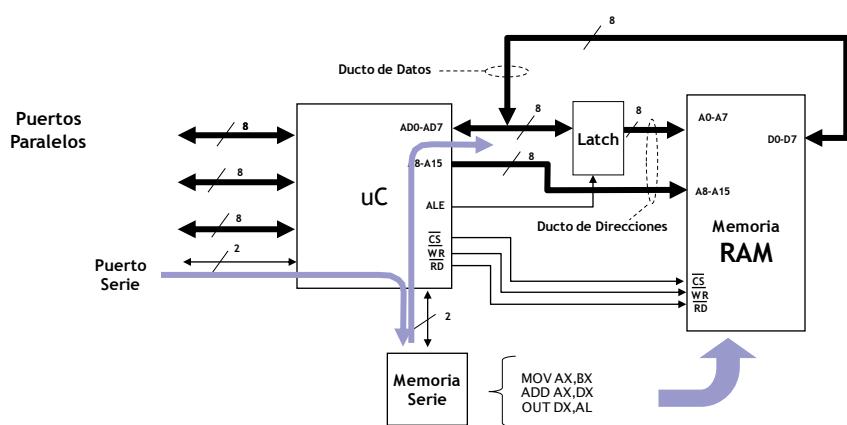


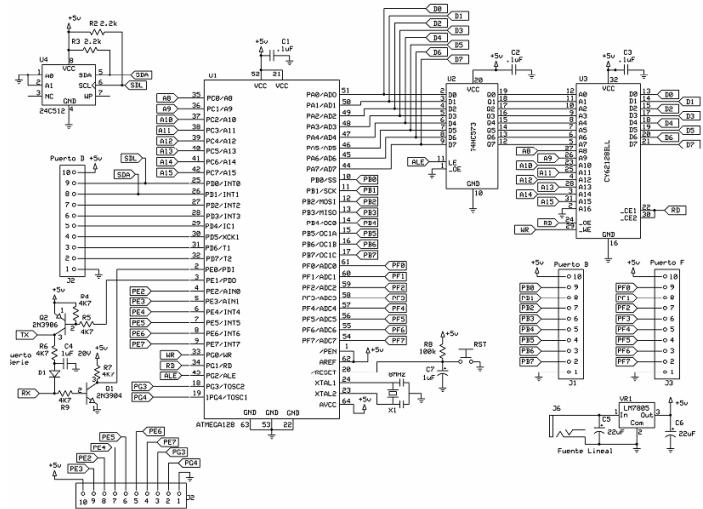
Interprete 80x86

Interprete de código de 16 bits 80x86

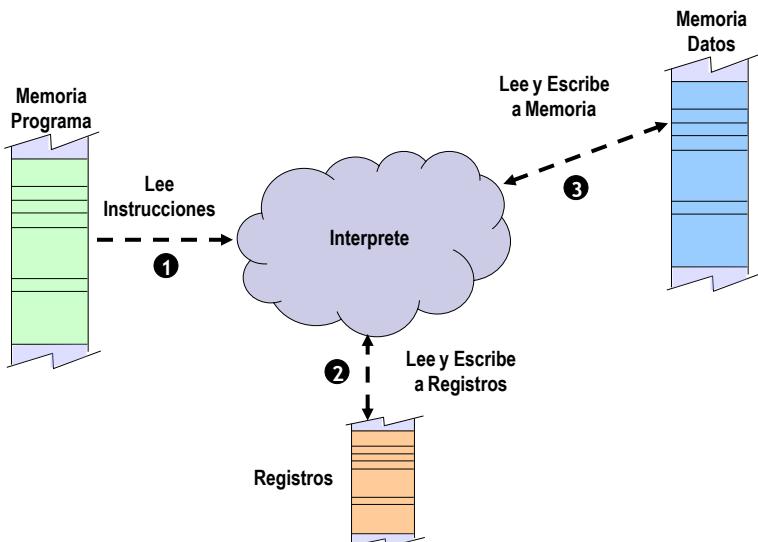
Diagrama Simplificado



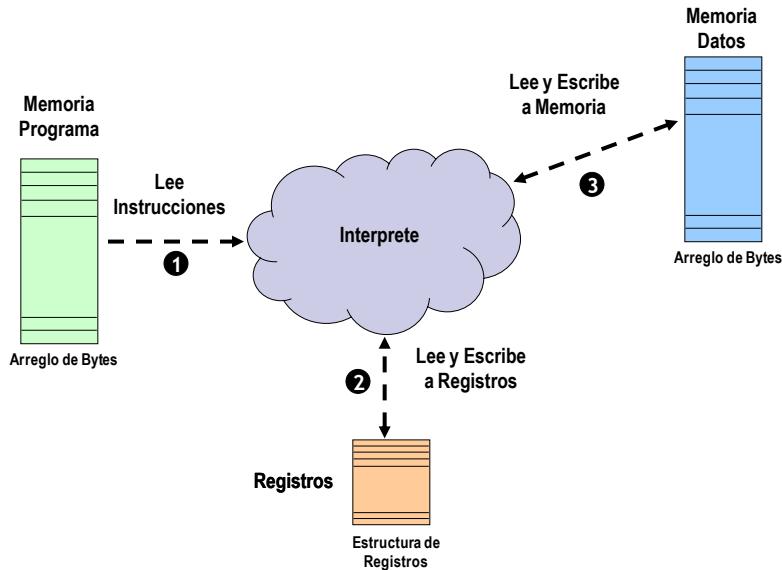
Descripción



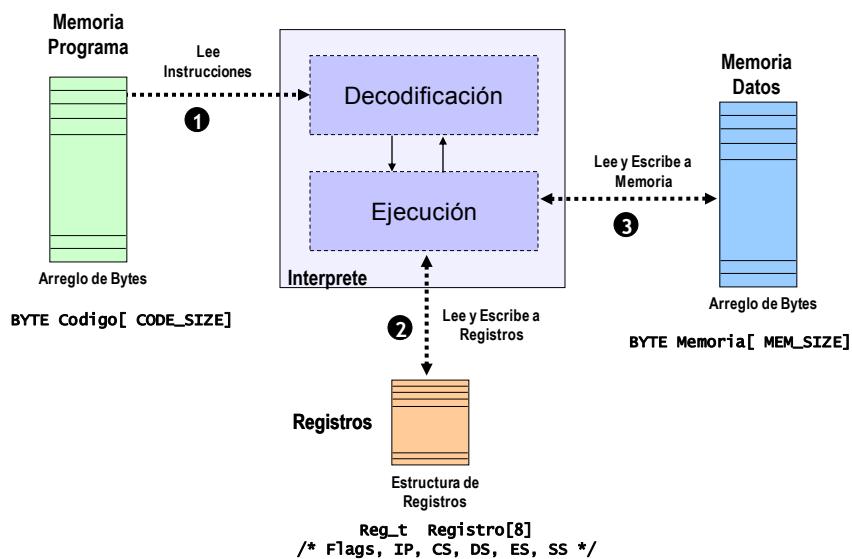
Interprete



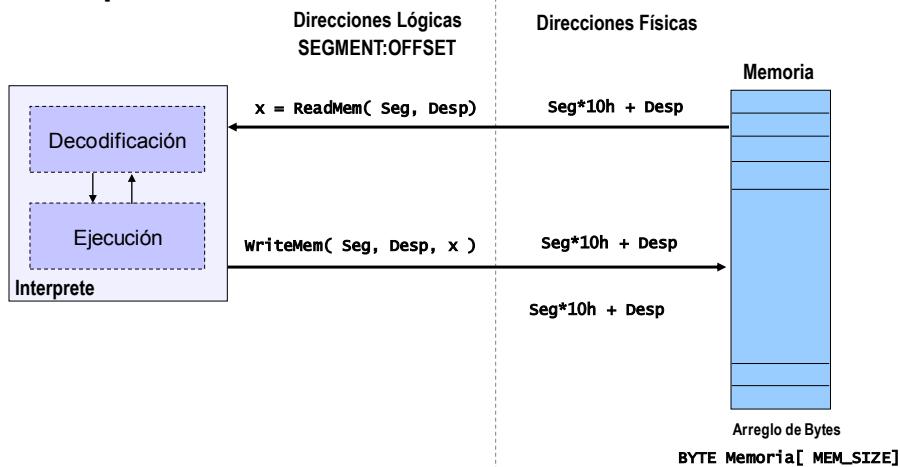
Interprete



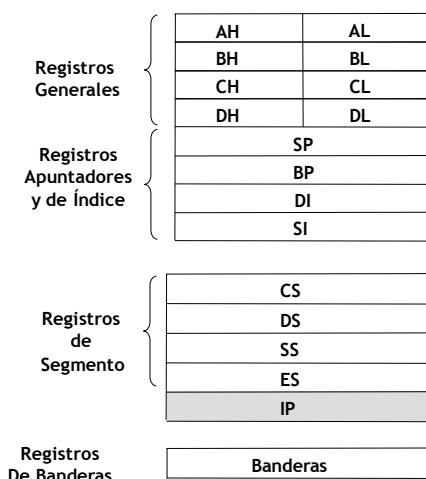
Interprete



Interprete



Registros



Instrucciones (1 o 2 bytes)

INC = Increment:

Register/memory

1 1 1 1 1 1 1 w	mod 0 0 0 r/m
-----------------	---------------

Register

0 1 0 0 0 reg

OUT = Output to:

Fixed port

1 1 1 0 0 1 1 w	port
-----------------	------

Variable port

1 1 1 0 1 1 1 w

PUSH = Push:

Memory

1 1 1 1 1 1 1	mod 1 1 0 r/m
---------------	---------------

Register

0 1 0 1 0 reg

Instrucciones (2, 3 o mas bytes)

Immediate to register

1 0 1 1 w reg	data	data if w = 1
---------------	------	---------------

Memory to accumulator

1 0 1 0 0 0 0 w	addr-low	addr-high
-----------------	----------	-----------

Accumulator to memory

1 0 1 0 0 0 1 w	addr-low	addr-high
-----------------	----------	-----------

Immediate to register/memory

1 1 0 0 0 1 1 w	mod 0 0 0 r/m	data	data if w = 1
-----------------	---------------	------	---------------

Instrucciones

FOOTNOTES

The Effective Address (EA) of the memory operand is computed according to the mod and r/m fields:

if mod = 11 then r/m is treated as a REG field
if mod = 00 then DISP = 0*, disp-low and disp-high are absent
if mod = 01 then DISP = disp-low sign-extended to 16-bits, disp-high is absent
if mod = 10 then DISP = disp-high: disp-low
if r/m = 000 then EA = (BX) + (SI) + DISP
if r/m = 001 then EA = (BX) + (DI) + DISP
if r/m = 010 then EA = (BP) + (SI) + DISP
if r/m = 011 then EA = (BP) + (DI) + DISP
if r/m = 100 then EA = (SI) + DISP
if r/m = 101 then EA = (DI) + DISP
if r/m = 110 then EA = (BP) + DISP*
if r/m = 111 then EA = (BX) + DISP

DISP follows 2nd byte of instruction (before data if required)

*except if mod = 00 and r/m = 110 then EA = disp-high: disp-low.

EA calculation time is 4 clock cycles for all modes, and is included in the execution times given whenever appropriate.

Segment Override Prefix

0	0	1	reg	1	1	0
---	---	---	-----	---	---	---

reg is assigned according to the following:

reg	Segment Register
00	ES
01	CS
10	SS
11	DS

REG is assigned according to the following table:

16-Bit (w = 1)	8-Bit (w = 0)
000 AX	000 AL
001 CX	001 CL
010 DX	010 DL
011 BX	011 BL
100 SP	100 AH
101 BP	101 CH
110 SI	110 DH
111 DI	111 BH

The physical addresses of all operands addressed by the BP register are computed using the SS segment register. The physical addresses of the destination operands of the string primitive operations (those addressed by the DI register) are computed using the ES segment, which may not be overridden.

Instrucciones

REG is assigned according to the following table:

16-Bit (w = 1) 8-Bit (w = 0)

000 AX	000 AL
001 CX	001 CL
010 DX	010 DL
011 BX	011 BL
100 SP	100 AH
101 BP	101 CH
110 SI	110 DH
111 DI	111 BH

Instrucciones

INC = Increment:

Register/memory

1111111w mod 000 r/m

Register

01000 reg

Ejemplo:

Código	Campos	Mnemónico
40h	01000 000 oper reg	INC AX
41h	01000 001	INC CX
	:	
47h	01000 111	INC DI

Decodificación y Ejecución

Ejemplo:

```
switch (codigo){  
    case 0x40:    AX++; /* incrementa AX */  
    break;  
  
    case 0x41:    CX++; /* incrementa cx */  
    break;  
  
    :  
  
    case 0x47:    DI++; /* incrementa cx */  
    break;  
}
```



Decodificación y Ejecución

Ejemplo:

```
/* a) Tener un arreglo de registros que conforman a AX, CX etc. */  
T_Registro Registro[8];  
  
/* b) Tener un registro de instrucción formado por campos de bits */  
/* teniendo así: Código.oper y Código.reg */  
  
/* ahora si */      #define INC_R16 0x08  
  
if ( Código.oper == INC_R16 )  
    Registro[ Código.reg ]++;
```



Esquema de Registros

INC = Increment:

Register/memory	1 1 1 1 1 1 1 w	mod 0 0 0 r/m
Register	0 1 0 0 0	reg

Registros Generales	AH	AL
	BH	BL
	CH	CL
	DH	DL
SP		
BP		
DI		
SI		
CS		
DS		
SS		
ES		
IP		

REG is assigned according to the following table:

16-Bit (w = 1)	8-Bit (w = 0)
000 AX	000 AL
001 CX	001 CL
010 DX	010 DL
011 BX	011 BL
100 SP	100 AH
101 BP	101 CH
110 SI	110 DH
111 DI	111 BH

Esquema de Registros

INC = Increment:

Register/memory

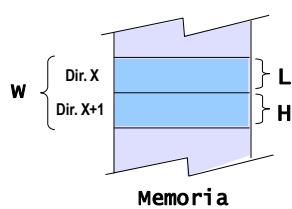
1111111w | mod 0 0 0 r/m

DIR. LÓGICA	CÓDIGO	MNEMÓNICO
136E:0100	FEC0	INC AL
136E:0102	FEC4	INC AH
136E:0108	FE060001	INC BYTE PTR [0100]
136E:0104	FF060001	INC WORD PTR [0100]

Declaración de Registros

```
#define WORD unsigned int  
#define BYTE unsigned char
```

```
union u_reg {  
    WORD w;          /* Registro de 16 bits */  
    BYTE L;          /* Registros de 8 bits */  
    BYTE H;          /* Registros de 8 bits */  
};
```



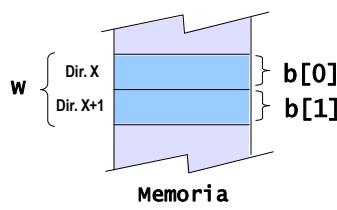
Ejemplo

```
/* declarando a AX como: */  
union u_reg AX;  
  
/* se puede accesar como */  
AX.w = 0x1234; /* 16 bits */  
  
/* 8 bits */  
AX.L = 0x23; /* AL */  
AX.H = 0xef; /* AH */
```

Declaración de Registros

```
#define WORD unsigned int
#define BYTE unsigned char

union u_reg {
    WORD w;           /* Registro de 16 bits */
    BYTE b[2];        /* Registros de 8 bits */
    -BYTE L;
    -BYTE H;
};
```



Ejemplo:

```
/* declarando a AX como: */
union u_reg AX;

/* se puede accesar como */
AX.w = 0x1234; /* 16 bits */

/* 8 bits */
AX.b[0] = 0x23; /* AL */
AX.b[1] = 0xef; /* AH */
```

Declaración de Registros

```
#define WORD unsigned int
#define BYTE unsigned char

union u_reg {
    WORD w;           /* Registro de 16 bits */
    BYTE b[2];        /* Registros de 8 bits */
};

/* nuevo tipo de dato */
typedef union u_reg Registro;

/* Ejemplo de Acceso */

Registro AX;

AX.w = 0x1234;      /* Como registro de 16 bits */
AX.b[0]= 0xff;       /* Como registro de 8 bits (baja) */
AX.b[1]= 0x3f;       /* Como registro de 8 bits (alta) */
```

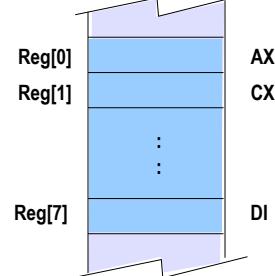
Declaración de Registros

```
#define WORD unsigned int
#define BYTE unsigned char

union u_reg {
    WORD w;           /* Registro de 16 bits */      */
    BYTE b[2];        /* Registros de 8 bits */     */
};

typedef union u_reg Registro; /* nuevo tipo de dato */

Registro Reg[8];
/* Acceso a nuevo tipo de dato */
```



Acceso a Registros

- Caso de Registros de 16 bits

```
/* si REG=7 */
dato=Reg[REG].w;

/* si REG=0 */
dato=Reg[REG].w;

/* si REG=3 */
dato=Reg[REG].w; /
```

Reg

16-Bit (w = 1)	8-Bit (w = 0)
000 AX	000 AL
001 CX	001 CL
010 DX	010 DL
011 BX	011 BL
100 SP	100 AH
101 BP	101 CH
110 SI	110 DH
111 DI	111 BH

Acceso a Registros

- Caso de Registros de 8 bits

```
/* si REG=0 → AL */
dato=Reg[REG].b[REG],  
  
/* si REG=1 → CL */
dato=Reg[REG].b[REG],
```

16-Bit (w = 1)	8-Bit (w = 0)
000 AX	000 AL
001 CX	001 CL
010 DX	010 DL
011 BX	011 BL
	100 AH
	101 CH
	110 DH
	111 BH

Acceso a Registros

- Caso de Registros de 8 bits

```
/* si REG=0 → AL */
dato=Reg[REG].b[REG],  
  
/* si REG=1 → CL */
dato=Reg[REG].b[REG],  
  
/* si REG=1 → CL */
dato=Reg[REG % 4].b[REG / 4];  
  
#define REG_PG16( reg ) reg & 0x03
#define REG_HL( reg ) reg>>2  
  
dato=Reg[ REG_PG( REG ) ].b[ REG_HL( REG ) ];
```

16-Bit (w = 1)	8-Bit (w = 0)
000 AX	000 AL
001 CX	001 CL
010 DX	010 DL
011 BX	011 BL
	100 AH
	101 CH
	110 DH
	111 BH

Acceso a Registros

- Caso de General

```
#define REG_PG16( reg ) reg & 0x03
#define REG_HL( reg ) reg>>2

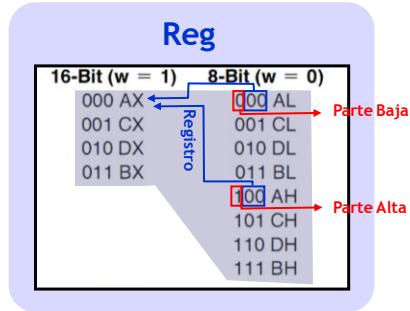
registro=codigo.reg;

If( codigo.w == 1)

    Reg[ registro ].w++;

else

    Reg[ REG_PG( registro ) ].b[ REG_HL( registro ) ]++;
```



Decodificación de Instrucciones (2 bytes)

FOOTNOTES

The Effective Address (EA) of the memory operand is computed according to the mod and r/m fields:

if mod = 11 then r/m is treated as a REG field
if mod = 00 then DISP = 0*, disp-low and disp-high are absent
if mod = 01 then DISP = disp-low sign-extended to 16-bits, disp-high is absent
if mod = 10 then DISP = disp-high; disp-low
if r/m = 000 then EA = (BX) + (SI) + DISP
if r/m = 001 then EA = (BX) + (DI) + DISP
if r/m = 010 then EA = (BP) + (SI) + DISP
if r/m = 011 then EA = (BP) + (DI) + DISP
if r/m = 100 then EA = (SI) + DISP
if r/m = 101 then EA = (DI) + DISP
if r/m = 110 then EA = (BP) + DISP*
if r/m = 111 then EA = (BX) + DISP

DISP follows 2nd byte of instruction (before data if required)

*except if mod = 00 and r/m = 110 then EA = disp-high: disp-low.

EA calculation time is 4 clock cycles for all modes, and is included in the execution times given whenever appropriate.

Segment Override Prefix

0	0	1	reg	1	1	0
---	---	---	-----	---	---	---

reg is assigned according to the following:

reg	Segment Register
00	ES
01	CS
10	SS
11	DS

REG is assigned according to the following table:

16-Bit (w = 1)	8-Bit (w = 0)
000 AX	000 AL
001 CX	001 CL
010 DX	010 DL
011 BX	011 BL
100 SP	100 AH
101 BP	101 CH
110 SI	110 DH
111 DI	111 BH

The physical addresses of all operands addressed by the BP register are computed using the SS segment register. The physical addresses of the destination operands of the string primitive operations (those addressed by the DI register) are computed using the ES segment, which may not be overridden.

