrupt does not conform to the operation described in this Advanced Information Sheet.

1. The level sensitive $\overline{IRQ}$ mode, which is by far the most frequently used, is **FULLY OPERATIONAL:** thus, most MC146805E2 applications are unaffected. However, the edge-triggered $\overline{IRQ}$ interrupt mode **MIGHT NOT BE SERVICED** under certain programming circumstances; therefore, it is recommended that the edge-triggered mode not be used.

2. An interrupt-vector address **CAN BE** improperly generated in some circumstances. There is a possibility that when an external interrupt ($\overline{IRQ}$) and timer interrupt occur during the WAIT mode (following wait instruction), address locations $1FF2 and $1FF3 are selected instead of vector locations $1FF6 and $1FF7. There are three specific examples listed below; two of these require no action and the third has a recommended solution.

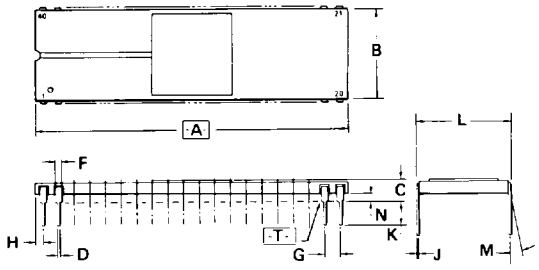
   a. Those not using the WAIT mode need not take any action.

   b. If the WAIT mode is used without external interrupt ($\overline{IRQ}$ pin held high), no precautions are required.

   c. When $\overline{IRQ}$ can be active (low) during the WAIT mode, the vector in locations $1FF6 and $1FF7 (the WAIT mode timer interrupt vector) should be duplicated in $1FF2 and $1FF3. In this way the circumstances that caused selection of the second vector do not disturb normal program execution.

On future MC146805E2 parts, no special actions will be necessary. If you have questions, contact your Motorola distributor or Motorola sales office, or contact Motorola Microprocessor Applications Engineering in Austin, Texas.

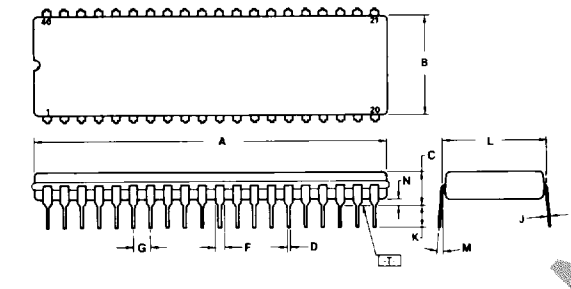
PACKAGE DIMENSIONS



NOTES:
1. DIMENSION A IS DATUM.
2. POSITIONAL TOLERANCE FOR LEADS:
   ⌖ 0.25 (0.010) M T A M
3. T IS SEATING PLANE.
4. DIMENSION "L" TO CENTER OF LEADS WHEN FORMED PARALLEL.
5. DIMENSIONING AND TOLERANCING PER ANSI Y14.5, 1973.

L SUFFIX
CERAMIC PACKAGE
CASE 715-05

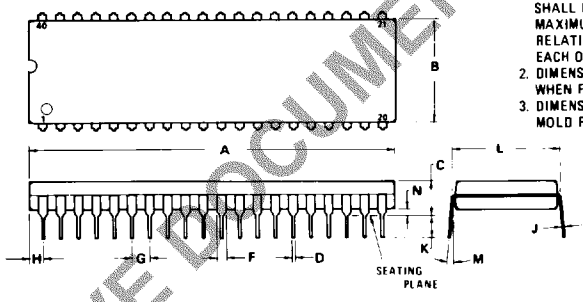
DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	50.29	51.31	1.980	2.020
B	14.63	15.49	0.576	0.610
C	3.05	4.32	0.120	0.170
D	0.38	0.53	0.015	0.021
F	0.76	1.40	0.030	0.055
G	2.54 BSC		0.100 BSC	
J	0.20	0.33	0.008	0.013
K	2.54	4.19	0.100	0.165
L	14.99	15.65	0.590	0.616
M	–	10°	–	10°
N	1.02	1.52	0.040	0.060



NOTES:
1. DIMENSION A IS DATUM.
2. POSITIONAL TOLERANCE FOR LEADS:
   ⌖ 0.25 (0.010) M T A M
3. T IS SEATING PLANE.
4. DIMENSION L TO CENTER OF LEADS WHEN FORMED PARALLEL.
5. DIMENSION A AND B INCLUDES MENISCUS.