Decodificación de Instrucciones (2 bytes)

DATA TRANSFER

MOV = Move:

Register to Register/Memory

1000100 w	mod reg r/m
-----------	-------------

Register/memory to register

1000101 w	mod reg r/m
-----------	-------------

Immediate to register/memory

1100011 w	mod 000 r/m	data	data if w = 1
-----------	-------------	------	---------------

Immediate to register

1011 w reg	data	data if w = 1
------------	------	---------------

Memory to accumulator

1010000 w	addr-low	addr-high
-----------	----------	-----------

Accumulator to memory

1010001 w	addr-low	addr-high
-----------	----------	-----------

Register/memory to segment register

1000111 0	mod 0 reg r/m
-----------	---------------

Segment register to register/memory

1000110 0	mod 0 reg r/m
-----------	---------------

Decodificación de Instrucciones

Immediate to register

1011 w reg	data	data if w = 1
------------	------	---------------

W=0

1011 0	reg	data
--------	-----	------

MOV Reg₈, Valor

Ejemplo:

1011 0	000	00010010
--------	-----	----------

B 0 1 2

MOV AL, 12h

REG is assigned according to the following table:

16-Bit (w = 1) 8-Bit (w = 0)

000 AX	000 AL
001 CX	001 CL
010 DX	010 DL
011 BX	011 BL
100 SP	100 AH
101 BP	101 CH
110 SI	110 DH
111 DI	111 BH

Decodificación de Instrucciones

Immediate to register

1 0 1 1 w	reg	data	data if w = 1
-----------	-----	------	---------------

W=1 ↓

1 0 1 1 1	reg	data (L)	data (H)
-----------	-----	----------	----------

MOV Reg₁₆, Valor

Ejemplo:

1 0 1 1 1	1 1 1	0 0 0 0 0 1 1 1	0 0 0 0 0 0 0 0
-----------	-------	-----------------	-----------------

B F 0 7 0 0

MOV DI, 0007h

16-Bit (w = 1)	8-Bit (w = 0)
000 AX	000 AL
001 CX	001 CL
010 DX	010 DL
011 BX	011 BL
100 SP	100 AH
101 BP	101 CH
110 SI	110 DH
111 DI	111 BH

Decodificación de Instrucciones (2 bytes)

DATA TRANSFER

MOV = Move:

Register to Register/Memory

1 0 0 0 1 0 0 w	mod reg r/m
-----------------	-------------

Register/memory to register

1 0 0 0 1 0 1 w	mod reg r/m
-----------------	-------------

Immediate to register/memory

1 1 0 0 0 1 1 w	mod 000 r/m	data	data if w = 1
-----------------	-------------	------	---------------

Immediate to register

1 0 1 1 w	reg	data	data if w = 1
-----------	-----	------	---------------

Memory to accumulator

1 0 1 0 0 0 0 w	addr-low	addr-high
-----------------	----------	-----------

Accumulator to memory

1 0 1 0 0 0 1 w	addr-low	addr-high
-----------------	----------	-----------

Register/memory to segment register

1 0 0 0 1 1 1 0	mod 0 reg r/m
-----------------	---------------

Segment register to register/memory

1 0 0 0 1 1 0 0	mod 0 reg r/m
-----------------	---------------

Decodificación de Instrucciones (2 bytes)

Memory to accumulator

1010000 w	addr-low	addr-high
-----------	----------	-----------

Accumulator to memory

1010001 w	addr-low	addr-high
-----------	----------	-----------

Memoria al Acumulador:

MOV AX, [offset]

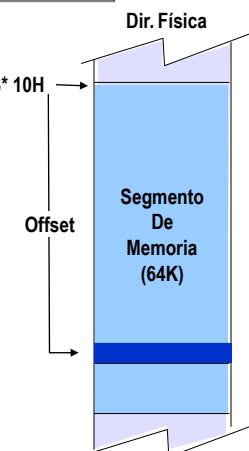
MOV AL, [offset]

Acumulador a Memoria:

MOV [offset], AX

MOV [offset], AL

Offset: Desplazamiento (valor de 16 bits)



Decodificación de Instrucciones

Memory to accumulator

1010000 w	addr-low	addr-high
-----------	----------	-----------

W=1

MOV AX, [offset]

101000 1	addr-low	addr-high
----------	----------	-----------

Ejemplo:

1010000 1	00110100	00010010
-----------	----------	----------

A 1 3 4 1 2

MOV AX, [1234h]

Decodificación de Instrucciones

Memory to accumulator

1010000w	addr-low	addr-high
----------	----------	-----------

w=0

MOV AL, [offset]

1010000	addr-low	addr-high
---------	----------	-----------

Ejemplo:

10100000	00110100	00010010
----------	----------	----------

A 0 3 4 1 2

MOV AL, [1234h]

Decodificación de Instrucciones (2 bytes)

DATA TRANSFER

MOV = Move:

Register to Register/Memory

1000100w	mod reg r/m
----------	-------------

Register/memory to register

1000101w	mod reg r/m
----------	-------------

Immediate to register/memory

1100011w	mod 000 r/m	data	data if w = 1
----------	-------------	------	---------------

Immediate to register

1011w reg	data	data if w = 1
-----------	------	---------------

Memory to accumulator

1010000w	addr-low	addr-high
----------	----------	-----------

Accumulator to memory

1010001w	addr-low	addr-high
----------	----------	-----------

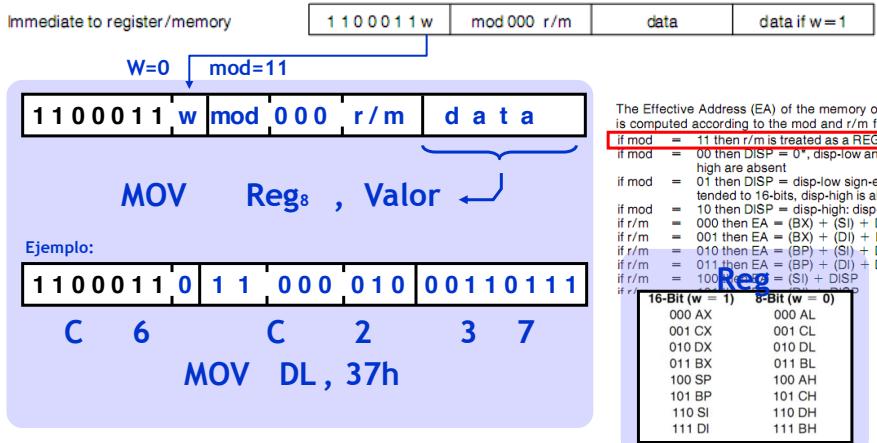
Register/memory to segment register

10001110	mod 0 reg r/m
----------	---------------

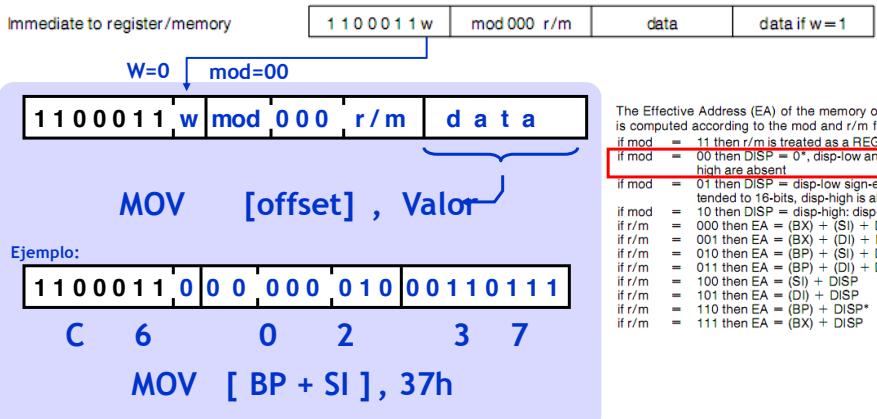
Segment register to register/memory

10001100	mod 0 reg r/m
----------	---------------

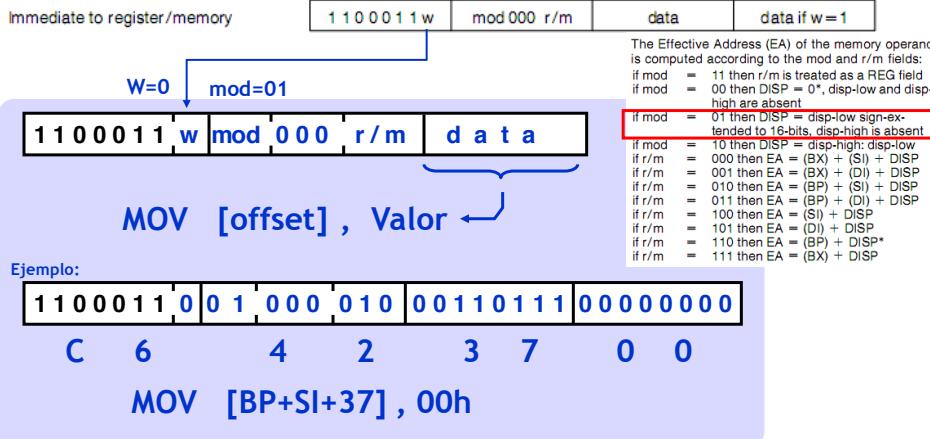
Continuación...



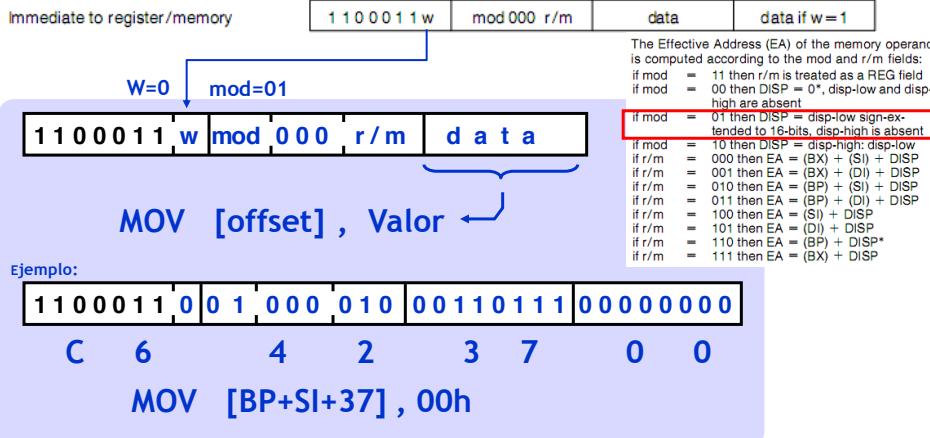
Continuación...



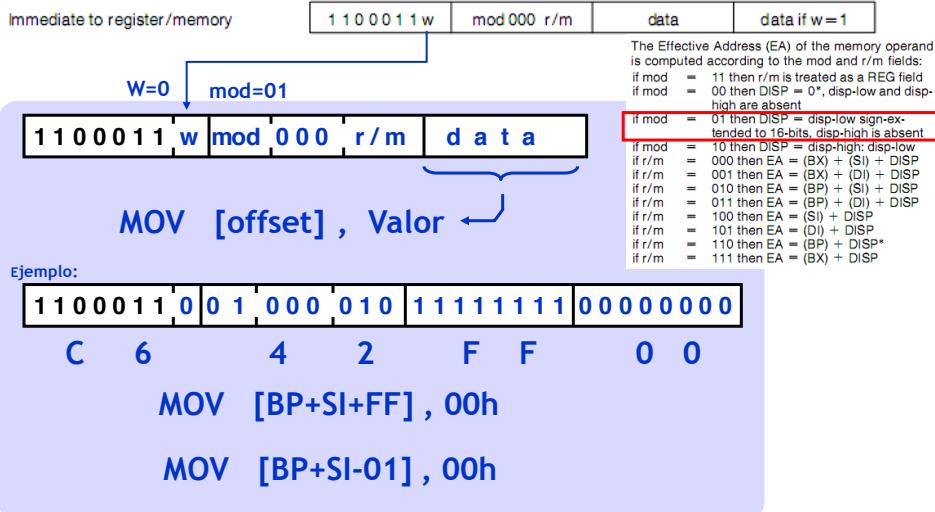
Continuación...



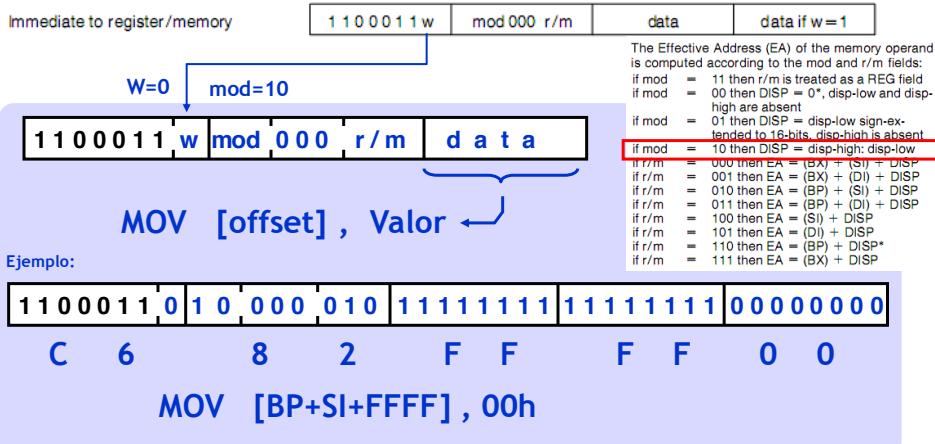
Continuación...



Continuación... (una más del mismo tipo)



Continuación...



Decodificación de Instrucciones (2 bytes)

DATA TRANSFER

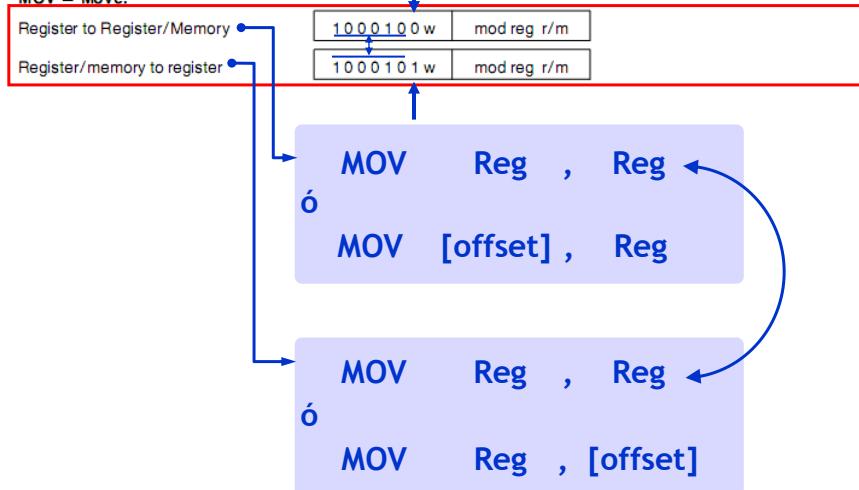
MOV = Move:

Register to Register/Memory	1000100 w	mod reg r/m		
Register/memory to register	1000101 w	mod reg r/m		
Immediate to register/memory	1100011 w	mod 000 r/m	data	data if w = 1
Immediate to register	1011 w	reg	data	data if w = 1
Memory to accumulator	1010000 w		addr-low	addr-high
Accumulator to memory	1010001 w		addr-low	addr-high
Register/memory to segment register	1000111 0	mod 0 reg r/m		
Segment register to register/memory	1000110 0	mod 0 reg r/m		

Decodificación de Instrucciones (2 bytes)

DATA TRANSFER

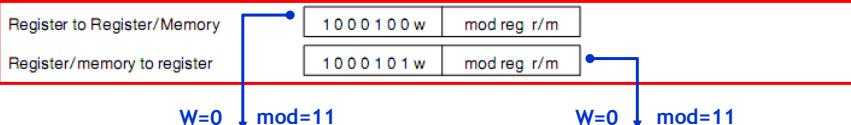
MOV = Move:



Decodificación de Instrucciones (2 bytes)

DATA TRANSFER

MOV = Move:



1000100 w mod reg r/m

1000101 w mod reg r/m

MOV Reg₈, Reg₈

MOV Reg₈, Reg₈

1000100 0 | 11 001 010

1000101 0 | 11 010 001

8 8 C A

8 A D 1

MOV DL , CL

MOV DL , CL

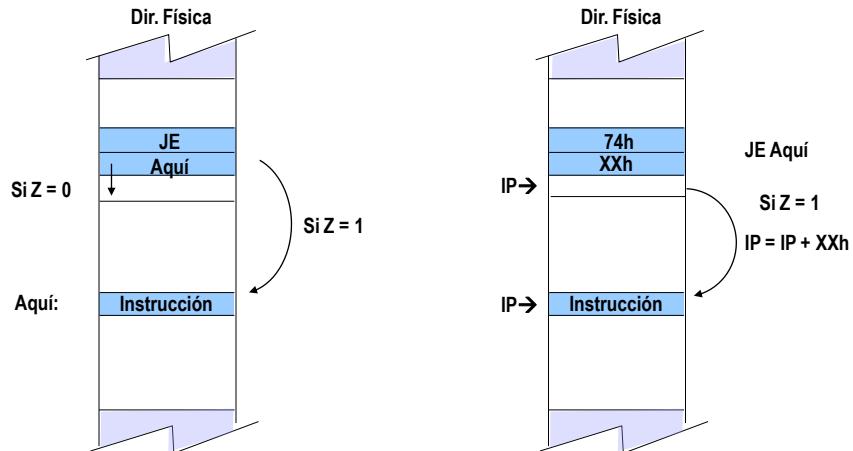
Decodificación de Instrucciones JMP condicional

JE/JZ = Jump on equal/zero	01110100	disp
JL/JNGE = Jump on less/not greater or equal	01111100	disp
JLE/JNG = Jump on less or equal/not greater	01111110	disp
JB/JNAE = Jump on below/not above or equal	01111010	disp
JBE/JNA = Jump on below or equal/not above	011110110	disp
JP/JPE = Jump on parity/parity even	011111010	disp
JO = Jump on overflow	01110000	disp
JS = Jump on sign	01111000	disp
JNE/JNZ = Jump on not equal/not zero	01110101	disp
JNL/JGE = Jump on not less/greater or equal	01111101	disp
JNLE/JG = Jump on not less or equal/greater	01111111	disp
JNB/JAE = Jump on not below/above or equal	01110011	disp
JNBE/JA = Jump on not below or equal/above	01110111	disp
JNP/JPO = Jump on not par/par odd	01111011	disp
JNO = Jump on not overflow	01110001	disp
JNS = Jump on not sign	01111001	disp
JCXZ = Jump on CX zero	11100011	disp
LOOP = Loop CX times	11100010	disp
LOOPZ/LOOPE = Loop while zero/equal	11100001	disp
LOOPNZ/LOOPNE = Loop while not zero/equal	11100000	disp

Decodificación de Instrucciones JMP y CALL

JE/JZ = Jump on equal/zero

0 1 1 1 0 1 0 0	disp
-----------------	------



Decodificación de Instrucciones JMP y CALL

JMP = Unconditional jump:

Short/long

1 1 1 0 1 0 1 1	disp-low
-----------------	----------

Direct within segment

1 1 1 0 1 0 0 1	disp-low	disp-high
-----------------	----------	-----------

Register/memory
indirect within segment

1 1 1 1 1 1 1 1	mod 1 0 0 r/m
-----------------	---------------

Direct intersegment

1 1 1 0 1 0 1 0	segment offset
	segment selector

Indirect intersegment

1 1 1 1 1 1 1 1	mod 1 0 1 r/m	(mod ≠ 11)
-----------------	---------------	------------

Terminales y sus Funciones

Descripción del circuito integrado

		8088	CPU	MODO MINIMO	MODO MAXIMO
GND	1			40	Vec
A14	2			39	A15
A13	3			38	A16/S3
A12	4			37	A17/S4
A11	5			36	A18/S5
A10	6			35	A19/S6
A9	7			34	SS0 (HIGH)
A8	8			33	MN/ MX
AD7	9			32	RD
AD6	10			31	HOLD (R0/ GT0)
AD5	11			30	HLDA (R0/ GT1)
AD4	12			29	WR (LOCK)
AD3	13			28	IO/M (S2)
AD2	14			27	DT/R (S1)
AD1	15			26	DEN (S0)
AD0	16			25	ALE (QS0)
NMI	17			24	INTA (QS1)
INTR	18			23	TEST
CLK	19			22	READY
GND	20			21	RESET

Figura 1. Esquema de terminales del microprocesador 8088

La figura 1 muestra el esquema de terminales del microprocesador 8088 con un encapsulado tipo DIP (*Dual In-line Packages*) de 40 terminales. El microprocesador 8088 requiere +5V de voltaje de alimentación con una tolerancia de ± 10 por ciento. El 8088 consume un máximo de 340 mA. Tanto el 8086 como el 8088 operan a temperatura ambiente entre 32° F y 180° F.

Características de Entrada: Las características de entrada de este microprocesador son compatibles con todas las familias lógicas estandares disponibles hoy en día. Los niveles de corriente de entrada son muy pequeños (representan solamente una corriente de fuga) debido a que las compuertas de entrada del dispositivo son del tipo MOSFET.
(Nivel lógico 0 = 0.8V max. ; Nivel lógico 1= 2.0 V min.)

Características de Salida: El nivel lógico 1 del 8088 es compatible con todas las familias lógicas estándares, pero el nivel lógico 0 no lo es. En los circuitos lógicos estándares se tiene un voltaje máximo de 0.4V para un nivel 0, y el 8088 tiene una máxima de 0.45V, existiendo una diferencia de 0.05V. Esta diferencia reduce la inmunidad al ruido de un nivel estándar de 400 mV a 350 mV. (La inmunidad al ruido es la diferencia entre el voltaje de salida de lógico 0 y el voltaje de entrada lógico 0). Debido a esto es recomendable conectar a las salidas del 8088 reforzadores (buffers).
 (Nivel lógico 1 = 0.45V max. @ 2.0 mA max. ; Nivel lógico 1 = 2.4 V min. @ -400 μA max.)

Señales y Terminales del microprocesador 8088

1. **AD₇-AD₀**-Bus de Direcciones/Datos: Estas líneas componen el bus de direcciones y datos multiplexado del 8088 y contienen los 8 bit menos significativos de la dirección de memoria o puerto de E/S cuando ALE está activa (1) o dato cuando ALE está inactivo (0). Estas terminales pasan a estado de alta impedancia durante un reconocimiento de retención (hold).
2. **A₁₅-A₈** -Bus de Direcciones: Estas terminales forman parte del bus de direcciones. Estas terminales pasan a estado de alta impedancia durante un reconocimiento de retención (hold).
3. **A₁₉/S₆, A₁₈/S₅, A₁₇/S₄ y A₁₆/S₃**-Dirección/Estado: Terminales multiplexadas que contienen los bit A₁₉-A₁₆ durante ALE activo, para el resto del ciclo maquina contienen los bits de estado S₆-S₃. Estas terminales pasan a estado de alta impedancia durante un reconocimiento de retención (hold).
4. **RD-Leer**: Se convierte en 0 lógico cuando el bus de datos recibe un dato de la memoria o de un puerto de E/S. Estas terminales pasan a estado de alta impedancia durante un reconocimiento de retención (hold).
5. **READY-Listo**: Se utiliza para interfazar memorias lentas y periféricos al 8088, si esta terminal está en alto se ejecutan instrucciones sin ciclos de espera y si está en bajo ciclos de espera serán insertados hasta que READY sea alto.
6. **INTR-Requerimiento de Interrupción**: Se utiliza para requerir una interrupción física (por circuitería).
7. **TEST-Prueba**: Esta terminal es probada por una instrucción WAIT. Si TEST es un 0 lógico, entonces la instrucción WAIT continua con la próxima instrucción en el programa, si TEST es un 1 lógico, la instrucción WAIT permanecerá en espera hasta que la terminal TEST se convierta en un 1 lógico.

8. **NMI-Interrupción no enmascarable:** Una entrada que causa una interrupción tipo 2 que es llamada al final de la actual instrucción esto cuando dicha terminal esta activa (1 lógico).
9. **RESET-Reinicializar:** Este terminal si se mantiene en alto por al menos cuatro ciclos de reloj, se reinicializara al 8088, Cuando el 8088 es reinicializado, **inicia la ejecución de la instrucción de la localidad de memoria FFFF0H** y **deshabilita futuras interrupciones.**
10. **CLK-Reloj:** Esta entrada proporciona el reloj básico para el 8088. Esta señal posee el 33% del tiempo en alto (1 lógico) y 66% en bajo (0 lógico).
11. **Vcc- Alimentación de Voltaje:** Terminal para la alimentación de +5V, ±10%.
12. **GND-Tierra:** El 8088 posee dos terminales de tierra; ambas deben ser conectadas a tierra para un funcionamiento apropiado.
13. **MN / \overline{MX} -Modo Mínimo/Máximo:** Esta terminal se utiliza para seleccionar la operación del 8088 en modo mínimo cuando esta conectada a +5V y en modo máximo cuando se conecta directamente a tierra.

Terminales de Modo Mínimo: La operación en modo mínimo del 8088 se obtiene mediante la conexión de la terminal MN / \overline{MX} a +5V. No se debe conectar a Vcc mediante un resistor pullup.

1. **IO / \overline{M} -Memoria o Entrada/Salida:** Esta terminal indica cuando el bus de direcciones contiene una dirección de memoria o una dirección de E/S.
2. **\overline{WR} -Escribir:** Se utiliza para indicar que el bus de datos del 8088 contiene un dato valido para ser almacenado en la memoria o enviado a E/S.
3. **\overline{INTA} -Reconocimiento de Interrupción:** Esta terminal responda a un INTR. Durante un requerimiento de interrupción el INTA se convierte en un 0 lógico, indicando que el bus 8088 esta esperando el número de la interrupción a realizar.
4. **ALE-Habilitador del retenedor de direcciones:** Esta terminal se utiliza para indicar que el bus de direcciones contiene una dirección de memoria valida o una dirección de un puerto valido. Esta terminal nunca pasa a estado de alta impedancia.
5. **DT / \overline{R} -Transmitir/Recibir dato:** Una terminal utilizada para control el dirección del flujo del dato mediante una conexión externa a los reforzadores del bus de datos. Esta terminal pasa al estado de alta impedancia durante un reconocimiento de retención.

6. **DEN-Habilitar Bus de Datos:** Esta señal indica que el bus de direcciones/datos contiene una dato valido. Esta terminal pasa al estado de alta impedancia durante un reconocimiento de retención.
7. **HOLD-Retener:** Esta entrada se utiliza para un requerimiento de acceso directo a memoria (DMA). Cuando HOLD es activada, el 8088 flota sus buses de direcciones, datos y control, de esta manera un controlador externo de DMA puede accesar el espacio de memoria y de E/S.
8. **HLDA-Reconocimiento de Retención:** Indica que la terminal HOLD esta en alto y que los buses están en estado de alta impedancia.
9. **SS0-Línea de estado:** Esta terminal junto con las terminales, proporcionan el estado actual del 8088, es decir, para conocer exactamente en que estado esta (obtención de código, escritura de memoria, lectura de memoria, escritura a un puerta de E/S, etc.), ver tabla 1.

Tabla 1. Estado del bus para el 8088

IO / \bar{M}	DT / \bar{R}	$\bar{SS0}$	Función
0	0	0	Reconocimiento de Interrupción
0	0	1	Lectura a Memoria
0	1	0	Escrutura a Memoria
0	1	1	Indica un alto (HALT)
1	0	0	Acceso a código
1	0	1	Lectura de E/S
1	1	0	Escritura a E/S
1	1	1	Permanece pasivo

Terminales de Modo Máximo:

1. **\bar{S}_2 , \bar{S}_1 y \bar{S}_0 -Estado:** Estos bits se utilizan en modo máximo para generar un mejor sistema de señales de control mediante el controlador de bus 8288 (ver tabla 2). Estas terminales pasan a estado de alta impedancia durante un requerimiento del bus.
2. **\bar{RQ}/\bar{GT}_0 y \bar{RQ}/\bar{GT}_1 -Requerimiento/Conceder:** Terminales usados para requerimiento del bus por el coprocesador. Cada terminal es bidireccional y permite que el coprocesador solicite el bus para un DMA (Acceso directo a memoria).
3. **LOCK-Candado:** Terminal de salida que se convierte en 0 lógico para una instrucciones prefijada con LOCK. En general se utiliza para prevenir que un coprocesador externo intente tomar el control de bus durante una instrucción prefijada con LOCK.
4. **QS_1 y QS_0 -Estado de la cola de instrucciones:** Bits que proveen un método para mantener el rastro de cola interna de prebúsqueda. La cola de instrucciones es de 4 bytes de longitud, el coprocesador numérico 8087 utiliza estos bits para sincronizarse con el 8088.

Tabla 2. Funciones de control del bus generadas por el 8288.

\bar{S}_2	\bar{S}_1	\bar{S}_0	Función de control
0	0	0	Reconocimiento de Interrupción
0	0	1	Lectura de E/S
0	1	0	Escritura a E/S
0	1	1	Indica un alto (HALT)
1	0	0	Acceso a código
1	0	1	Lectura a Memoria
1	1	0	Escritura a Memoria
1	1	1	Permanece pasivo

Modo Máximo vs. Modo Mínimo: Existen dos maneras de operar el microprocesador 8088: modo mínimo y modo máximo. El modo mínimo se obtiene conectando la terminal MN / \overline{MX} a $+5V$, y el modo máximo se selecciona conectando esta terminal a tierra. Ambos modos habilitan diferentes estructuras de control para el microprocesador 8088.

Operación en Modo Mínimo

La operación en modo mínimo es la manera más económica de operar el microprocesador 8088. Es más económico debido a que todas las señales de control para la memoria y E/S son generadas dentro del microprocesador.

Operación en Modo Máximo

La diferencia de la operación en este modo con el modo mínimo es que algunas señales de control deben ser generadas externamente. Para ello se requiere un controlador de bus externo (8288). El microprocesador 8088 no posee las terminales suficientes para controlar el bus en la operación de modo máximo esto debido a que nuevas terminales y nuevas funciones remplazan algunas terminales en este modo. El modo máximo se usa solamente cuando el sistema contiene un coprocesador matemático (8087).

Data Sheet(harris)

March 1997

CMOS 8/16-Bit Microprocessor
Features

- Compatible with NMOS 8088
- Direct Software Compatibility with 80C86, 8086, 8088
- 8-Bit Data Bus Interface; 16-Bit Internal Architecture
- Completely Static CMOS Design
 - DC 5MHz (80C88)
 - DC 8MHz (80C88-2)
- Low Power Operation
 - ICCSB 500 μ A Maximum
 - ICCOP 10mA/MHz Maximum
- 1 Megabyte of Direct Memory Addressing Capability
- 24 Operand Addressing Modes
- Bit, Byte, Word, and Block Move Operations
- 8-Bit and 16-Bit Signed/Unsigned Arithmetic
- Bus-Hold Circuitry Eliminates Pull-up Resistors
- Wide Operating Temperature Ranges
 - C80C88 0°C to +70°C
 - I80C88 -40°C to +85°C
 - M80C88 -55°C to +125°C

Description

The Harris 80C88 high performance 8/16-bit CMOS CPU is manufactured using a self-aligned silicon gate CMOS process (Scaled SAJI IV). Two modes of operation, MINimum for small systems and MAXimum for larger applications such as multiprocessing, allow user configuration to achieve the highest performance level.

Full TTL compatibility (with the exception of CLOCK) and industry-standard operation allow use of existing NMOS 8088 hardware and Harris CMOS peripherals.

Complete software compatibility with the 80C86, 8086, and 8088 microprocessors allows use of existing software in new designs.

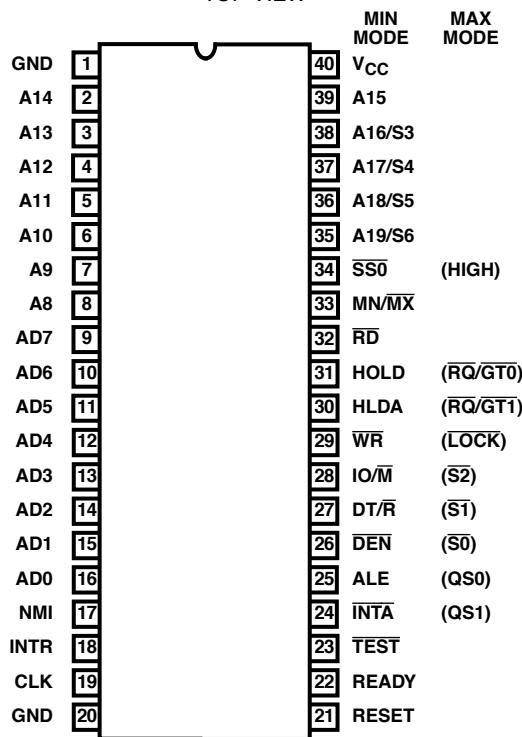
Ordering Information

PACKAGE	TEMPERATURE RANGE	5MHz	8MHz	PKG. NO.
Plastic DIP	0°C to +70°C	CP80C88	CP80C88-2	E40.6
	-40°C to +85°C	IP80C88	IP80C88-2	E40.6
PLCC	0°C to +70°C	CS80C88	CS80C88-2	N44.65
	-40°C to +85°C	IS80C88	IS80C88-2	N44.65
CERDIP	0°C to +70°C	CD80C88	CD80C88-2	F40.6
	-40°C to +85°C	ID80C88	ID80C88-2	F40.6
	-55°C to +125°C	MD80C88/B	MD80C88-2/B	F40.6
SMD#	-55°C to +125°C	5962-8601601QA	-	F40.6
LCC	-55°C to +125°C	MR80C88/B	MR80C88-2/B	J44.A
SMD#	-55°C to +125°C	5962-8601601XA	-	J44.A

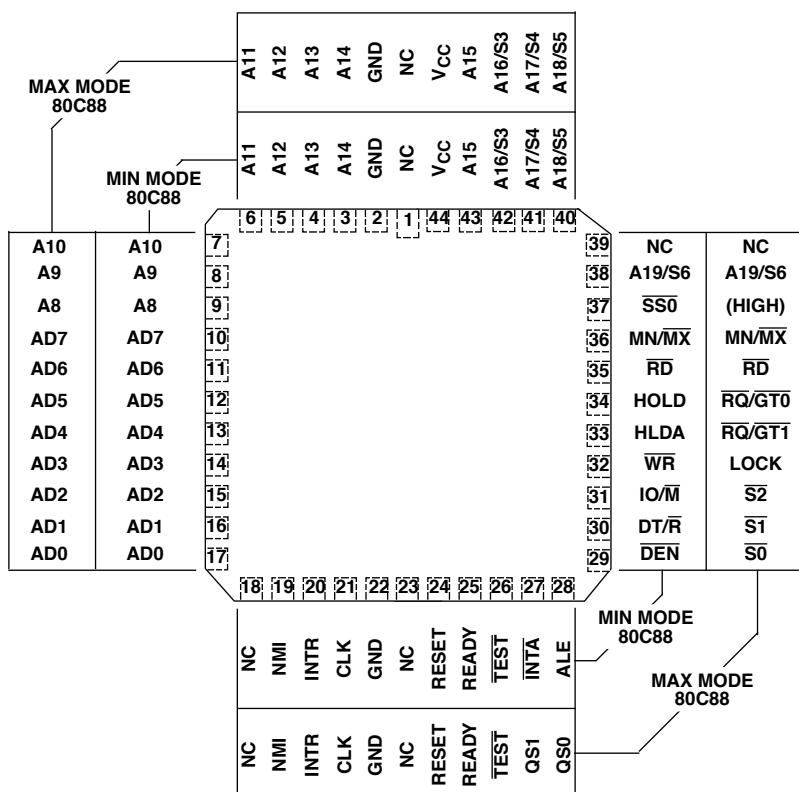
80C88

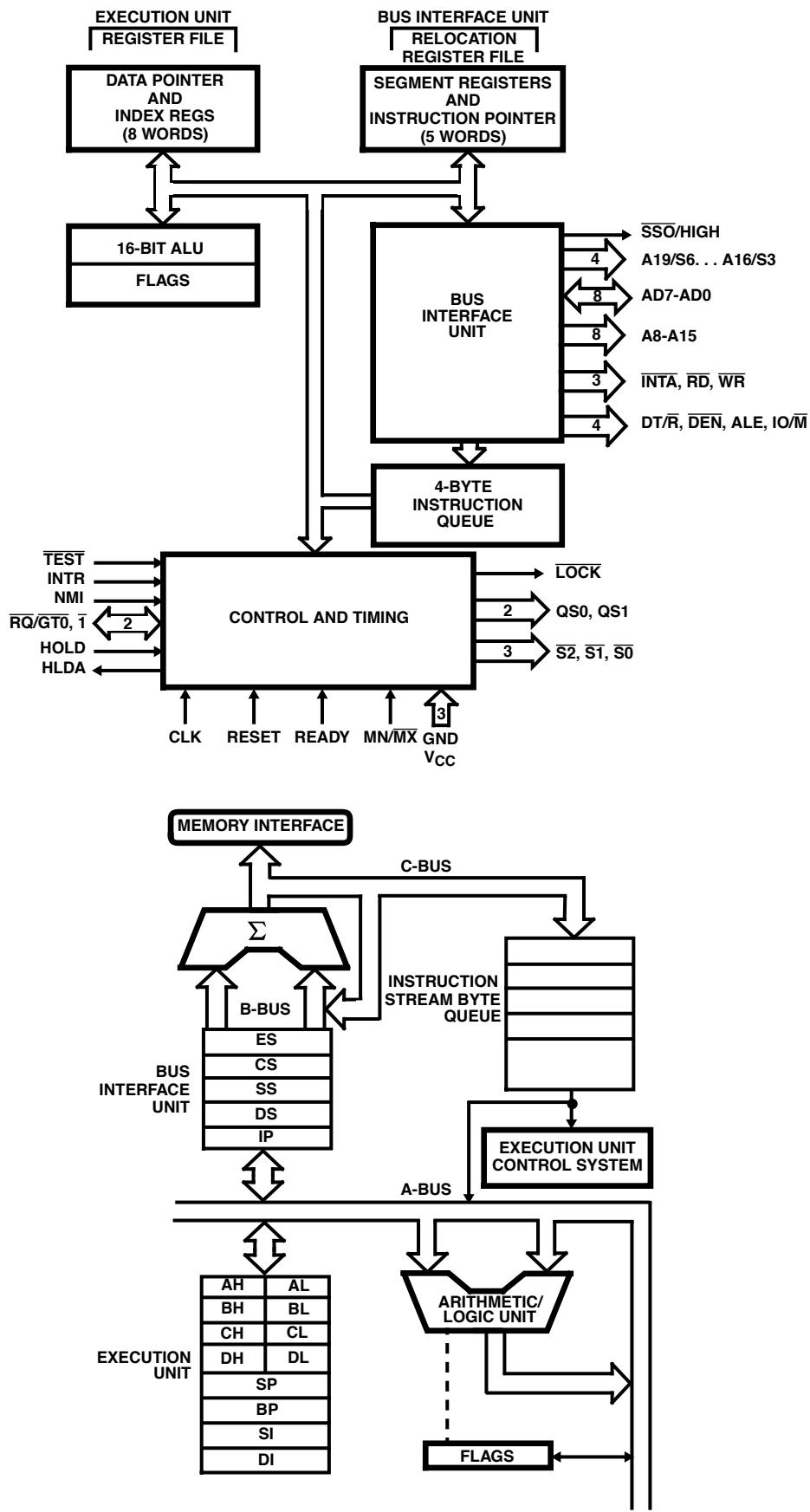
Pinouts

80C88 (DIP)
TOP VIEW



80C88 (PLCC/LCC)
TOP VIEW



Functional Diagram

Pin Description

The following pin function descriptions are for 80C88 systems in either minimum or maximum mode. The "local bus" in these descriptions is the direct multiplexed bus interface connection to the 80C88 (without regard to additional bus buffers).

SYMBOL	PIN NUMBER	TYPE	DESCRIPTION
AD7-AD0	9-16	I/O	ADDRESS DATA BUS: These lines constitute the time multiplexed memory/IO address (T1) and data (T2,T3,Tw and T4) bus. These lines are active HIGH and are held at high impedance to the last valid level during interrupt acknowledge and local bus "hold acknowledge" or "grant sequence"
A15-A8	2-8, 39	O	ADDRESS BUS: These lines provide address bits 8 through 15 for the entire bus cycle (T1-T4). These lines do not have to be latched by ALE to remain valid. A15-A8 are active HIGH and are held at high impedance to the last valid logic level during interrupt acknowledge and local bus "hold acknowledge" or "grant sequence".
A19/S6, A18/S5, A17/S4, A16/S3	35 36 37 38	O	ADDRESS/STATUS: During T1, these are the four most significant address lines for memory operations. During I/O operations, these lines are LOW. During memory and I/O operations, status information is available on these lines during T2, T3, TW and T4. S6 is always LOW. The status of the interrupt enable flag bit (S5) is updated at the beginning of each clock cycle. S4 and S3 are encoded as shown. This information indicates which segment register is presently being used for data accessing. These lines are held at high impedance to the last valid logic level during local bus "hold acknowledge" or "grant Sequence".
RD	32	O	READ: Read strobe indicates that the processor is performing a memory or I/O read cycle, depending on the state of the IO/M pin or $\overline{S2}$. This signal is used to read devices which reside on the 80C88 local bus. RD is active LOW during T2, T3, Tw of any read cycle, and is guaranteed to remain HIGH in T2 until the 80C88 local bus has floated. This line is held at a high impedance logic one state during "hold acknowledge" or "grant sequence".
READY	22	I	READY: is the acknowledgment from the address memory or I/O device that it will complete the data transfer. The RDY signal from memory or I/O is synchronized by the 82C84A clock generator to from READY. This signal is active HIGH. The 80C88 READY input is not synchronized. Correct operation is not guaranteed if the set up and hold times are not met.
INTR	18	I	INTERRUPT REQUEST: is a level triggered input which is sampled during the last clock cycle of each instruction to determine if the processor should enter into an interrupt acknowledge operation. A subroutine is vectored to via an interrupt vector lookup table located in system memory. It can be internally masked by software resetting the interrupt enable bit. INTR is internally synchronized. This signal is active HIGH.
TEST	23	I	TEST: input is examined by the "wait for test" instruction. If the TEST input is LOW, execution continues, otherwise the processor waits in an "idle" state. This input is synchronized internally during each clock cycle on the leading edge of CLK.
NMI	17	I	NONMASKABLE INTERRUPT: is an edge triggered input which causes a type 2 interrupt. A subroutine is vectored to via an interrupt vector lookup table located in system memory. NMI is not maskable internally by software. A transition from a LOW to HIGH initiates the interrupt at the end of the current instruction. This input is internally synchronized.
RESET	21	I	RESET: causes the processor to immediately terminate its present activity. The signal must transition LOW to HIGH and remain active HIGH for at least four clock cycles. It restarts execution, as described in the instruction set description, when RESET returns LOW. RESET is internally synchronized.
CLK	19	I	CLOCK: provides the basic timing for the processor and bus controller. It is asymmetric with a 33% duty cycle to provide optimized internal timing.
V _{CC}	40		V _{CC} : is the +5V power supply pin. A 0.1μF capacitor between pins 20 and 40 recommended for decoupling.
GND	1, 20		GND: are the ground pins (both pins must be connected to system ground). A 0.1μF capacitor between pins 1 and 20 is recommended for decoupling.
MN/MX	33 ¹	I	MINIMUM/MAXIMUM: indicates the mode in which the processor is to operate. The two modes are discussed in the following sections.

CHARACTERISTICS		
S4	S3	CHARACTERISTICS
0	0	Alternate Data
0	1	Stack
1	0	Code or None
1	1	Data

Pin Description (Continued)

The following pin function descriptions are for 80C88 system in minimum mode (i.e., MN/MX = V_{CC}). Only the pin functions which are unique to the minimum mode are described; all other pin functions are as described above.

MINIMUM MODE SYSTEM

SYMBOL	PIN NUMBER	TYPE	DESCRIPTION
IO/M	28	O	STATUS LINE: is an inverted maximum mode $\overline{S_2}$. It is used to distinguish a memory access from an I/O access. IO/M becomes valid in the T4 preceding a bus cycle and remains valid until the final T4 of the cycle (I/O = HIGH, M = LOW). IO/M is held to a high impedance logic one during local bus "hold acknowledge".
WR	29	O	Write: strobe indicates that the processor is performing a write memory or write I/O cycle, depending on the state of the IO/M signal. WR is active for T2, T3, and Tw of any write cycle. It is active LOW, and is held to high impedance logic one during local bus "hold acknowledge".
INTA	24	O	INTA: is used as a read strobe for interrupt acknowledge cycles. It is active LOW during T2, T3 and Tw of each interrupt acknowledge cycle. Note that INTA is never floated.
ALE	25	O	ADDRESS LATCH ENABLE: is provided by the processor to latch the address into the 82C82/82C83 address latch. It is a HIGH pulse active during clock low of T1 of any bus cycle. Note that ALE is never floated.
DT/R	27	O	DATA TRANSMIT/RECEIVE: is needed in a minimum system that desires to use an 82C86/82C87 data bus transceiver. It is used to control the direction of data flow through the transceiver. Logically, DT/R is equivalent to S1 in the maximum mode, and its timing is the same as for IO/M (T = HIGH, R = LOW). This signal is held to a high impedance logic one during local bus "hold acknowledge".
DEN	26	O	DATA ENABLE: is provided as an output enable for the 82C86/82C87 in a minimum system which uses the transceiver. DEN is active LOW during each memory and I/O access, and for INTA cycles. For a read or INTA cycle, it is active from the middle of T2 until the middle of T4, while for a write cycle, it is active from the beginning of T2 until the middle of T4. DEN is held to high impedance logic one during local bus "hold acknowledge".
HOLD, HLDA	31 30	I O	HOLD: indicates that another master is requesting a local bus "hold". To be acknowledged, HOLD must be active HIGH. The processor receiving the "hold" request will issue HLDA (HIGH) as an acknowledgment, in the middle of a T4 or T1 clock cycle. Simultaneous with the issuance of HLDA the processor will float the local bus and control lines. After HOLD is detected as being LOW, the processor lowers HLDA, and when the processor needs to run another cycle, it will again drive the local bus and control lines. Hold is not an asynchronous input. External synchronization should be provided if the system cannot otherwise guarantee the set up time.
SS0	34	O	STATUS LINE: is logically equivalent to $\overline{S_0}$ in the maximum mode. The combination of SS0, IO/M and DT/R allows the system to completely decode the current bus cycle status. SS0 is held to high impedance logic one during local bus "hold acknowledge".

IO/M	DT/R	SS0	CHARACTERISTICS
1	0	0	Interrupt Acknowledge
1	0	1	Read I/O Port
1	1	0	Write I/O Port
1	1	1	Halt
0	0	0	Code Access
0	0	1	Read Memory
0	1	0	Write Memory
0	1	1	Passive

Pin Description (Continued)

The following pin function descriptions are for 80C88 system in maximum mode (i.e., MN/MX = GND). Only the pin functions which are unique to the maximum mode are described; all other pin functions are as described above.