S SUFFIX
CERDIP PACKAGE
CASE 734-03

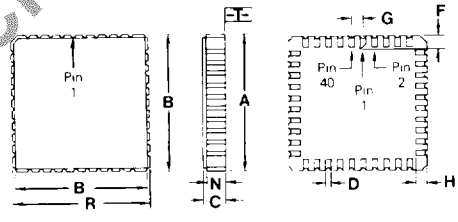
DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	51.31	53.24	2.020	2.096
B	12.70	15.49	0.500	0.610
C	4.06	5.84	0.160	0.230
D	0.38	0.56	0.015	0.022
F	1.27	1.65	0.050	0.065
G	2.54 BSC		0.100 BSC	
J	0.20	0.30	0.008	0.012
K	3.18	4.06	0.125	0.160
L	15.24 BSC		0.600 BSC	
M	5°	15°	5°	15°
N	0.51	1.27	0.020	0.050



NOTES:
1. POSITIONAL TOLERANCE OF LEADS (D), SHALL BE WITHIN 0.25 mm (0.010) AT MAXIMUM MATERIAL CONDITION, IN RELATION TO SEATING PLANE AND EACH OTHER.
2. DIMENSION L TO CENTER OF LEADS WHEN FORMED PARALLEL.
3. DIMENSION B DOES NOT INCLUDE MOLD FLASH.

P SUFFIX
PLASTIC PACKAGE
CASE 711-01

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	51.69	52.45	2.035	2.065
B	13.72	14.22	0.540	0.560
C	3.94	5.08	0.155	0.200
D	0.36	0.56	0.014	0.022
F	1.02	1.52	0.040	0.060
G	2.54 BSC		0.100 BSC	
H	1.65	2.16	0.065	0.085
J	0.20	0.38	0.008	0.015
K	2.92	3.43	0.115	0.135
L	15.24 BSC		0.600 BSC	
M	0°	15°	0°	15°
N	0.51	1.02	0.020	0.040



NOTES:
1. DIMENSIONS A & R ARE DATUMS.
2. T IS GAUGE PLANE.
3. POSITIONAL TOLERANCE FOR TERMINALS (D): 40 PLACES:
   ⌖ 0.25 (0.010) M T A S R S
4. DIMENSIONING AND TOLERANCING PER ANSI Y14.5, 1973.

Z SUFFIX
CHIP CARRIER
CASE 761-01

DIM	MILLIMETERS MIN	MILLIMETERS MAX	INCHES MIN	INCHES MAX
A	11.94	12.57	0.470	0.495
B	11.05	11.30	0.435	0.445
C	1.60	2.08	0.063	0.082
D	0.33	0.69	0.013	0.027
F	1.07	1.47	0.042	0.058
G	1.02 BSC		0.040 BSC	
H	0.84	1.19	0.033	0.047
N	1.27	1.79	0.050	0.070
R	11.94	12.57	0.470	0.495

See front page for chip carrier equivalent pin assignments.

ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

ARCHIVE DOCUMENT - NOT FOR NEW DESIGN

MOTOROLA *Semiconductor Products Inc.*

3501 ED BLUESTEIN BLVD., AUSTIN, TEXAS 78721 • A SUBSIDIARY OF MOTOROLA INC.

A13322-4 PRINTED IN USA 2-84 IMPERIAL LITHO C19855 18,000

AD1850-R3