MAXIMUM MODE SYSTEM

SYMBOL	PIN NUMBER	TYPE	DESCRIPTION																																				
$\overline{S_0}$ $\overline{S_1}$ $\overline{S_2}$	26 27 28	O	<p>STATUS: is active during clock high of T4, T1 and T2, and is returned to the passive state (1, 1, 1) during T3 or during Tw when READY is HIGH. This status is used by the 82C88 bus controller to generate all memory and I/O access control signals. Any change by $\overline{S_2}$, $\overline{S_1}$ or $\overline{S_0}$ during T4 is used to indicate the beginning of a bus cycle, and the return to the passive state in T3 or Tw is used to indicate the end of a bus cycle.</p> <p>These signals are held at a high impedance logic one state during "grant sequence".</p>																																				
			<table border="1"> <thead> <tr> <th>$\overline{S_2}$</th><th>$\overline{S_1}$</th><th>$\overline{S_0}$</th><th>CHARACTERISTICS</th></tr> </thead> <tbody> <tr> <td>0</td><td>0</td><td>0</td><td>Interrupt Acknowledge</td></tr> <tr> <td>0</td><td>0</td><td>1</td><td>Read I/O Port</td></tr> <tr> <td>0</td><td>1</td><td>0</td><td>Write I/O Port</td></tr> <tr> <td>0</td><td>1</td><td>1</td><td>Halt</td></tr> <tr> <td>1</td><td>0</td><td>0</td><td>Code Access</td></tr> <tr> <td>1</td><td>0</td><td>1</td><td>Read Memory</td></tr> <tr> <td>1</td><td>1</td><td>0</td><td>Write Memory</td></tr> <tr> <td>1</td><td>1</td><td>1</td><td>Passive</td></tr> </tbody> </table>	$\overline{S_2}$	$\overline{S_1}$	$\overline{S_0}$	CHARACTERISTICS	0	0	0	Interrupt Acknowledge	0	0	1	Read I/O Port	0	1	0	Write I/O Port	0	1	1	Halt	1	0	0	Code Access	1	0	1	Read Memory	1	1	0	Write Memory	1	1	1	Passive
$\overline{S_2}$	$\overline{S_1}$	$\overline{S_0}$	CHARACTERISTICS																																				
0	0	0	Interrupt Acknowledge																																				
0	0	1	Read I/O Port																																				
0	1	0	Write I/O Port																																				
0	1	1	Halt																																				
1	0	0	Code Access																																				
1	0	1	Read Memory																																				
1	1	0	Write Memory																																				
1	1	1	Passive																																				
$\overline{RQ/GT_0}$, $\overline{RQ/GT_1}$	31 30	I/O	<p>REQUEST/GRANT: pins are used by other local bus masters to force the processor to release the local bus at the end of the processor's current bus cycle. Each pin is bidirectional with $\overline{RQ/GT_0}$ having higher priority than $\overline{RQ/GT_1}$. $\overline{RQ/GT}$ has internal bus-hold high circuitry and, if unused, may be left unconnected. The request/grant sequence is as follows (see RQ/GT Timing Sequence):</p> <ol style="list-style-type: none"> 1. A pulse of one CLK wide from another local bus master indicates a local bus request ("hold") to the 80C88 (pulse 1). 2. During a T4 or T1 clock cycle, a pulse one clock wide from the 80C88 to the requesting master (pulse 2), indicates that the 80C88 has allowed the local bus to float and that it will enter the "grant sequence" state at the next CLK. The CPUs bus interface unit is disconnected logically from the local bus during "grant sequence". 3. A pulse one CLK wide from the requesting master indicates to the 80C88 (pulse 3) that the "hold" request is about to end and that the 80C88 can reclaim the local bus at the next CLK. The CPU then enters T4 (or T1 if no bus cycles pending). <p>Each master-master exchange of the local bus is a sequence of three pulses. There must be one idle CLK cycle after bus exchange. Pulses are active LOW.</p> <p>If the request is made while the CPU is performing a memory cycle, it will release the local bus during T4 of the cycle when all the following conjugations are met:</p> <ol style="list-style-type: none"> 1. Request occurs on or before T2. 2. Current cycle is not the low bit of a word. 3. Current cycle is not the first acknowledge of an interrupt acknowledge sequence. 4. A locked instruction is not currently executing. <p>If the local bus is idle when the request is made the two possible events will follow:</p> <ol style="list-style-type: none"> 1. Local bus will be released during the next clock. 2. A memory cycle will start within 3 clocks. Now the four rules for a currently active memory cycle apply with condition number 1 already satisfied. 																																				
LOCK	29	O	LOCK: indicates that other system bus masters are not to gain control of the system bus while LOCK is active (LOW). The LOCK signal is activated by the "LOCK" prefix instruction and remains active until the completion of the next instruction. This signal is active LOW, and is held at a high impedance logic one state during "grant sequence". In Max Mode, LOCK is automatically generated during T2 of the first INTA cycle and removed during T2 of the second INTA cycle.																																				
QS1, QS0	24, 25	O	<p>QUEUE STATUS: provide status to allow external tracking of the internal 80C88 instruction queue.</p> <p>The queue status is valid during the CLK cycle after which the queue operation is performed. Note that the queue status never goes to a high impedance state (floated).</p> <table border="1"> <thead> <tr> <th>QS1</th><th>QS0</th><th>CHARACTERISTICS</th></tr> </thead> <tbody> <tr> <td>0</td><td>0</td><td>No Operation</td></tr> <tr> <td>0</td><td>1</td><td>First Byte of Opcode from Queue</td></tr> <tr> <td>1</td><td>0</td><td>Empty the Queue</td></tr> <tr> <td>1</td><td>1</td><td>Subsequent Byte from Queue</td></tr> </tbody> </table>	QS1	QS0	CHARACTERISTICS	0	0	No Operation	0	1	First Byte of Opcode from Queue	1	0	Empty the Queue	1	1	Subsequent Byte from Queue																					
QS1	QS0	CHARACTERISTICS																																					
0	0	No Operation																																					
0	1	First Byte of Opcode from Queue																																					
1	0	Empty the Queue																																					
1	1	Subsequent Byte from Queue																																					
-	34	O	Pin 34 is always a logic one in the maximum mode and is held at a high impedance logic one during a "grant sequence".																																				

Functional Description

Static Operation

All 80C88 circuitry is static in design. Internal registers, counters and latches are static and require not refresh as with dynamic circuit design. This eliminates the minimum operating frequency restriction placed on other microprocessors. The CMOS 80C88 can operate from DC to the specified upper frequency limit. The processor clock may be stopped in either state (high/low) and held there indefinitely. This type of operation is especially useful for system debug or power critical applications.

The 80C88 can be single stepped using only the CPU clock. This state can be maintained as long as is necessary. Single step clock operation allows simple interface circuitry to provide critical information for start-up.

Static design also allows very low frequency operation (as low as DC). In a power critical situation, this can provide extremely low power operation since 80C88 power dissipation is directly related to operation frequency. As the system frequency is reduced, so is the operating power until, at a DC input frequency, the power requirement is the 80C88 standby current.

Internal Architecture

The internal functions of the 80C88 processor are partitioned logically into two processing units. The first is the Bus Interface Unit (BIU) and the second is the Execution Unit (EU) as shown in the CPU block diagram.

These units can interact directly but for the most part perform as separate asynchronous operational processors. The bus interface unit provides the functions related to instruction fetching and queuing, operand fetch and store, and address relocation. This unit also provides the basic bus control. The overlap of instruction pre-fetching provided by this unit serves to increase processor performance through improved bus bandwidth utilization. Up to 4 bytes of the instruction stream can be queued while waiting for decoding and execution.

The instruction stream queuing mechanism allows the BIU to keep the memory utilized very efficiently. Whenever there is space for at least 1 byte in the queue, the BIU will attempt a byte fetch memory cycle. This greatly reduces "dead time": on the memory bus. The queue acts as a First-In-First-Out (FIFO) buffer, from which the EU extracts instruction bytes as required. If the queue is empty (following a branch instruction, for example), the first byte into the queue immediately becomes available to the EU.

The execution unit receives pre-fetched instructions from the BIU queue and provides unrellocated operand addresses to the BIU. Memory operands are passed through the BIU for processing by the EU, which passes results to the BIU for storage.

Memory Organization

The processor provides a 20-bit address to memory which locates the byte being referenced. The memory is organized

as a linear array of up to 1 million bytes, addressed as 00000(H) to FFFFF(H). The memory is logically divided into code, data, extra, and stack segments of up to 64K bytes each, with each segment falling on 16 byte boundaries. (See Figure 1).

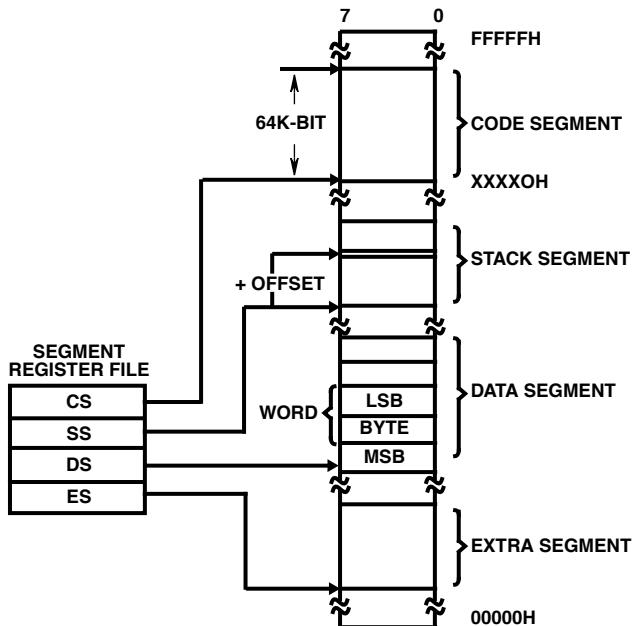


FIGURE 1. MEMORY ORGANIZATION

All memory references are made relative to base addresses contained in high speed segment registers. The segment types were chosen based on the addressing needs of programs. The segment register to be selected is automatically chosen according to specific rules as shown in Table 1. All information in one segment type share the same logical attributes (e.g., code or data). By structuring memory into relocatable areas of similar characteristics and by automatically selecting segment registers, programs are shorter, faster, and more structured.

TABLE 1.

MEMORY REFERENCE NEED	SEGMENT REGISTER USED	SEGMENT SELECTION RULE
Instructions	CODE (CS)	Automatic with all instruction prefetch.
Stack	STACK (SS)	All stack pushes and pops. Memory references relative to BP base register except data references.
Local Data	DATA (DS)	Data references when: relative to stack, destination of string operation, or explicitly overridden.
External Data (Global)	EXTRA (ES)	Destination of string operations: Explicitly selected using a segment override.

Word (16-bit) operands can be located on even or odd address boundaries. For address and data operands, the least significant byte of the word is stored in the lower valued address location and the most significant byte in the next higher address location.

The BIU will automatically execute two fetch or write cycles for 16-bit operands.

Certain locations in memory are reserved for specific CPU operations. (See Figure 2). Locations from addresses FFFF0H through FFFFFH are reserved for operations including a jump to initial system initialization routine. Following RESET, the CPU will always begin execution at location FFFF0H where the jump must be located. Locations 00000H through 003FFH are reserved for interrupt operations. Each of the 256 possible interrupt service routines is accessed through its own pair of 16-bit pointers - segment address pointer and offset address pointer. The first pointer, used as the offset address, is loaded into the IP, and the second pointer, which designates the base address, is loaded into the CS. At this point program control is transferred to the interrupt routine. The pointer elements are assumed to have been stored at their respective places in reserved memory prior to the occurrence of interrupts.

Minimum and Maximum Modes

The requirements for supporting minimum and maximum 80C88 systems are sufficiently different that they cannot be done efficiently with 40 uniquely defined pins. Consequently, the 80C88 is equipped with a strap pin (MN/MX) which defines the system configuration. The definition of a certain subset of the pins changes, dependent on the condition of

the strap pin. When the MN/MX pin is strapped to GND, the 80C88 defines pins 24 through 31 and 34 in maximum mode. When the MN/MX pins is strapped to V_{CC}, the 80C88 generates bus control signals itself on pins 24 through 31 and 34.

The minimum mode 80C88 can be used with either a multiplexed or demultiplexed bus. This architecture provides the 80C88 processing power in a highly integrated form.

The demultiplexed mode requires one latch (for 64K addressability) or two latches (for a full megabyte of addressing). An 82C86 or 82C87 transceiver can also be used if data bus buffering is required. (See Figure 3). The 80C88 provides DEN and DT/R to control the transceiver, and ALE to latch the addresses. This configuration of the minimum mode provides the standard demultiplexed bus structure with heavy bus buffering and relaxed bus timing requirements.

The maximum mode employs the 82C88 bus controller (See Figure 4). The 82C88 decode status lines S0, S1 and S2, and provides the system with all bus control signals. Moving the bus control to the 82C88 provides better source and sink current capability to the control lines, and frees the 80C88 pins for extended large system features. Hardware lock, queue status, and two request/grant interfaces are provided by the 80C88 in maximum mode. These features allow coprocessors in local bus and remote bus configurations.

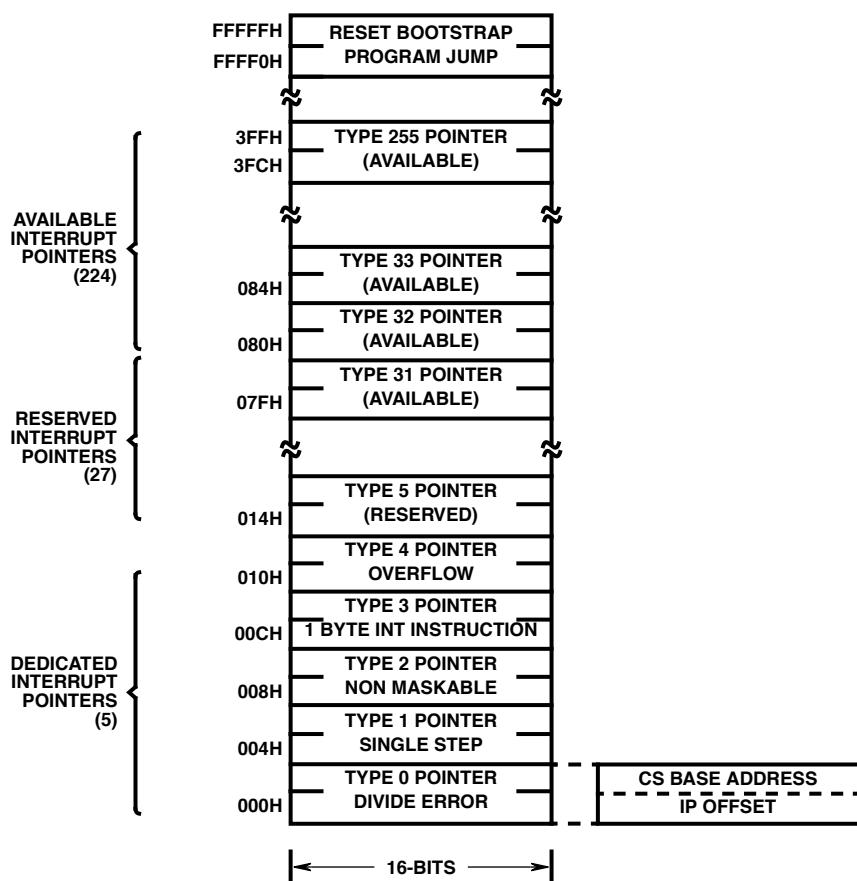
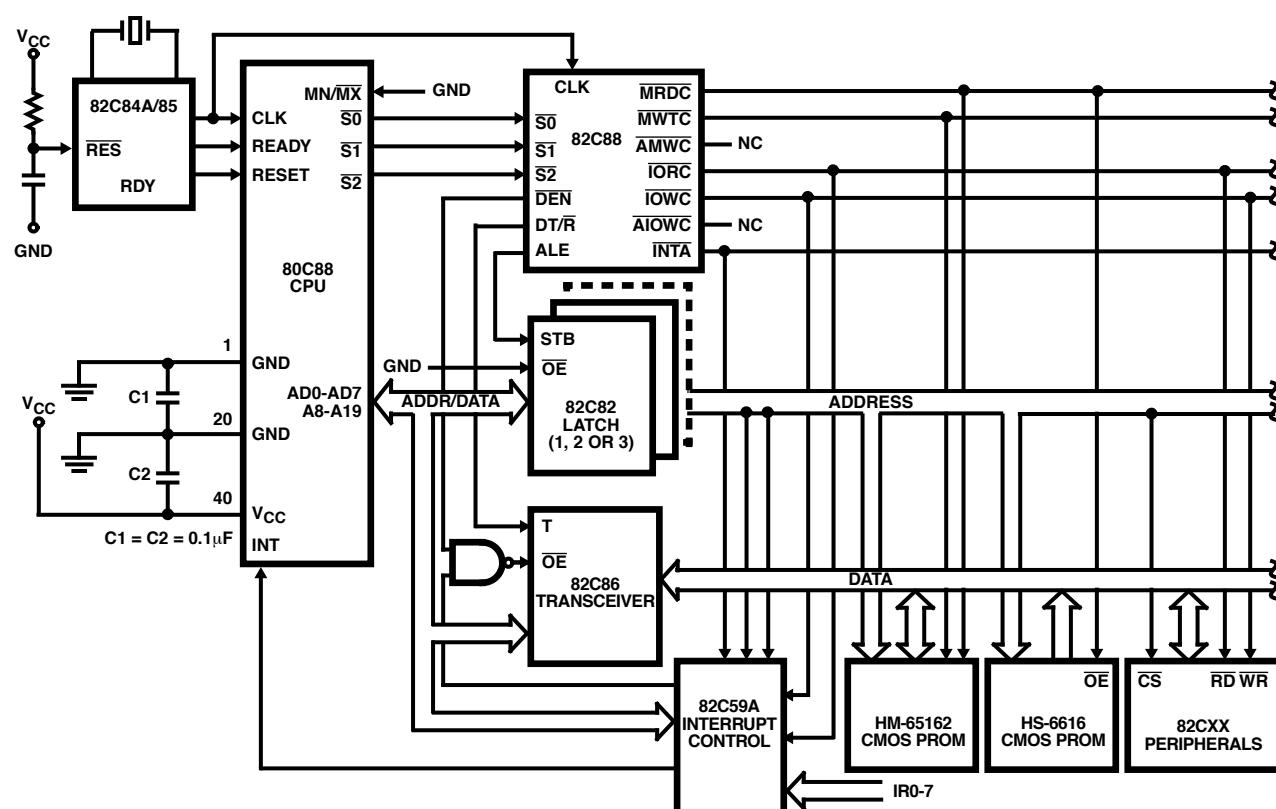
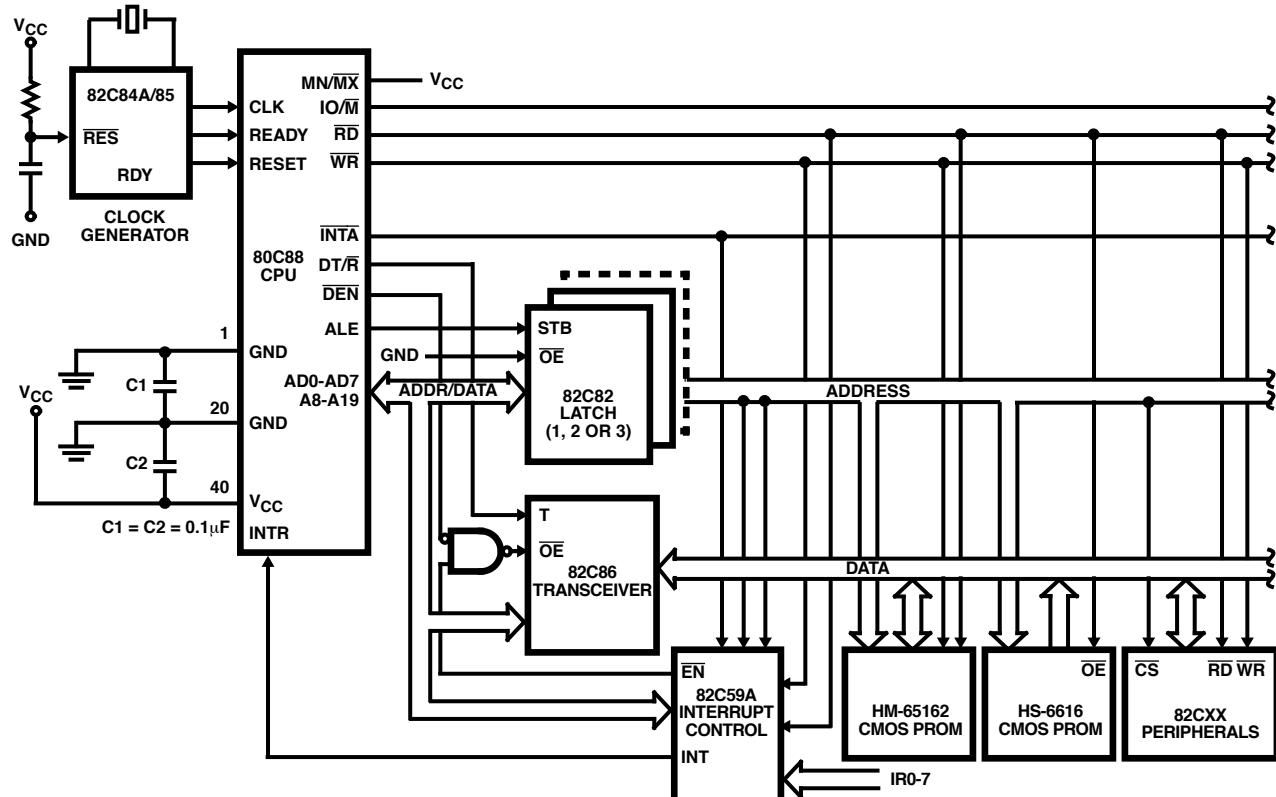


FIGURE 2. RESERVED MEMORY LOCATIONS



Bus Operation

The 80C88 address/data bus is broken into three parts: the lower eight address/data bits (AD0-AD7), the middle eight address bits (A8-A15), and the upper four address bits (A16-A19). The address/data bits and the highest four address bits are time multiplexed. This technique provides the most efficient use of pins on the processor, permitting the use of standard 40 lead package. The middle eight address bits are not multiplexed, i.e., they remain valid throughout each bus cycle. In addition, the bus can be demultiplexed at the processor with a single address latch if a standard, nonmultiplexed bus is desired for the system.

Each processor bus cycle consists of at least four CLK cycles. These are referred to as T1, T2, T3 and T4. (See Figure 5). The address is emitted from the processor during T1 and data transfer occurs on the bus during T3 and T4. T2 is used primarily for changing the direction of the bus during read operations. In the event that a "Not Ready" indication is given by the addressed device, "wait" states (TW) are inserted between T3 and T4. Each inserted "wait" state is of

the same duration as a CLK cycle. Periods can occur between 80C88 driven bus cycles. These are referred to as "idle" states (T1), or inactive CLK cycles. The processor uses these cycles for internal housekeeping.

During T1 of any bus cycle, the ALE (Address latch enable) signal is emitted (by either the processor or the 82C88 bus controller, depending on the MN/MX strap). At the trailing edge of this pulse, a valid address and certain status information for the cycle may be latched.

Status bits $\overline{S0}$, $\overline{S1}$, and $\overline{S2}$ are used by the bus controller, in maximum mode, to identify the type of bus transaction according to Table 2.

Status bits S3 through S6 are multiplexed with high order address bits and are therefore valid during T2 through T4. S3 and S4 indicate which segment register was used to this bus cycle in forming the address according to Table 3.

S5 is a reflection of the PSW interrupt enable bit. S6 is always equal to 0.

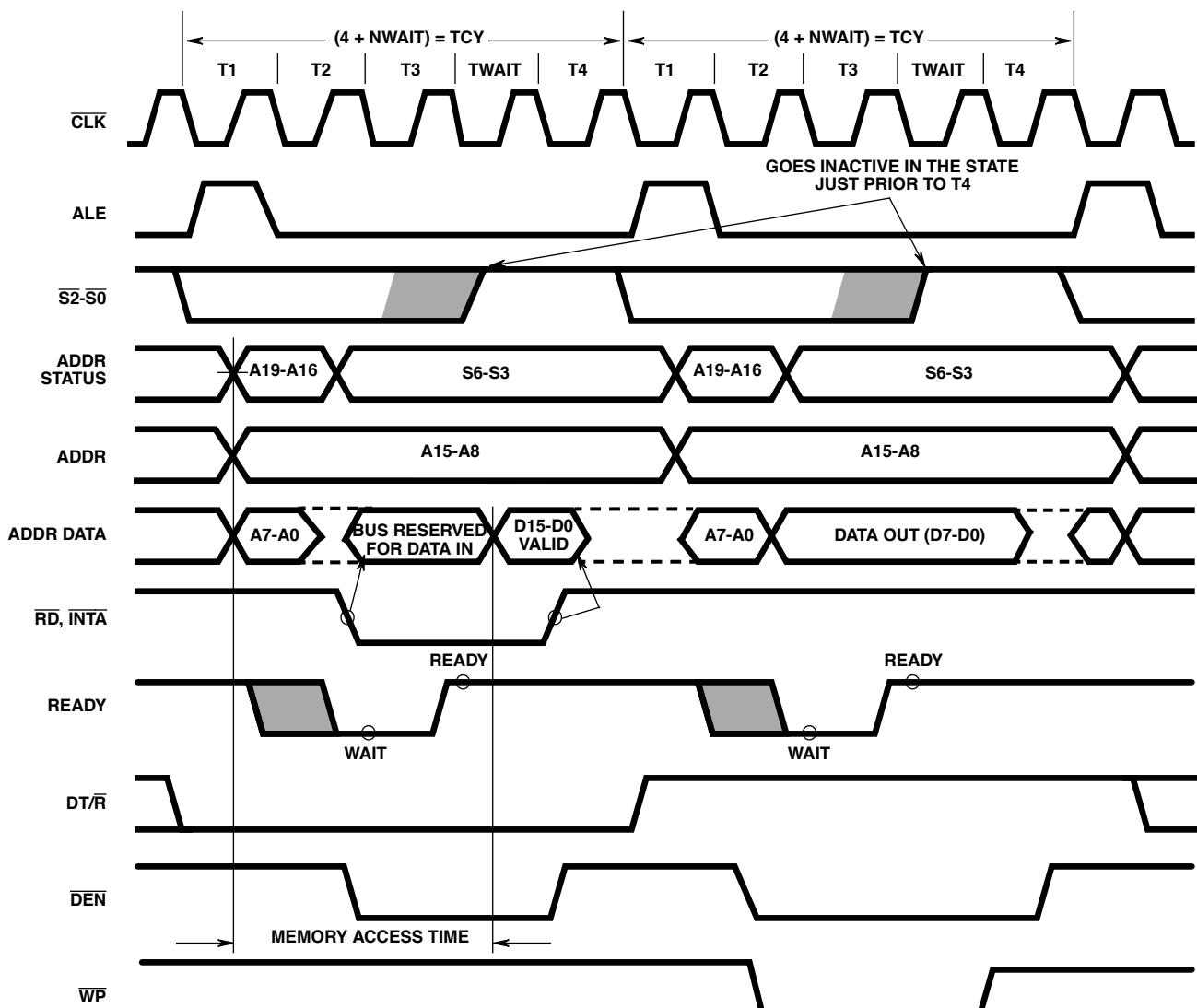


FIGURE 5. BASIC SYSTEM TIMING

TABLE 2.

S2	S1	S0	CHARACTERISTICS
0	0	0	Interrupt Acknowledge
0	0	1	Read I/O
0	1	0	Write I/O
0	1	1	Halt
1	0	0	Instruction Fetch
1	0	1	Read Data from Memory
1	1	0	Write Data to Memory
1	1	1	Passive (No Bus Cycle)

TABLE 3.

S4	S3	CHARACTERISTICS
0	0	Alternate Data (Extra Segment)
0	1	Stack
1	0	Code or None
1	1	Data

I/O Addressing

In the 80C88, I/O operations can address up to a maximum of 64K I/O registers. The I/O address appears in the same format as the memory address on bus lines A15-A0. The address lines A19-A16 are zero in I/O operations. The variable I/O instructions, which use register DX as a pointer, have full address capability, while the direct I/O instructions directly address one or two of the 256 I/O byte locations in page 0 of the I/O address space. I/O ports are addressed in the same manner as memory locations.

Designers familiar with the 8085 or upgrading an 8085 design should note that the 8085 addresses I/O with an 8-bit address on both halves of the 16-bit address bus. The 80C88 uses a full 16-bit address on its lower 16 address lines.

External Interface

Processor Reset and Initialization

Processor initialization or start up is accomplished with activation (HIGH) of the RESET pin. The 80C88 RESET is required to be HIGH for greater than four clock cycles. The 80C88 will terminate operations on the high-going edge of RESET and will remain dormant as long as RESET is HIGH. The low-going transition of RESET triggers an internal reset sequence for approximately 7 clock cycles. After this interval the 80C88 operates normally, beginning with the instruction in absolute location FFFF0H (see Figure 2). The RESET input is internally synchronized to the processor clock. At initialization, the HIGH to LOW transition of RESET must occur no sooner than 50µs after power up, to allow complete initialization of the 80C88.

NMI will not be recognized if asserted prior to the second CLK cycle following the end of RESET.

Bus Hold Circuitry

To avoid high current conditions caused by floating inputs to CMOS devices and to eliminate the need for pull-up/down resistors, "bus-hold" circuitry has been used on 80C88 pins 2-16, 26-32 and 34-39 (see Figure 6A and 6B). These circuits maintain a valid logic state if no driving source is present (i.e., an unconnected pin or a driving source which goes to a high impedance state).

To override the "bus hold" circuits, an external driver must be capable of supplying 400µA minimum sink or source current at valid input voltage levels. Since this "bus hold" circuitry is active and not a "resistive" type element, the associated power supply current is negligible. Power dissipation is significantly reduced when compared to the use of passive pull-up resistors.

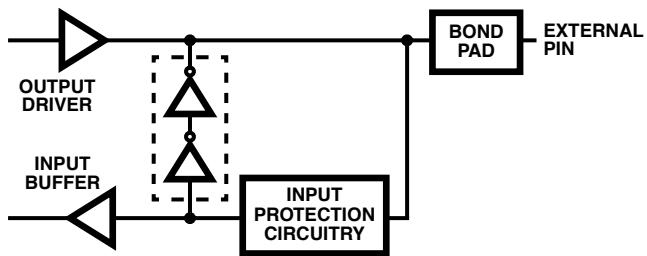


FIGURE 6A. BUS HOLD CIRCUITRY PIN 2-16, 35-39

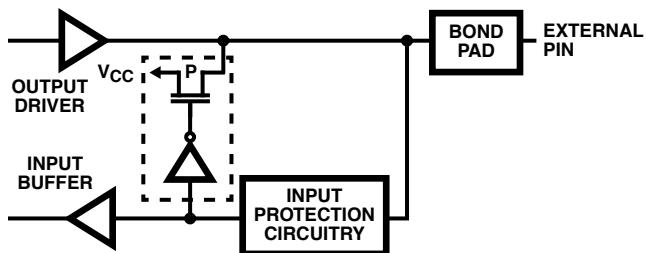


FIGURE 6B. BUS HOLD CIRCUITRY PIN 26-32, 34

Interrupt Operations

Interrupt operations fall into two classes: software or hardware initiated. The software initiated interrupts and software aspects of hardware interrupts are specified in the instruction set description. Hardware interrupts can be classified as nonmaskable or maskable.

Interrupts result in a transfer of control to a new program location. A 256 element table containing address pointers to the interrupt service program locations resides in absolute locations 0 through 3FFH (see Figure 2), which are reserved for this purpose. Each element in the table is 4 bytes in size and corresponds to an interrupt "type". An interrupting device supplies an 8-bit type number, during the interrupt acknowledge sequence, which is used to vector through the appropriate element to the new interrupt service program location.

Non-Maskable Interrupt (NMI)

The processor provides a single non-maskable interrupt (NMI) pin which has higher priority than the maskable interrupt request (INTR) pin. A typical use would be to activate a power failure routine. The NMI is edge-triggered on a LOW to High transition. The activation of this pin causes a type 2 interrupt.

NMI is required to have a duration in the HIGH state of greater than two clock cycles, but is not required to be synchronized to the clock. An high going transition of NMI is latched on-chip and will be serviced at the end of the current instruction or between whole moves (2 bytes in the case of word moves) of a block type instruction. Worst case response to NMI would be for multiply, divide, and variable shift instructions. There is no specification on the occurrence of the low-going edge; it may occur before, during, or after the servicing of NMI. Another high-going edge triggers another response if it occurs after the start of the NMI procedure.

The signal must be free of logical spikes in general and be free of bounces on the low-going edge to avoid triggering extraneous responses.

Maskable Interrupt (INTR)

The 80C88 provides a single interrupt request input (INTR) which can be masked internally by software with the resetting of the interrupt enable (IF) flag bit. The interrupt request signal is level triggered. It is internally synchronized during each clock cycle on the high-going edge of CLK.

To be responded to, INTR must be present (HIGH) during the clock period preceding the end of the current instruction or the end of a whole move for a block type instruction. INTR may be removed anytime after the falling edge of the first INTA signal. During interrupt response sequence, further interrupts are disabled. The enable bit is reset as part of the response to any interrupt (INTR, NMI, software interrupt, or single step). The FLAGS register, which is automatically pushed onto the stack, reflects the state of the processor prior to the interrupt. The enable bit will be zero until the old FLAGS register is restored, unless specifically set by an instruction.

During the response sequence (see Figure 7), the processor executes two successive (back-to-back) interrupt acknowledge cycles. The 80C88 emits a LOCK signal (maximum mode only) from T2 of the first bus cycle until T2 of the second. A local bus "hold" request will not be honored until the end of the second bus cycle. In the second bus cycle, a byte is fetched from the external interrupt system (e.g., 82C59A PIC) which identifies the source (type) of the interrupt. This byte is multiplied by four and used as a pointer into the interrupt vector lookup table.

An INTR signal left HIGH will be continually responded to within the limitations of the enable bit and sample period. INTR may be removed anytime after the falling edge of the first INTA signal. The interrupt return instruction includes a flags pop which returns the status of the original interrupt enable bit when it restores the flags.

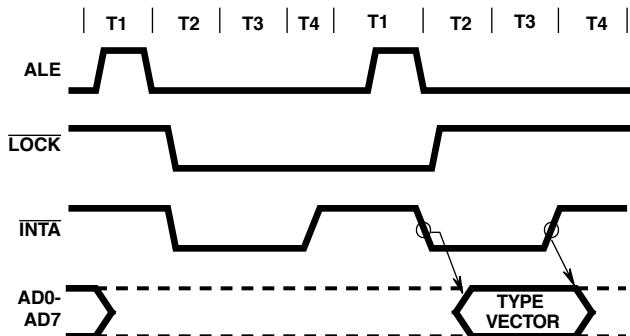


FIGURE 7. INTERRUPT ACKNOWLEDGE SEQUENCE

Halt

When a software HALT instruction is executed, the processor indicates that it is entering the HALT state in one of two ways, depending upon which mode is strapped. In minimum mode, the processor issues ALE, delayed by one clock cycle, to allow the system to latch the halt status. Halt status is available on IO/M, DT/R, and SS0. In maximum mode, the processor issues appropriate HALT status on S2, S1 and S0, and the 82C88 bus controller issues one ALE. The 80C88 will not leave the HALT state when a local bus hold is entered while in HALT. In this case, the processor reissues the HALT indicator at the end of the local bus hold. An interrupt request or RESET will force the 80C88 out of the HALT state.

Read/Modify/Write (Semaphore) Operations Via LOCK

The LOCK status information is provided by the processor when consecutive bus cycles are required during the execution of an instruction. This allows the processor to perform read/modify/write operations on memory (via the "exchange register with memory" instruction), without another system bus master receiving intervening memory cycles. This is useful in multiprocessor system configurations to accomplish "test and set lock" operations. The LOCK signal is activated (LOW) in the clock cycle following decoding of the LOCK prefix instruction. It is deactivated at the end of the last bus cycle of the instruction following the LOCK prefix. While LOCK is active, a request on a RQ/GT pin will be recorded, and then honored at the end of the LOCK.

External Synchronization Via TEST

As an alternative to interrupts, the 80C88 provides a single software-testable input pin (TEST). This input is utilized by executing a WAIT instruction. The single WAIT instruction is repeatedly executed until the TEST input goes active (LOW). The execution of WAIT does not consume bus cycles once the queue is full.

If a local bus request occurs during WAIT execution, the 80C88 three-states all output drivers while inputs and I/O pins are held at valid logic levels by internal bus-hold circuits. If interrupts are enabled, the 80C88 will recognize interrupts and process them when it regains control of the bus.

Basic System Timing

In minimum mode, the MN/MX pin is strapped to V_{CC} and the processor emits bus control signals (\overline{RD} , \overline{WR} , IO/M, etc.) directly. In maximum mode, the MN/MX pin is strapped to GND and the processor emits coded status information which the 82C88 bus controller uses to generate MULTIBUS™ compatible bus control signals.

System Timing - Minimum System

The read cycle begins in T1 with the assertion of the address latch enable (ALE) signal (see Figure 5). The trailing (low going) edge of this signal is used to latch the address information, which is valid on the address data bus (ADO-AD7) at this time, into the 82C82/82C83 latch. Address lines A8 through A15 do not need to be latched because they remain valid throughout the bus cycle. From T1 to T4 the IO/M signal indicates a memory or I/O operation. At T2 the address is removed from the address data bus and the bus is held at the last valid logic state by internal bus-hold devices. The read control signal is also asserted at T2. The read (\overline{RD}) signal causes the addressed device to enable its data bus drivers to the local bus. Some time later, valid data will be available on the bus and the addressed device will drive the READY line HIGH. When the processor returns the read signal to a HIGH level, the addressed device will again three-state its bus drivers. If a transceiver (82C86/82C87) is required to buffer the local bus, signals DT/R and \overline{DEN} are provided by the 80C88.

A write cycle also begins with the assertion of ALE and the emission of the address. The IO/M signal is again asserted to indicate a memory or I/O write operation. In T2, immediately following the address emission, the processor emits the data to be written into the addressed location. This data remains valid until at least the middle of T4. During T2, T3, and Tw, the processor asserts the write control signal. The write (\overline{WR}) signal becomes active at the beginning of T2, as opposed to the read, which is delayed somewhat into T2 to provide time for output drivers to become inactive.

The basic difference between the interrupt acknowledge cycle and a read cycle is that the interrupt acknowledge (INTA) signal is asserted in place of the read (\overline{RD}) signal and the address bus is held at the last valid logic state by internal bus-hold devices (see Figure 6). In the second of two successive INTA cycles, a byte of information is read from the data bus, as supplied by the interrupt system logic (i.e., 82C59A priority interrupt controller). This byte identifies the source (type) of the interrupt. It is multiplied by four and used as a pointer into the interrupt vector lookup table, as described earlier.

Bus Timing - Medium Complexity Systems

For medium complexity systems, the MN/MX pin is connected to GND and the 82C88 bus controller is added to the system, as well as an 82C82/82C83 latch for latching the system address, and an 82C86/82C87 transceiver to allow for bus loading greater than the 80C88 is capable of handling (see Figure 8). Signals ALE, \overline{DEN} , and DT/R are generated by the 82C88 instead of the processor in this

configuration, although their timing remains relatively the same. The 80C88 status outputs ($\overline{S2}$, $\overline{S1}$ and $\overline{S0}$) provide type of cycle information and become 82C88 inputs. This bus cycle information specifies read (code, data or I/O), write (data or I/O), interrupt acknowledge, or software halt. The 82C88 thus issues control signals specifying memory read or write, I/O read or write, or interrupt acknowledge. The 82C88 provides two types of write strobes, normal and advanced, to be applied as required. The normal write strobes have data valid at the leading edge of write. The advanced write strobes have the same timing as read strobes, and hence, data is not valid at the leading edge of write. The 82C86/82C87 transceiver receives the usual T and \overline{OE} inputs from the 82C88 DT/R and \overline{DEN} outputs.

The pointer into the interrupt vector table, which is passed during the second INTA cycle, can derive from an 82C59A located on either the local bus or the system bus. If the master 82C59A priority interrupt controller is positioned on the local bus, the 82C86/82C87 transceiver must be disabled when reading from the master 82C59A during the interrupt acknowledge sequence and software "poll".

The 80C88 Compared to the 80C86

The 80C88 CPU is a 8-bit processor designed around the 8086 internal structure. Most internal functions of the 80C88 are identical to the equivalent 80C86 functions. The 80C88 handles the external bus the same way the 80C86 does with the distinction of handling only 8-bits at a time. Sixteen-bit operands are fetched or written in two consecutive bus cycles. Both processors will appear identical to the software engineer, with the exception of execution time. The internal register structure is identical and all instructions have the same end result. Internally, there are three differences between the 80C88 and the 80C86. All changes are related to the 8-bit bus interface.

- The queue length is 4 bytes in the 80C88, whereas the 80C86 queue contains 6 bytes, or three words. The queue was shortened to prevent overuse of the bus by the BIU when prefetching instructions. This was required because of the additional time necessary to fetch instructions 8-bits at a time.
- To further optimize the queue, the prefetching algorithm was changed. The 80C88 BIU will fetch a new instruction to load into the queue each time there is a 1 byte space available in the queue. The 80C86 waits until a 2 byte space is available.
- The internal execution time of the instruction set is affected by the 8-bit interface. All 16-bit fetches and writes from/to memory take an additional four clock cycles. The CPU is also limited by the speed of instruction fetches. This latter problem only occurs when a series of simple operations occur. When the more sophisticated instructions of the 80C88 are being used, the queue has time to fill the execution proceeds as fast as the execution unit will allow.

The 80C88 and 80C86 are completely software compatible by virtue of their identical execution units. Software that is system dependent may not be completely transferable, but software that is not system dependent will operate equally as well on an 80C88 or an 80C86.

The hardware interface of the 80C88 contains the major differences between the two CPUs. The pin assignments are nearly identical, however, with the following functional changes:

- A8-A15: These pins are only address outputs on the 80C88. These address lines are latched internally and remain valid throughout a bus cycle in a manner similar to the 8085 upper address lines.

- BHE has no meaning on the 80C88 and has been eliminated.
- $\overline{SS_0}$ provides the $\overline{S_0}$ status information in the minimum mode. This output occurs on pin 34 in minimum mode only. DT/R, IO/M and $\overline{SS_0}$ provide the complete bus status in minimum mode.
- IO/M has been inverted to be compatible with the 8085 bus structure.
- ALE is delayed by one clock cycle in the minimum mode when entering HALT, to allow the status to be latched with ALE.

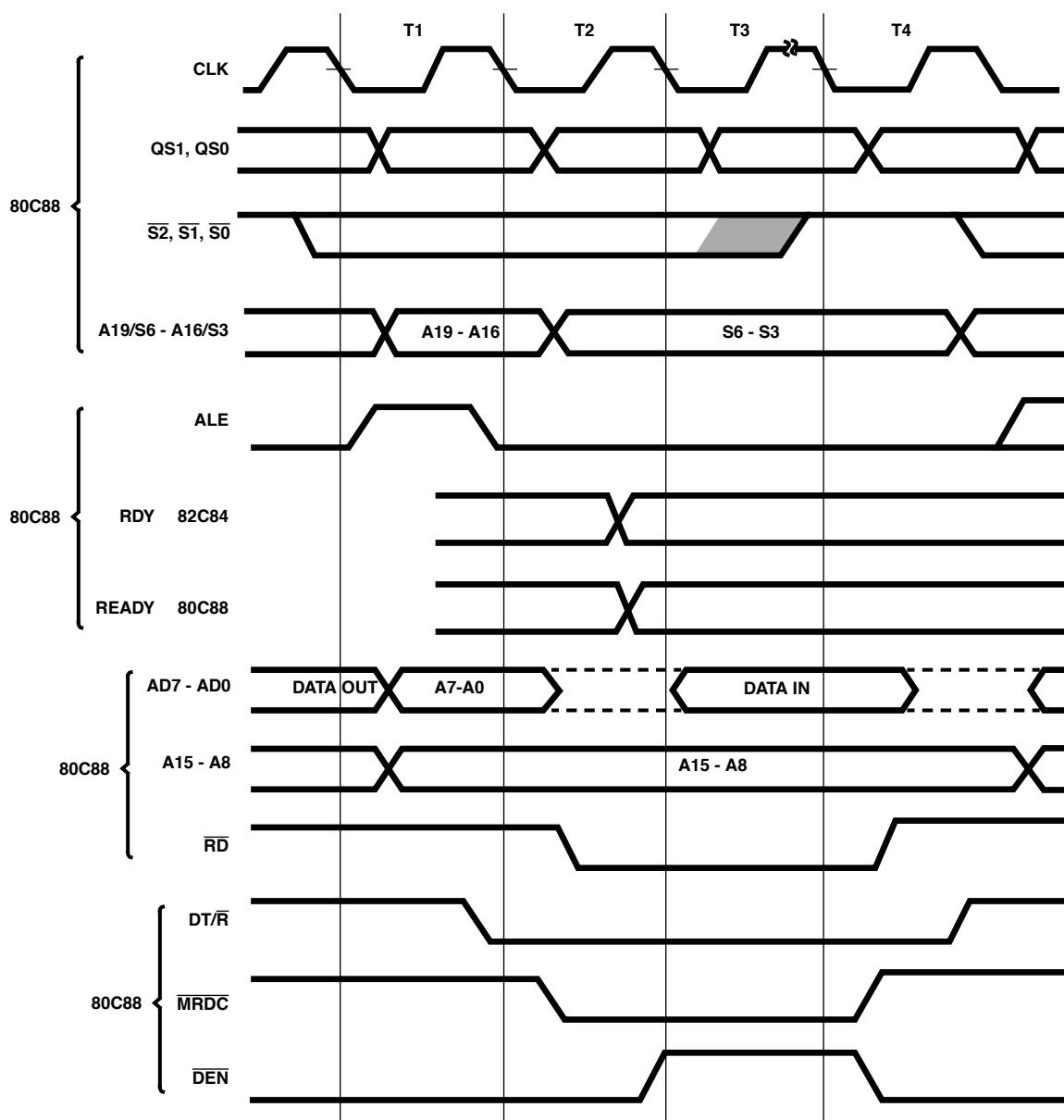


FIGURE 8. MEDIUM COMPLEXITY SYSTEM TIMING

Absolute Maximum Ratings

Supply Voltage	+8.0V
Input, Output or I/O Voltage	GND -0.5V to V_{CC} +0.5V
ESD Classification	Class 1

Operating Conditions

Operating Voltage Range	+4.5V to +5.5V
M80C88-2 Only.....	+4.75V to +5.25V
Operating Temperature Range	
C80C88/-2.....	0°C to +70°C
I80C88/-2	-40°C to +85°C
M80C88/-2	-55°C to +125°C

Thermal Information

Thermal Resistance (Typical)	θ_{JA} (°C/W)	θ_{JC} (°C/W)
PDIP Package	50	-
PLCC Package	46	-
SBDIP Package.....	30	N/A
CLCC Package	40	N/A
Maximum Junction Temperature		
Ceramic Package	+175°C	
Plastic Package	+150°C	
Storage Temperature Range.....	-65°C to +150°C	
Maximum Lead Temperature (Soldering 10s).....	+300°C	
(Lead tips only for surface mount packages)		

Die Characteristics

Gate Count	9750 Gates
------------------	------------

CAUTION: Stresses above those listed in "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress only rating and operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied.

DC Electrical Specifications

$V_{CC} = 5.0V, \pm 10\%$; $T_A = 0^{\circ}C$ to $+70^{\circ}C$ (C80C88, C80C88-2)
$V_{CC} = 5.0V, \pm 10\%$; $T_A = -40^{\circ}C$ to $+85^{\circ}C$ (I80C88, I80C88-2)
$V_{CC} = 5.0V, \pm 10\%$; $T_A = -55^{\circ}C$ to $+125^{\circ}C$ (M80C88)
$V_{CC} = 5.0V, \pm 5\%$; $T_A = -55^{\circ}C$ to $+125^{\circ}C$ (M80C88-2)

SYMBOL	PARAMETER	MIN	MAX	UNITS	TEST CONDITION
V_{IH}	Logical One Input Voltage	2.0 2.2	-	V V	C80C88, I80C88 (Note 4) M80C88 (Note 4)
V_{IL}	Logical Zero Input Voltage	-	0.8	V	
V_{IHC}	CLK Logical One Input Voltage	V_{CC} -0.8	-	V	
V_{ILC}	CLK Logical Zero Input Voltage	-	0.8	V	
V_{OH}	Output High Voltage	3.0 V_{CC} -0.4	-	V V	$IOH = -2.5mA$ $IOH = -100\mu A$
V_{OL}	Output Low Voltage	-	0.4	V	$IOL = +2.5mA$
I_I	Input Leakage Current	-1.0	1.0	μA	$V_{IN} = 0V$ or V_{CC} Pins 17-19, 21-23, 33
IBHH	Input Current-Bus Hold High	-40	-400	μA	$V_{IN} = -3.0V$ (Note 1)
IBHL	Input Current-Bus Hold Low	40	400	μA	$V_{IN} = -0.8V$ (Note 2)
I_O	Output Leakage Current	-	-10.0	μA	$V_{OUT} = 0V$ (Note 5)
ICCSB	Standby Power Supply Current	-	500	μA	$V_{CC} = 5.5V$ (Note 3)
ICCOP	Operating Power Supply Current	-	10	mA/MHz	FREQ = Max, $V_{IN} = V_{CC}$ or GND, Outputs Open

NOTES:

1. IBHH should be measured after raising V_{IN} to V_{CC} and then lowering to 3.0V on the following pins 2-16, 26-32, 34-39.
2. IBHL should be measured after lowering V_{IN} to GND and then raising to 0.8V on the following pins: 2-16, 35-39.
3. ICCSB tested during clock high time after HALT instruction executed. $V_{IN} = V_{CC}$ or GND, $V_{CC} = 5.5V$, Outputs unloaded.
4. MN/MX is a strap option and should be held to V_{CC} or GND.
5. IO should be measured by putting the pin in a high impedance state and then driving V_{OUT} to GND on the following pins: 26-29 and 32.

Capacitance $T_A = 25^{\circ}C$

SYMBOL	PARAMETER	TYPICAL	UNITS	TEST CONDITIONS
CIN	Input Capacitance	25	pF	FREQ = 1MHz. All measurements are referenced to device GND
COUT	Output Capacitance	25	pF	FREQ = 1MHz. All measurements are referenced to device GND
CI/O	I/O Capacitance	25	pF	FREQ = 1MHz. All measurements are referenced to device GND

80C88

AC Electrical Specifications	$V_{CC} = 5.0V \pm 10\%$; $T_A = 0^{\circ}C$ to $+70^{\circ}C$ (C80C88, C80C88-2)
	$V_{CC} = 5.0V \pm 100\%$; $T_A = -40^{\circ}C$ to $+85^{\circ}C$ (I80C88, I80C88-2)
	$V_{CC} = 5.0V \pm 100\%$; $T_A = -55^{\circ}C$ to $+125^{\circ}C$ (M80C88)
	$V_{CC} = 5.0V \pm 5\%$; $T_A = -55^{\circ}C$ to $+125^{\circ}C$ (M80C88-2)

MINIMUM COMPLEXITY SYSTEM

SYMBOL	PARAMETER	80C88		80C88-2		UNITS	TEST CONDITIONS
		MIN	MAX	MIN	MAX		
TIMING REQUIREMENTS							
(1)	TCLCL	CLK Cycle Period	200	-	125	-	ns
(2)	TCLCH	CLK Low Time	118	-	68	-	ns
(3)	TCHCL	CLK High Time	69	-	44	-	ns
(4)	TCH1CH2	CLK Rise Time	-	10	-	10	ns From 1.0V to 3.5V
(5)	TCL2CL1	CLK Fall Time	-	10	-	10	ns From 3.5V to 1.0V
(6)	TDVCL	Data In Setup Time	30	-	20	-	ns
(7)	TCLDX1	Data In Hold Time	10	-	10	-	ns
(8)	TR1VCL	RDY Setup Time into 82C84A (Notes 6, 7)	35	-	35	-	ns
(9)	TCLR1X	RDY Hold Time into 82C84A (Notes 6, 7)	0	-	0	-	ns
(10)	TRYHCH	READY Setup Time into 80C88	118	-	68	-	ns
(11)	TCHRYX	READY Hold Time into 80C88	30	-	20	-	ns
(12)	TRYLCL	READY Inactive to CLK (Note 8)	-8	-	-8	-	ns
(13)	THVCH	HOLD Setup Time	35	-	20	-	ns
(14)	TINVCH	INTR, NMI, TEST Setup Time (Note 7)	30	-	15	-	ns
(15)	TILIH	Input Rise Time (Except CLK)	-	15	-	15	ns From 0.8V to 2.0V
(16)	TIHIL	Input Fall Time (Except CLK)	-	15	-	15	ns From 2.0V to 0.8V
TIMING RESPONSES							
(17)	TCLAV	Address Valid Delay	10	110	10	60	ns CL = 100pF
(18)	TCLAX	Address Hold Time	10	-	10	-	ns CL = 100pF
(19)	TCLAZ	Address Float Delay	TCLAX	80	TCLAX	50	ns CL = 100pF
(20)	TCHSZ	Status Float Delay	-	80	-	50	ns CL = 100pF
(21)	TCHSV	Status Active Delay	10	110	10	60	ns CL = 100pF
(22)	TLHLL	ALE Width	TCLCH-20	-	TCLCH-10	-	ns CL = 100pF
(23)	TCLLH	ALE Active Delay	-	80	-	50	ns CL = 100pF
(24)	TCHLL	ALE Inactive Delay	-	85	-	55	ns CL = 100pF

80C88

AC Electrical Specifications	$V_{CC} = 5.0V \pm 10\%$; $T_A = 0^{\circ}C$ to $+70^{\circ}C$ (C80C88, C80C88-2)
	$V_{CC} = 5.0V \pm 100\%$; $T_A = -40^{\circ}C$ to $+85^{\circ}C$ (I80C88, I80C88-2)
	$V_{CC} = 5.0V \pm 100\%$; $T_A = -55^{\circ}C$ to $+125^{\circ}C$ (M80C88)
	$V_{CC} = 5.0V \pm 5\%$; $T_A = -55^{\circ}C$ to $+125^{\circ}C$ (M80C88-2) (Continued)

MINIMUM COMPLEXITY SYSTEM

SYMBOL	PARAMETER	80C88		80C88-2		UNITS	TEST CONDITIONS	
		MIN	MAX	MIN	MAX			
(25)	TLLAX	Address Hold Time to ALE Inactive	TCHCL-10	-	TCHCL-10	-	ns	CL = 100pF
(26)	TCLDV	Data Valid Delay	10	110	10	60	ns	CL = 100pF
(27)	TCLDX2	Data Hold Time	10	-	10	-	ns	CL = 100pF
(28)	TWHDX	Data Hold Time After WR	TCLCL-30	-	TCLCL-30	-	ns	CL = 100pF
(29)	TCVCTV	Control Active Delay 1	10	110	10	70	ns	CL = 100pF
(30)	TCHCTV	Control Active Delay 2	10	110	10	60	ns	CL = 100pF
(31)	TCVCTX	Control Inactive Delay	10	110	10	70	ns	CL = 100pF
(32)	TAZRL	Address Float to READ Active	0	-	0	-	ns	CL = 100pF
(33)	TCLRRL	RD Active Delay	10	165	10	100	ns	CL = 100pF
(34)	TCLRH	RD Inactive Delay	10	150	10	80	ns	CL = 100pF
(35)	TRHAV	RD Inactive to Next Address Active	TCLCL-45	-	TCLCL-40	-	ns	CL = 100pF
(36)	TCLHAV	HLDA Valid Delay	10	160	10	100	ns	CL = 100pF
(37)	TRLRH	RD Width	2TCLCL-75	-	2TCLCL-50	-	ns	CL = 100pF
(38)	TWLWH	WR Width	2TCLCL-60	-	2TCLCL-40	-	ns	CL = 100pF
(39)	TAVAL	Address Valid to ALE Low	TCLCH-60	-	TCLCH-40	-	ns	CL = 100pF
(40)	TOLOH	Output Rise Time	-	15	-	15	ns	From 0.8V to 2.0V
(41)	TOHOL	Output Fall Time	-	15	-	15	ns	From 2.0V to 0.8V

NOTES:

6. Signal at 82C84A shown for reference only.
7. Setup requirement for asynchronous signal only to guarantee recognition at next CLK.
8. Applies only to T2 state (8ns into T3).

Waveforms

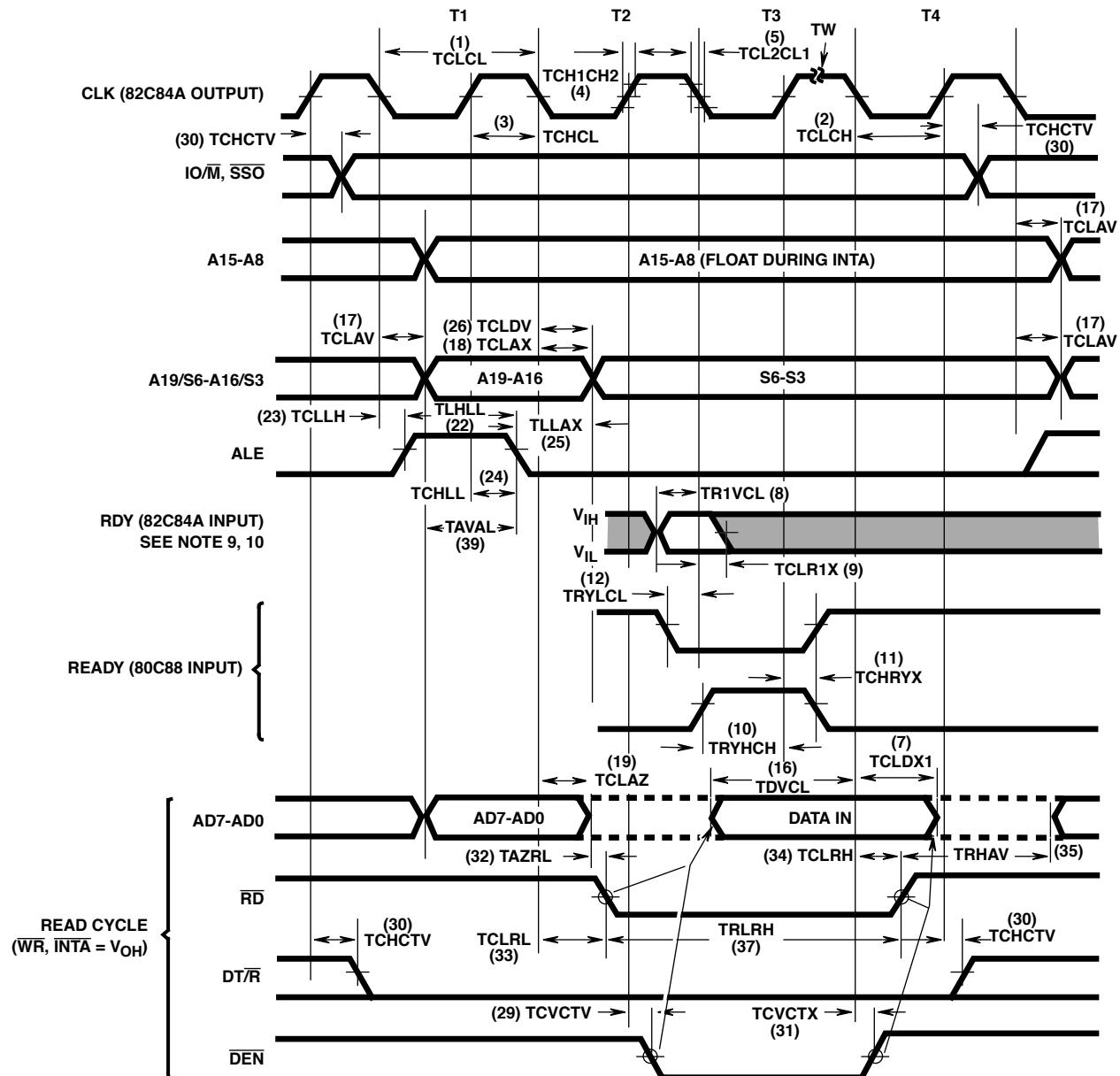
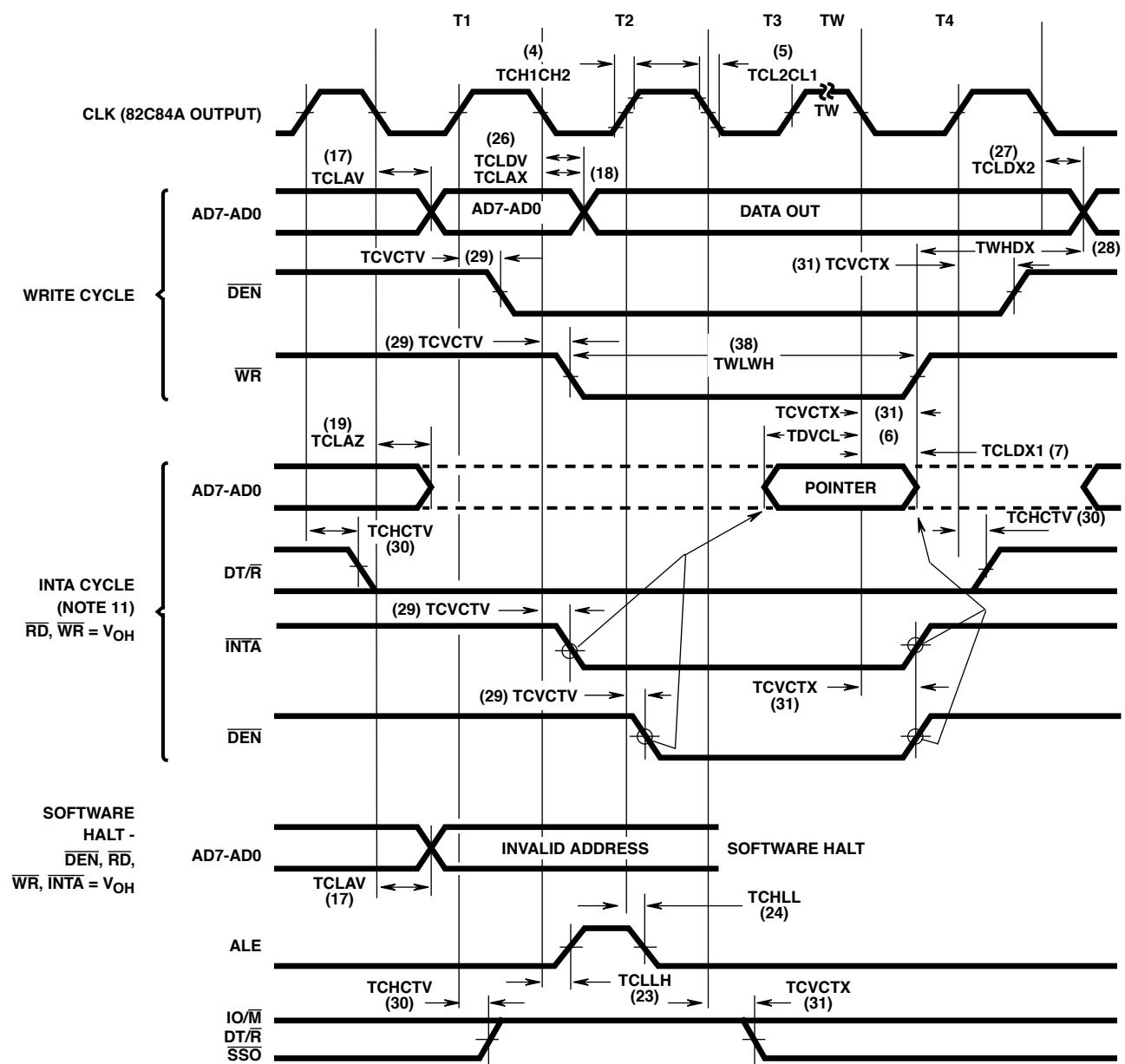


FIGURE 9. BUS TIMING - MINIMUM MODE SYSTEM

NOTES:

9. RDY is sampled near the end of T2, T3, TW to determine if TW machine states are to be inserted.
10. Signals at 82C84A are shown for reference only.

Waveforms (Continued)**FIGURE 10. BUS TIMING - MINIMUM MODE SYSTEM (Continued)****NOTES:**

11. Two INTA cycles run back-to-back. The 80C88 local ADDR/DATA bus is floating during both INTA cycles. Control signals are shown for the second INTA cycle.
12. Signals at 82C84A are shown for reference only.

AC Electrical Specifications

$V_{CC} = 5.0V \pm 10\%$	$T_A = 0^{\circ}C$ to $+70^{\circ}C$	(C80C88, C80C88-2)
$V_{CC} = 5.0V \pm 10\%$	$T_A = -40^{\circ}C$ to $+85^{\circ}C$	(I80C88, I80C88-2)
$V_{CC} = 5.0V \pm 10\%$	$T_A = -55^{\circ}C$ to $+125^{\circ}C$	(M80C88)
$V_{CC} = 5.0V \pm 5\%$	$T_A = -55^{\circ}C$ to $+125^{\circ}C$	(M80C88-2)

MAX MODE SYSTEM (USING 82C88 BUS CONTROLLER)

SYMBOL	PARAMETER	80C88		80C88-2		UNITS	TEST CONDITIONS
		MIN	MAX	MIN	MAX		
TIMING REQUIREMENTS							
(1)	TCLCL	CLK Cycle Period	200	-	125	-	ns
(2)	TCLCH	CLK Low Time	118	-	68	-	ns
(3)	TCHCL	CLK High Time	69	-	44	-	ns
(4)	TCH1CH2	CLK Rise Time	-	10	-	10	ns From 1.0V to 3.5V
(5)	TCL2CL1	CLK Fall Time	-	10	-	10	ns From 3.5V to 1.0V
(6)	TDVCL	Data in Setup Time	30	-	20	-	ns
(7)	TCLDX1	Data In Hold Time	10	-	10	-	ns
(8)	TR1VCL	RDY Setup Time into 82C84 (Notes 13, 14)	35	-	35	-	ns
(9)	TCLR1X	RDY Hold Time into 82C84 (Notes 13, 14)	0	-	0	-	ns
(10)	TRYHCH	READY Setup Time into 80C88	118	-	68	-	ns
(11)	TCHRYX	READY Hold Time into 80C88	30	-	20	-	ns
(12)	TRYLCL	READY Inactive to CLK (Note 15)	-8	-	-8	-	ns
(13)	TINVCH	Setup Time for Recognition (INTR, NMI, \overline{TEST}) (Note 14)	30	-	15	-	ns
(14)	TGVCH	RQ/GT Setup Time	30	-	15	-	ns
(15)	TCHGX	RQ Hold Time into 80C88 (Note 16)	40	TCHCL+ 10	30	TCHCL+ 10	ns
(16)	TILIH	Input Rise Time (Except CLK)	-	15	-	15	ns From 0.8V to 2.0V
(17)	TIHIL	Input Fall Time (Except CLK)	-	15	-	15	ns From 2.0V to 0.8V
TIMING RESPONSES							
(18)	TCLML	Command Active Delay (Note 13)	5	35	5	35	ns
(19)	TCLMH	Command Inactive (Note 13)	5	35	5	35	ns
(20)	TRYHSH	READY Active to Status Passive (Notes 15, 17)	-	110	-	65	ns
(21)	TCHSV	Status Active Delay	10	110	10	60	ns
(22)	TCLSH	Status Inactive Delay (Note 17)	10	130	10	70	ns
(23)	TCLAV	Address Valid Delay	10	110	10	60	ns
(24)	TCLAX	Address Hold Time	10	-	10	-	ns
(25)	TCLAZ	Address Float Delay	TCLAX	80	TCLAX	50	ns
(26)	TCHSZ	Status Float Delay	-	80	-	50	ns
(27)	TSVLH	Status Valid to ALE High (Note 13)	-	20	-	20	ns
(28)	TSVMCH	Status Valid to MCE High (Note 13)	-	30	-	30	ns
(29)	TCLLH	CLK Low to ALE Valid (Note 13)	-	20	-	20	ns
(30)	TCLMCH	CLK Low to MCE High (Note 13)	-	25	-	25	ns
(31)	TCHLL	ALE Inactive Delay (Note 13)	4	18	4	18	ns

CL = 100pF
for all 80C88 outputs
in addition to internal loads.

80C88

AC Electrical Specifications

$V_{CC} = 5.0V \pm 10\%$	$T_A = 0^{\circ}C$ to $+70^{\circ}C$	(C80C88, C80C88-2)
$V_{CC} = 5.0V \pm 10\%$	$T_A = -40^{\circ}C$ to $+85^{\circ}C$	(I80C88, I80C88-2)
$V_{CC} = 5.0V \pm 10\%$	$T_A = -55^{\circ}C$ to $+125^{\circ}C$	(M80C88)
$V_{CC} = 5.0V \pm 5\%$	$T_A = -55^{\circ}C$ to $+125^{\circ}C$	(M80C88-2) (Continued)

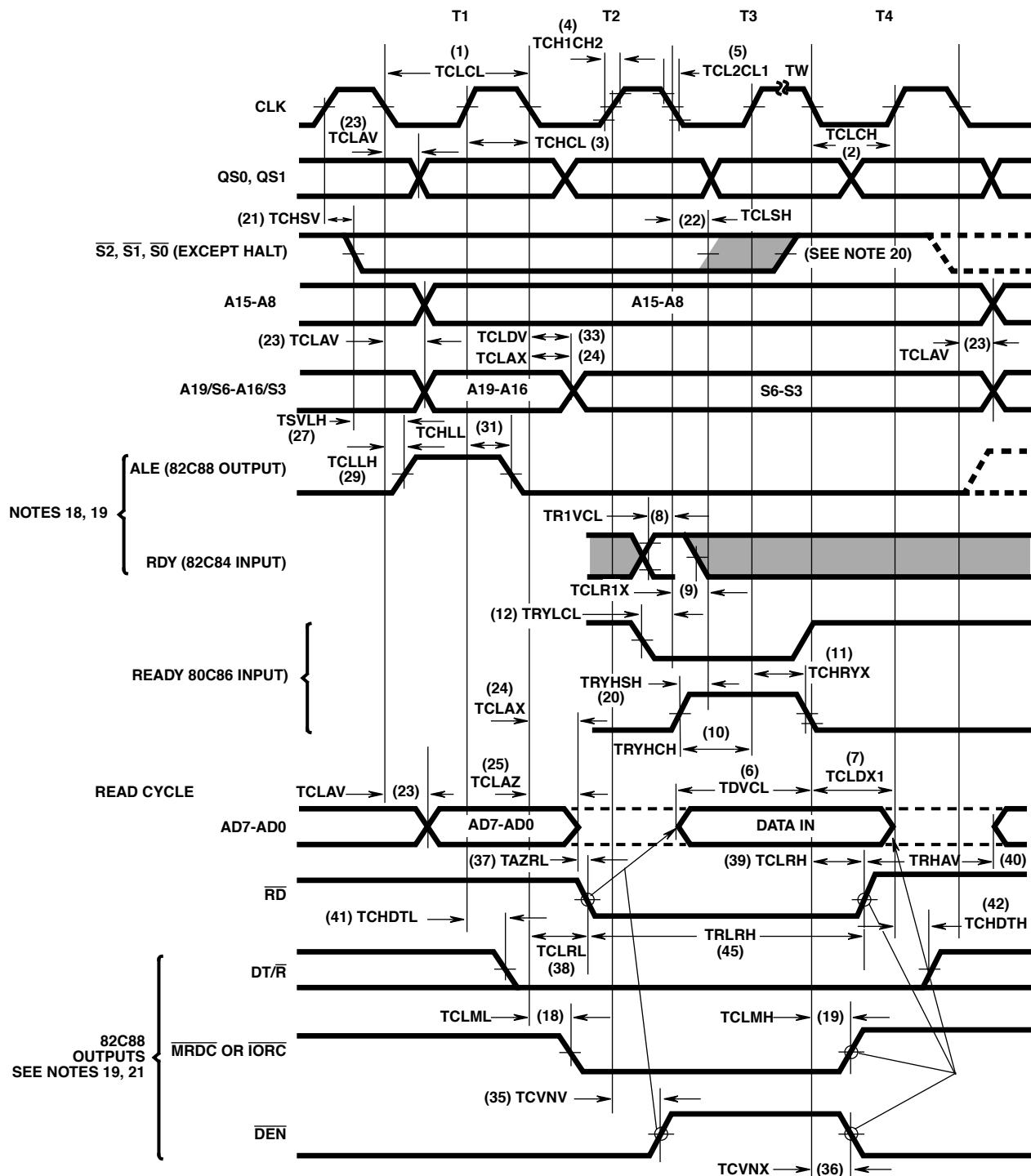
MAX MODE SYSTEM (USING 82C88 BUS CONTROLLER)

SYMBOL	PARAMETER	80C88		80C88-2		UNITS	TEST CONDITIONS
		MIN	MAX	MIN	MAX		
(32)	TCLMCL	MCE Inactive Delay (Note 13)	-	15	-	15	ns
(33)	TCLDV	Data Valid Delay	10	110	10	60	ns
(34)	TCLDX2	Data Hold Time	10	-	10	-	ns
(35)	TCVN	Control Active Delay (Note 13)	5	45	5	45	ns
(36)	TCVNX	Control Inactive Delay (Note 13)	10	45	10	45	ns
(37)	TAZRL	Address Float to Read Active	0	-	0	-	ns
(38)	TCLRL	\bar{RD} Active Delay	10	165	10	100	ns
(39)	TCLRH	\bar{RD} Inactive Delay	10	150	10	80	ns
(40)	TRHAV	\bar{RD} Inactive to Next Address Active	TCLCL -45	-	TCLCL -40	-	ns
(41)	TCHDTL	Direction Control Active Delay (Note 13)	-	50	-	50	ns
(42)	TCHDTH	Direction Control Inactive Delay (Note 1)	-	30	-	30	ns
(43)	TCLGL	\bar{GT} Active Delay	0	85	0	50	ns
(44)	TCLGH	\bar{GT} Inactive Delay	0	85	0	50	ns
(45)	TRLRH	RD Width	2TCLC L -75	-	2TCLC L -50	-	ns
(46)	TOLOH	Output Rise Time	-	15	-	15	ns
(47)	TOHOL	Output Fall Time	-	15	-	15	ns

CL = 100pF
for all 80C88 outputs
in addition to internal loads.

NOTES:

13. Signal at 82C84A or 82C88 shown for reference only.
14. Setup requirement for asynchronous signal only to guarantee recognition at next CLK.
15. Applies only to T2 state (8ns into T3).
16. The 80C88 actively pulls the \bar{RQ}/\bar{GT} pin to a logic one on the following clock low time.
17. Status lines return to their inactive (logic one) state after CLK goes low and READY goes high.

Waveforms**FIGURE 11. BUS TIMING - MAXIMUM MODE (USING 82C88)**

NOTES:

18. RDY is sampled near the end of T2, T3, TW to determine if TW machine states are to be inserted.
19. Signals at 82C84A or 82C88 are shown for reference only.
20. Status inactive in state just prior to T4.
21. The issuance of the 82C88 command and control signals (MRDC, MWTC, AMWC, IORC, IOWC, AIOWC, INTA, and DEN) lags the active high 82C88 CEN.

Waveforms (Continued)

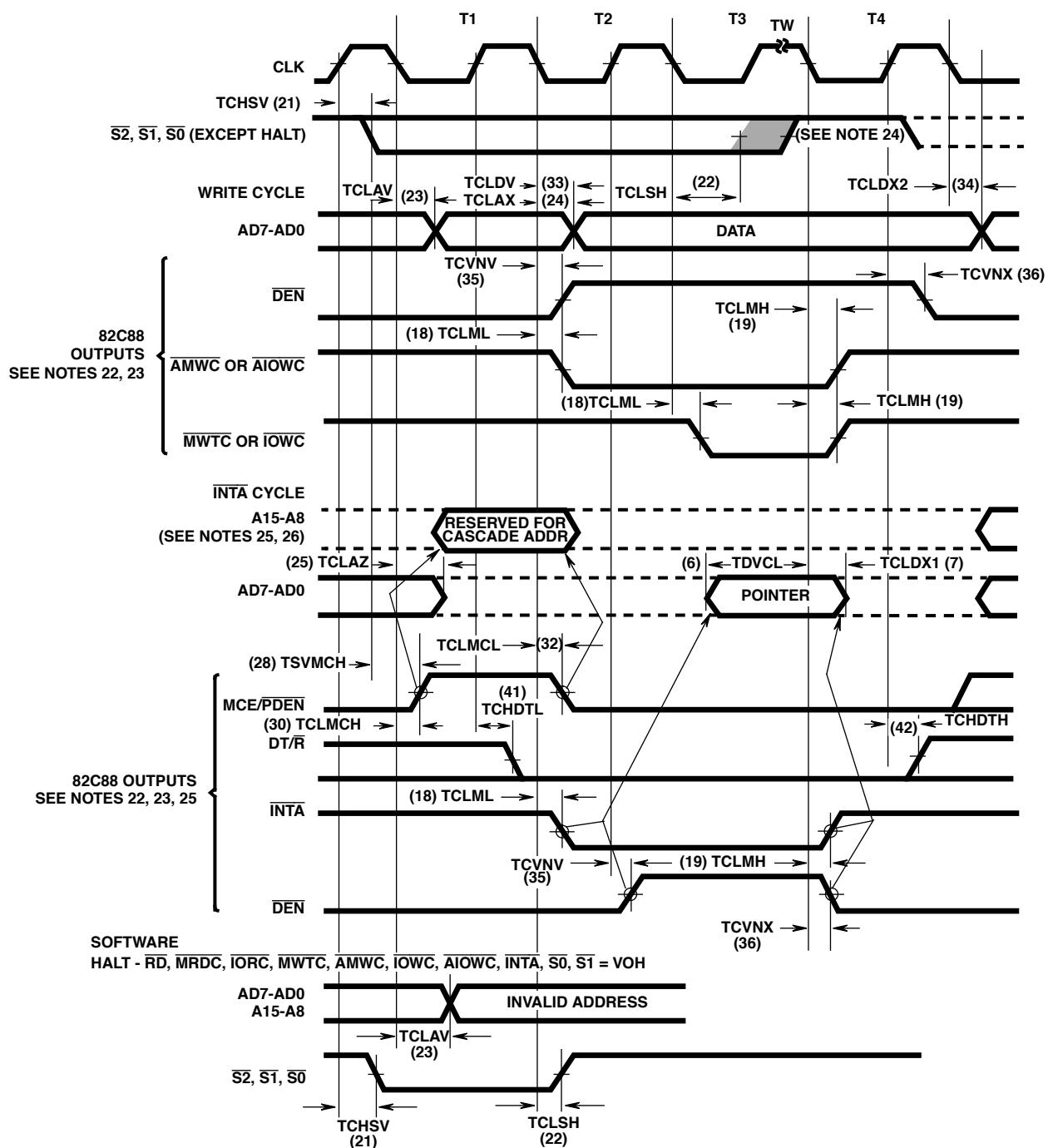
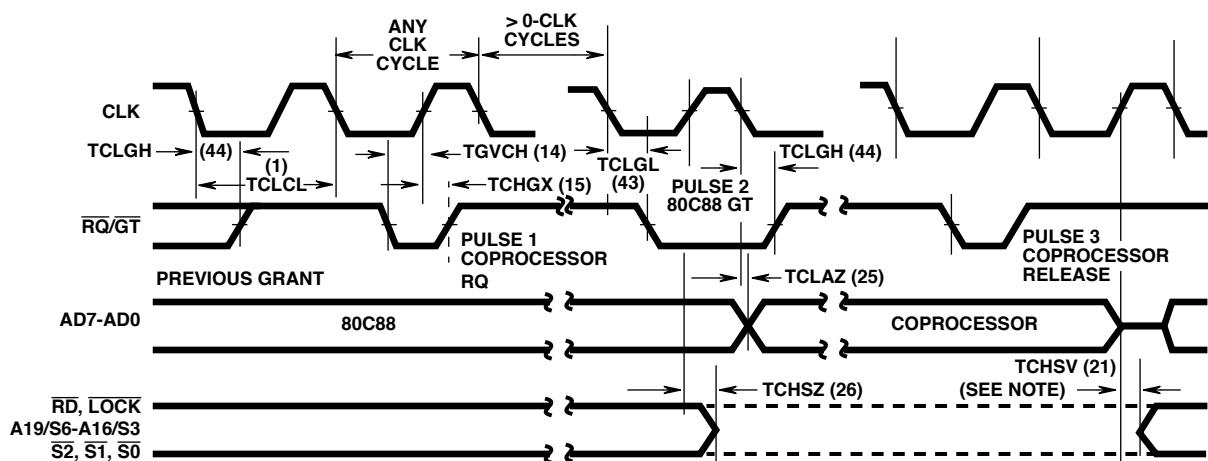


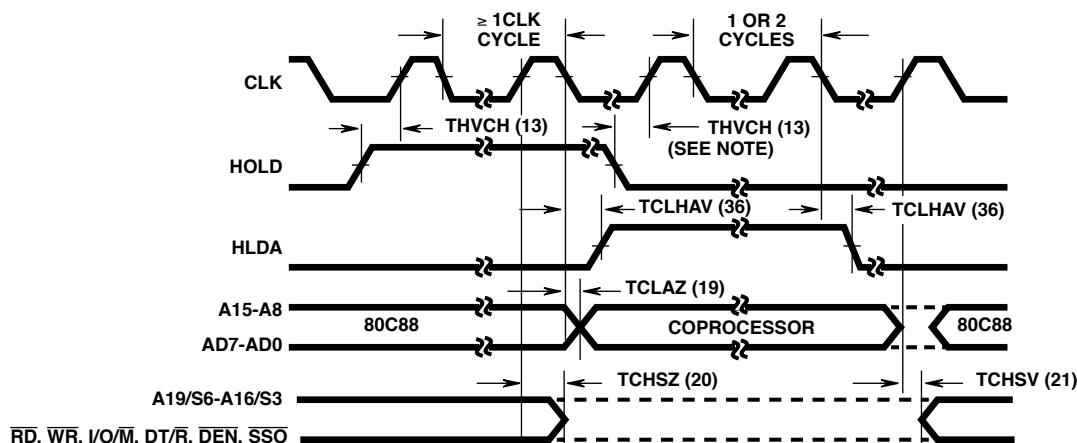
FIGURE 12. BUS TIMING - MAXIMUM MODE SYSTEM (USING 82C88) (Continued)

NOTES:

22. Signals at 82C84A or 82C86 are shown for reference only.
23. The issuance of the 82C88 command and control signals (MRDC, MWTC, AMWC, IORC, IOWC, AIOWC, INTA and DEN) lags the active high 82C88 CEN.
24. Status inactive in state just prior to T4.
25. Cascade address is valid between first and second INTA cycles.
26. Two INTA cycles run back-to-back. The 80C88 local ADDR/DATA bus is floating during both INTA cycles. Control for pointer address is shown for second INTA cycle.

Waveforms (Continued)**FIGURE 13. REQUEST/GRANT SEQUENCE TIMING (MAXIMUM MODE ONLY)**

NOTE: The coprocessor may not drive the busses outside the region shown without risking contention.

**FIGURE 14. HOLD/HOLD ACKNOWLEDGE TIMING (MINIMUM MODE ONLY)**

NOTE: Setup requirements for asynchronous signals only to guarantee recognition at next CLK.

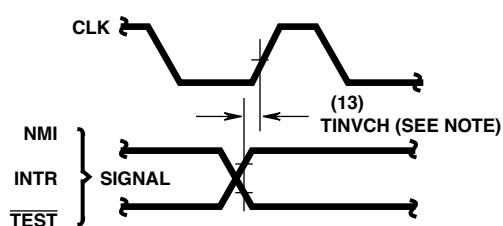


FIGURE 15. ASYNCHRONOUS SIGNAL RECOGNITION
NOTE: Setup requirements for asynchronous signals only to guarantee recognition at next CLK.

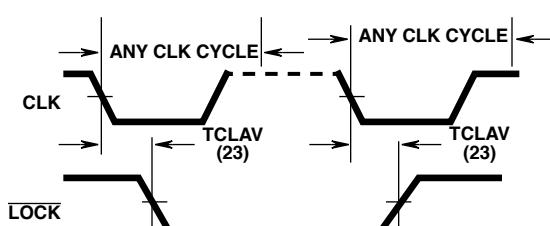


FIGURE 16. BUS LOCK SIGNAL TIMING (MAXIMUM MODE ONLY)

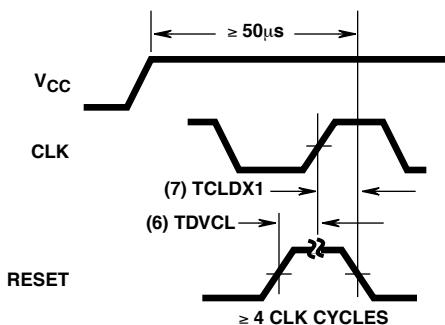
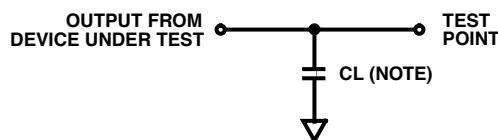
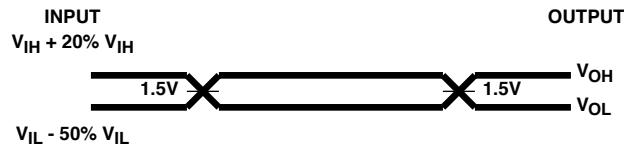
Waveforms (Continued)

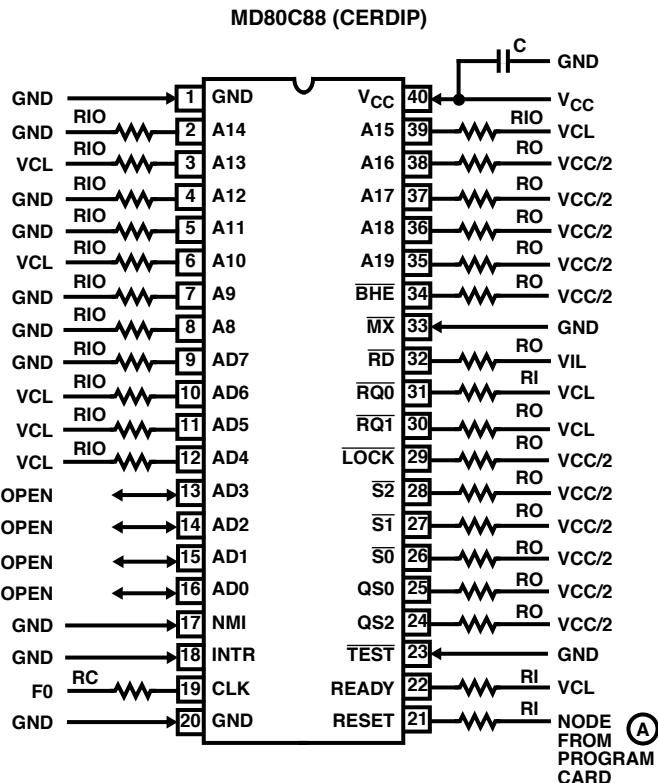
FIGURE 17. RESET TIMING

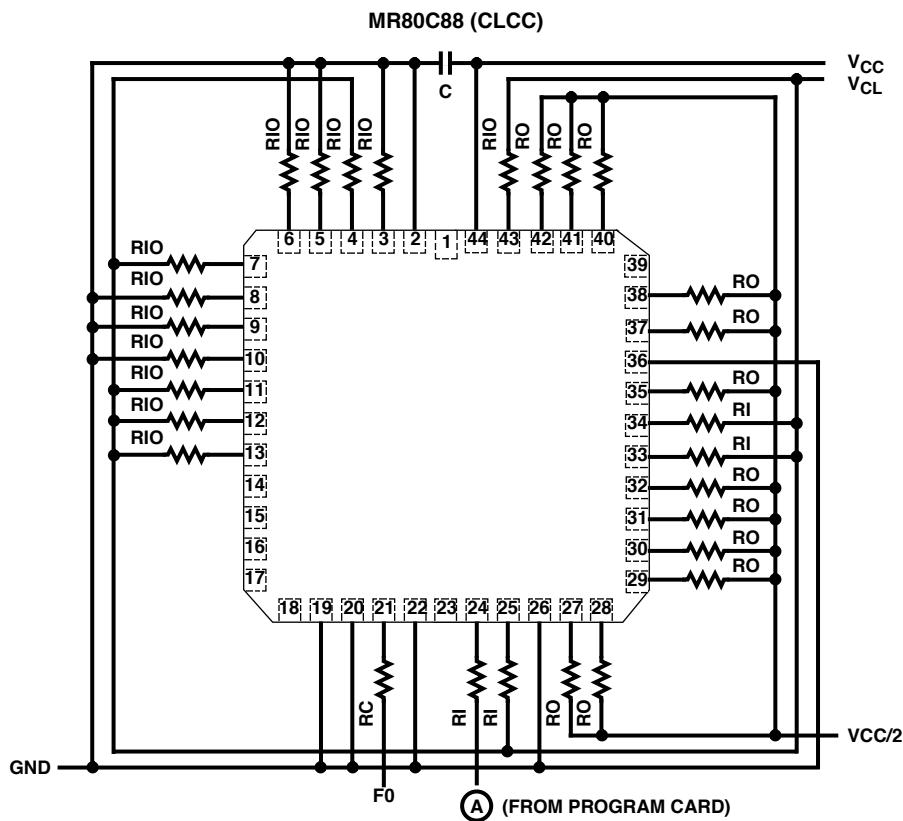
AC Test Circuit

NOTE: Includes stay and jig capacitance.

AC Testing Input, Output Waveform

AC Testing: All input signals (other than CLK) must switch between $V_{ILMAX} - 50\% V_{IL}$ and $V_{IHMIM} + 20\% V_{IH}$. CLK must switch between 0.4V and $V_{CC} - 0.4V$. Input rise and fall times are driven at 1ns/V.

Burn-In Circuits

Burn-In Circuits (Continued)

NOTES:

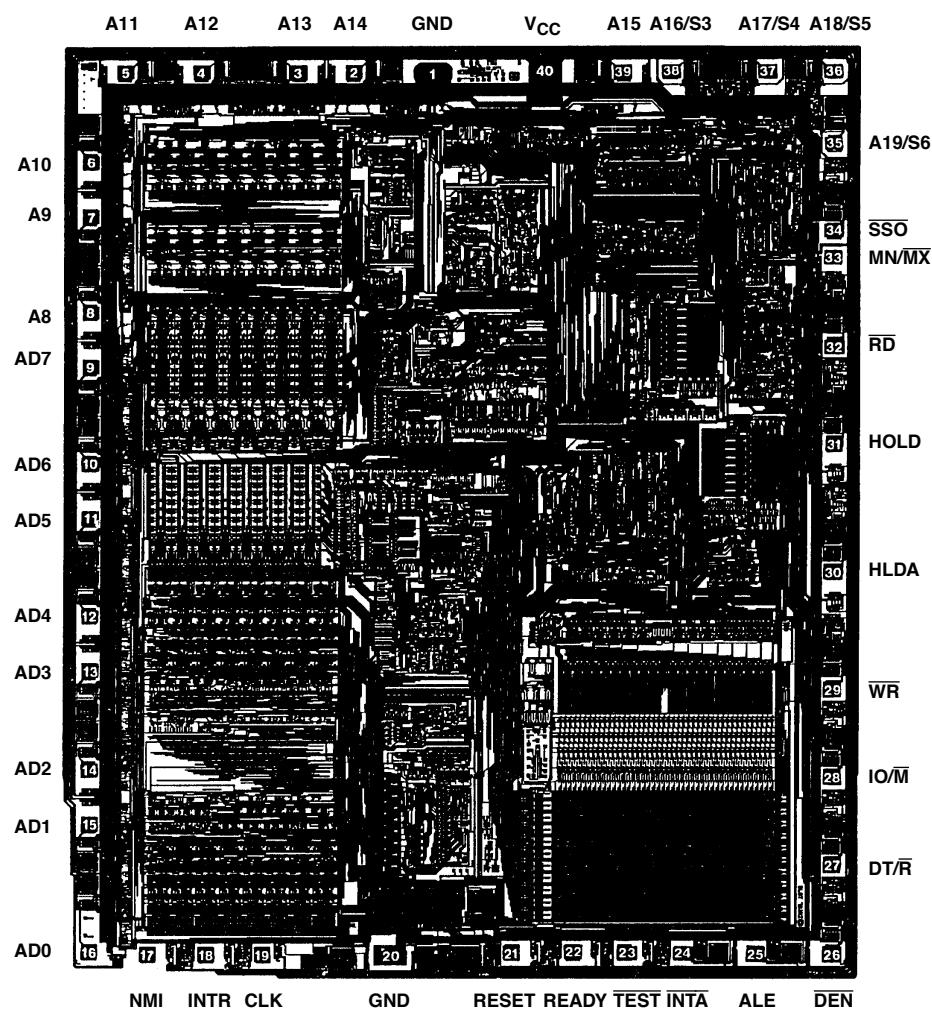
1. $V_{CC} = 5.5V \pm 0.5V$, $GND = 0V$.
2. Input voltage limits (except clock):
 - V_{IL} (Maximum) = 0.4V
 - V_{IH} (Minimum) = 2.6V, V_{IH} (Clock) = $V_{CC} - 0.4V$ minimum.
3. $V_{CC}/2$ is external supply set to $2.7V \pm 10\%$.
4. V_{CL} is generated on program card ($V_{CC} - 0.65V$).
5. Pins 13 - 16 input sequenced instructions from internal hold devices, (DIP Only).
6. $F_0 = 100kHz \pm 10\%$.
7. Node A = a $40\mu s$ pulse every 2.56ms.

COMPONENTS:

1. $RI = 10k\Omega \pm 5\%$, 1/4W
2. $RO = 1.2k\Omega \pm 5\%$, 1/4W
3. $RIO = 2.7k\Omega \pm 5\%$, 1/4W
4. $RC = 1k\Omega \pm 5\%$, 1/4W
5. $C = 0.01\mu F$ (Minimum)

Die Characteristics**DIE DIMENSIONS:**249.2 x 290.9 x 19 \pm 1 mils**METALLIZATION:**Type: Silicon - Aluminum
Thickness: 11kÅ \pm 2kÅ**GLASSIVATION:**Type: SiO₂
Thickness: 8kÅ \pm 1kÅ**WORST CASE CURRENT DENSITY:**1.5 x 10⁵ A/cm²**Metallization Mask Layout**

80C88



Instruction Set Summary

Mnemonic and Description	Instruction Code			
	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0
DATA TRANSFER				
MOV = MOVE:				
Register/Memory to/from Register	1 0 0 0 1 0 d w	mod reg r/m		
Immediate to Register/Memory	1 1 0 0 0 1 1 w	mod 0 0 0 r/m	data	data if w = 1
Immediate to Register	1 0 1 1 w reg	data	data if w = 1	
Memory to Accumulator	1 0 1 0 0 0 0 w	addr-low	addr-high	
Accumulator to Memory	1 0 1 0 0 0 1 w	addr-low	addr-high	
Register/Memory to Segment Register ††	1 0 0 0 1 1 1 0	mod 0 reg r/m		
Segment Register to Register/Memory	1 0 0 0 1 1 0 0	mod 0 reg r/m		
PUSH = Push:				
Register/Memory	1 1 1 1 1 1 1 1	mod 1 1 0 r/m		
Register	0 1 0 1 0 reg			
Segment Register	0 0 0 reg 1 1 0			
POP = Pop:				
Register/Memory	1 0 0 0 1 1 1 1	mod 0 0 0 r/m		
Register	0 1 0 1 1 reg			
Segment Register	0 0 0 reg 1 1 1			
XCHG = Exchange:				
Register/Memory with Register	1 0 0 0 0 1 1 w	mod reg r/m		
Register with Accumulator	1 0 0 1 0 reg			
IN = Input from:				
Fixed Port	1 1 1 0 0 1 0 w	port		
Variable Port	1 1 1 0 1 1 0 w			
OUT = Output to:				
Fixed Port	1 1 1 0 0 1 1 w	port		
Variable Port	1 1 1 0 1 1 1 w			
XLAT = Translate Byte to AL	1 1 0 1 0 1 1 1			
LEA = Load EA to Register2	1 0 0 0 1 1 0 1	mod reg r/m		
LDS = Load Pointer to DS	1 1 0 0 0 1 0 1	mod reg r/m		
LES = Load Pointer to ES	1 1 0 0 0 1 0 0	mod reg r/m		
LAHF = Load AH with Flags	1 0 0 1 1 1 1 1			
SAHF = Store AH into Flags	1 0 0 1 1 1 1 0			
PUSHF = Push Flags	1 0 0 1 1 1 0 0			
POPF = Pop Flags	1 0 0 1 1 1 0 1			
ARITHMETIC				
ADD = Add:				
Register/Memory with Register to Either	0 0 0 0 0 0 d w	mod reg r/m		
Immediate to Register/Memory	1 0 0 0 0 0 s w	mod 0 0 0 r/m	data	data if s:w = 01
Immediate to Accumulator	0 0 0 0 0 1 0 w	data	data if w = 1	
ADC = Add with Carry:				
Register/Memory with Register to Either	0 0 0 1 0 0 d w	mod reg r/m		

Instruction Set Summary (Continued)

Mnemonic and Description	Instruction Code			
	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0
Immediate to Register/Memory	1 0 0 0 0 0 s w	mod 0 1 0 r/m	data	data if s:w = 01
Immediate to Accumulator	0 0 0 1 0 1 0 w	data	data if w = 1	
INC = Increment:				
Register/Memory	1 1 1 1 1 1 1 w	mod 0 0 0 r/m		
Register	0 1 0 0 0 reg			
AAA = ASCII Adjust for Add	0 0 1 1 0 1 1 1			
DAA = Decimal Adjust for Add	0 0 1 0 0 1 1 1			
SUB = Subtract:				
Register/Memory and Register to Either	0 0 1 0 1 0 d w	mod reg r/m		
Immediate from Register/Memory	1 0 0 0 0 0 s w	mod 1 0 1 r/m	data	data if s:w = 01
Immediate from Accumulator	0 0 1 0 1 1 0 w	data	data if w = 1	
SBB = Subtract with Borrow				
Register/Memory and Register to Either	0 0 0 1 1 0 d w	mod reg r/m		
Immediate from Register/Memory	1 0 0 0 0 0 s w	mod 0 1 1 r/m	data	data if s:w = 01
Immediate from Accumulator	0 0 0 1 1 1 0 w	data	data if w = 1	
DEC = Decrement:				
Register/Memory	1 1 1 1 1 1 1 w	mod 0 0 1 r/m		
Register	0 1 0 0 1 reg			
NEG = Change Sign	1 1 1 1 0 1 1 w	mod 0 1 1 r/m		
CMP = Compare:				
Register/Memory and Register	0 0 1 1 1 0 d w	mod reg r/m		
Immediate with Register/Memory	1 0 0 0 0 0 s w	mod 1 1 1 r/m	data	data if s:w = 01
Immediate with Accumulator	0 0 1 1 1 1 0 w	data	data if w = 1	
AAS = ASCII Adjust for Subtract	0 0 1 1 1 1 1 1			
DAS = Decimal Adjust for Subtract	0 0 1 0 1 1 1 1			
MUL = Multiply (Unsigned)	1 1 1 1 0 1 1 w	mod 1 0 0 r/m		
IMUL = Integer Multiply (Signed)	1 1 1 1 0 1 1 w	mod 1 0 1 r/m		
AAM = ASCII Adjust for Multiply	1 1 0 1 0 1 0 0	0 0 0 0 1 0 1 0		
DIV = Divide (Unsigned)	1 1 1 1 0 1 1 w	mod 1 1 0 r/m		
IDIV = Integer Divide (Signed)	1 1 1 1 0 1 1 w	mod 1 1 1 r/m		
AAD = ASCII Adjust for Divide	1 1 0 1 0 1 0 1	0 0 0 0 1 0 1 0		
CBW = Convert Byte to Word	1 0 0 1 1 0 0 0			
CWD = Convert Word to Double Word	1 0 0 1 1 0 0 1			
LOGIC				
NOT = Invert	1 1 1 1 0 1 1 w	mod 0 1 0 r/m		
SHL/SAL = Shift Logical/Arithmetic Left	1 1 0 1 0 0 v w	mod 1 0 0 r/m		
SHR = Shift Logical Right	1 1 0 1 0 0 v w	mod 1 0 1 r/m		
SAR = Shift Arithmetic Right	1 1 0 1 0 0 v w	mod 1 1 1 r/m		
ROL = Rotate Left	1 1 0 1 0 0 v w	mod 0 0 0 r/m		
ROR = Rotate Right	1 1 0 1 0 0 v w	mod 0 0 1 r/m		
RCL = Rotate Through Carry Flag Left	1 1 0 1 0 0 v w	mod 0 1 0 r/m		

Instruction Set Summary (Continued)

Mnemonic and Description	INSTRUCTION CODE								
	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0					
RCR = Rotate Through Carry Right	1 1 0 1 0 0 v w	mod 0 1 1 r/m							
AND = And:									
Reg./Memory and Register to Either	0 0 1 0 0 0 0 d w	mod reg r/m							
Immediate to Register/Memory	1 0 0 0 0 0 0 w	mod 1 0 0 r/m	data						
Immediate to Accumulator	0 0 1 0 0 1 0 w	data	data if w = 1						
TEST = And Function to Flags, No Result:									
Register/Memory and Register	1 0 0 0 0 1 0 w	mod reg r/m							
Immediate Data and Register/Memory	1 1 1 1 0 1 1 w	mod 0 0 0 r/m	data	data if w = 1					
Immediate Data and Accumulator	1 0 1 0 1 0 0 w	data	data if w = 1						
OR = Or:									
Register/Memory and Register to Either	0 0 0 0 1 0 d w	mod reg r/m							
Immediate to Register/Memory	1 0 0 0 0 0 0 w	mod 1 0 1 r/m	data	data if w = 1					
Immediate to Accumulator	0 0 0 0 1 1 0 w	data	data if w = 1						
XOR = Exclusive or:									
Register/Memory and Register to Either	0 0 1 1 0 0 d w	mod reg r/m							
Immediate to Register/Memory	1 0 0 0 0 0 0 w	mod 1 1 0 r/m	data	data if w = 1					
Immediate to Accumulator	0 0 1 1 0 1 0 w	data	data if w = 1						
STRING MANIPULATION									
REP = Repeat	1 1 1 1 0 0 1 z								
MOVS = Move Byte/Word	1 0 1 0 0 1 0 w								
CMPS = Compare Byte/Word	1 0 1 0 0 1 1 w								
SCAS = Scan Byte/Word	1 0 1 0 1 1 1 w								
LODS = Load Byte/Word to AL/AX	1 0 1 0 1 1 0 w								
STOS = Stor Byte/Word from AL/A	1 0 1 0 1 0 1 w								
CONTROL TRANSFER									
CALL = Call:									
Direct Within Segment	1 1 1 0 1 0 0 0	disp-low	disp-high						
Indirect Within Segment	1 1 1 1 1 1 1 1	mod 0 1 0 r/m							
Direct Intersegment	1 0 0 1 1 0 1 0	offset-low	offset-high						
		seg-low	seg-high						
Indirect Intersegment	1 1 1 1 1 1 1 1	mod 0 1 1 r/m							
JMP = Unconditional Jump:									
Direct Within Segment	1 1 1 0 1 0 0 1	disp-low	disp-high						
Direct Within Segment-Short	1 1 1 0 1 0 1 1	disp							
Indirect Within Segment	1 1 1 1 1 1 1 1	mod 1 0 0 r/m							
Direct Intersegment	1 1 1 0 1 0 1 0	offset-low	offset-high						
		seg-low	seg-high						
Indirect Intersegment	1 1 1 1 1 1 1 1	mod 1 0 1 r/m							
RET = Return from CALL:									
Within Segment	1 1 0 0 0 0 1 1								
Within Seg Adding Immed to SP	1 1 0 0 0 0 1 0	data-low	data-high						

Instruction Set Summary (Continued)

Mnemonic and Description	Instruction Code			
	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0
Intersegment	1 1 0 0 1 0 1 1			
Intersegment Adding Immediate to SP	1 1 0 0 1 0 1 0	data-low	data-high	
JE/JZ = Jump on Equal/Zero	0 1 1 1 0 1 0 0	disp		
JL/JNGE = Jump on Less/Not Greater or Equal	0 1 1 1 1 1 0 0	disp		
JLE/JNG = Jump on Less or Equal/ Not Greater	0 1 1 1 1 1 1 0	disp		
JB/JNAE = Jump on Below/Not Above or Equal	0 1 1 1 0 0 1 0	disp		
JBE/JNA = Jump on Below or Equal/Not Above	0 1 1 1 0 1 1 0	disp		
JP/JPE = Jump on Parity/Parity Even	0 1 1 1 1 0 1 0	disp		
JO = Jump on Overflow	0 1 1 1 0 0 0 0	disp		
JS = Jump on Sign	0 1 1 1 1 0 0 0	disp		
JNE/JNZ = Jump on Not Equal/Not Zero	0 1 1 1 0 1 0 1	disp		
JNL/JGE = Jump on Not Less/Greater or Equal	0 1 1 1 1 1 0 1	disp		
JNLE/JG = Jump on Not Less or Equal/Greater	0 1 1 1 1 1 1 1	disp		
JNB/JAE = Jump on Not Below/Above or Equal	0 1 1 1 0 0 1 1	disp		
JNBE/JA = Jump on Not Below or Equal/Above	0 1 1 1 0 1 1 1	disp		
JNP/JPO = Jump on Not Par/Par Odd	0 1 1 1 1 0 1 1	disp		
JNO = Jump on Not Overflow	0 1 1 1 0 0 0 1	disp		
JNS = Jump on Not Sign	0 1 1 1 1 0 0 1	disp		
LOOP = Loop CX Times	1 1 1 0 0 0 1 0	disp		
LOOPZ/LOOPE = Loop While Zero/Equal	1 1 1 0 0 0 0 1	disp		
LOOPNZ/LOOPNE = Loop While Not Zero/Equal	1 1 1 0 0 0 0 0	disp		
JCXZ = Jump on CX Zero	1 1 1 0 0 0 1 1	disp		
INT = Interrupt				
Type Specified	1 1 0 0 1 1 0 1	type		
Type 3	1 1 0 0 1 1 0 0			
INTO = Interrupt on Overflow	1 1 0 0 1 1 1 0			
IRET = Interrupt Return	1 1 0 0 1 1 1 1			
PROCESSOR CONTROL				
CLC = Clear Carry	1 1 1 1 1 0 0 0			
CMC = Complement Carry	1 1 1 1 0 1 0 1			
STC = Set Carry	1 1 1 1 1 0 0 1			
CLD = Clear Direction	1 1 1 1 1 1 0 0			
STD = Set Direction	1 1 1 1 1 1 0 1			
CLI = Clear Interrupt	1 1 1 1 1 0 1 0			
ST = Set Interrupt	1 1 1 1 1 0 1 1			
HLT = Halt	1 1 1 1 0 1 0 0			
WAIT = Wait	1 0 0 1 1 0 1 1			
ESC = Escape (to External Device)	1 1 0 1 1 x x x	mod x x x r/m		
LOCK = Bus Lock Prefix	1 1 1 1 0 0 0 0			

Instruction Set Summary (Continued)

Mnemonic and Description	INSTRUCTION CODE			
	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0

NOTES:

AL = 8-bit accumulator

AX = 16-bit accumulator

CX = Count register

DS= Data segment

ES = Extra segment

Above/below refers to unsigned value.

Greater = more positive;

Less = less positive (more negative) signed values

if d = 1 then "to" reg; if d = 0 then "from" reg

if w = 1 then word instruction; if w = 0 then byte instruction

if mod = 11 then r/m is treated as a REG field

if mod = 00 then DISP = 0†, disp-low and disp-high are absent

if mod = 01 then DISP = disp-low sign-extended

16-bits, disp-high is absent

if mod = 10 then DISP = disp-high:disp-low

if r/m = 000 then EA = (BX) + (SI) + DISP

if r/m = 001 then EA = (BX) + (DI) + DISP

if r/m = 010 then EA = (BP) + (SI) + DISP

if r/m = 011 then EA = (BP) + (DI) + DISP

if r/m = 100 then EA = (SI) + DISP

if r/m = 101 then EA = (DI) + DISP

if r/m = 110 then EA = (BP) + DISP †

if r/m = 111 then EA = (BX) + DISP

DISP follows 2nd byte of instruction (before data if required)

† except if mod = 00 and r/m = 110 then

EA = disp-high: disp-low.

†† MOV CS, REG/MEMORY not allowed.

if s:w = 01 then 16-bits of immediate data form the operand.

if s:w = 11 then an immediate data byte is sign extended to form the 16-bit operand.

if v = 0 then "count" = 1; if v = 1 then "count" in (CL)

x = don't care

z is used for string primitives for comparison with ZF FLAG.

SEGMENT OVERRIDE PREFIX

001 reg 11 0

REG is assigned according to the following table:

16-BIT (w = 1)	8-BIT (w = 0)	SEGMENT
000 AX	000 AL	00 ES
001 CX	001 CL	01 CS
010 DX	010 DL	10 SS
011 BX	011 BL	11 DS
100 SP	100 AH	
101 BP	101 CH	
110 SI	110 DH	
111 DI	111 BH	

Instructions which reference the flag register file as a 16-bit object use the symbol FLAGS to represent the file:

FLAGS =

X:X:X:(OF):(DF):(IF):(TF):(SF):(ZF):X:(AF):X:(PF):X:(CF)

Mnemonics © Intel, 1978

Data Sheet(intel)



8088 8-BIT HMOS MICROPROCESSOR 8088/8088-2

- 8-Bit Data Bus Interface
- 16-Bit Internal Architecture
- Direct Addressing Capability to 1 Mbyte of Memory
- Direct Software Compatibility with 8086 CPU
- 14-Word by 16-Bit Register Set with Symmetrical Operations
- 24 Operand Addressing Modes
- Byte, Word, and Block Operations
- 8-Bit and 16-Bit Signed and Unsigned Arithmetic in Binary or Decimal, Including Multiply and Divide
- Two Clock Rates:
 - 5 MHz for 8088
 - 8 MHz for 8088-2
- Available in EXPRESS
 - Standard Temperature Range
 - Extended Temperature Range

The Intel 8088 is a high performance microprocessor implemented in N-channel, depletion load, silicon gate technology (HMOS-II), and packaged in a 40-pin CERDIP package. The processor has attributes of both 8- and 16-bit microprocessors. It is directly compatible with 8086 software and 8080/8085 hardware and peripherals.

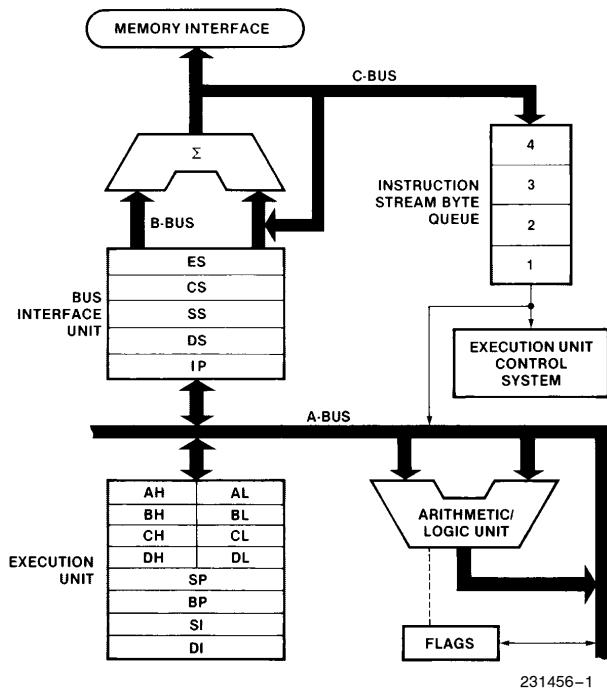


Figure 1. 8088 CPU Functional Block Diagram

	MIN MODE	MAX MODE
GND	1	40 Vcc
A14	2	39 A15
A13	3	38 A16/S3
A12	4	37 A17/S4
A11	5	36 A18/S5
A10	6	35 A19/S6
A9	7	34 SS0 (HIGH)
A8	8	33 MN/MX
AD7	9	32 RD
AD6	10	8088 CPU 31 HOLD (RQ/GT0)
AD5	11	30 HLLDA (RQ/GT1)
AD4	12	29 WR (LOCK)
AD3	13	28 IO/M (S2)
AD2	14	27 DT/R (S1)
AD1	15	26 DEN (S0)
ADD	16	25 ALE (QS0)
NMI	17	24 INTA (QS1)
INTR	18	23 TEST
CLK	19	22 READY
GND	20	21 RESET

Figure 2. 8088 Pin Configuration

Table 1. Pin Description

The following pin function descriptions are for 8088 systems in either minimum or maximum mode. The "local bus" in these descriptions is the direct multiplexed bus interface connection to the 8088 (without regard to additional bus buffers).

Symbol	Pin No.	Type	Name and Function																		
AD7-AD0	9-16	I/O	ADDRESS DATA BUS: These lines constitute the time multiplexed memory/I/O address (T1) and data (T2, T3, Tw, T4) bus. These lines are active HIGH and float to 3-state OFF during interrupt acknowledge and local bus "hold acknowledge".																		
A15-A8	2-8, 39	O	ADDRESS BUS: These lines provide address bits 8 through 15 for the entire bus cycle (T1-T4). These lines do not have to be latched by ALE to remain valid. A15-A8 are active HIGH and float to 3-state OFF during interrupt acknowledge and local bus "hold acknowledge".																		
A19/S6, A18/S5, A17/S4, A16/S3	35-38	O	ADDRESS/STATUS: During T1, these are the four most significant address lines for memory operations. During I/O operations, these lines are LOW. During memory and I/O operations, status information is available on these lines during T2, T3, Tw, and T4. S6 is always low. The status of the interrupt enable flag bit (S5) is updated at the beginning of each clock cycle. S4 and S3 are encoded as shown. This information indicates which segment register is presently being used for data accessing. These lines float to 3-state OFF during local bus "hold acknowledge".																		
			<table border="1"> <thead> <tr> <th>S4</th> <th>S3</th> <th>Characteristics</th> </tr> </thead> <tbody> <tr> <td>0 (LOW)</td> <td>0</td> <td>Alternate Data</td> </tr> <tr> <td>0</td> <td>1</td> <td>Stack</td> </tr> <tr> <td>1 (HIGH)</td> <td>0</td> <td>Code or None</td> </tr> <tr> <td>1</td> <td>1</td> <td>Data</td> </tr> <tr> <td>S6 is 0 (LOW)</td> <td></td> <td></td> </tr> </tbody> </table>	S4	S3	Characteristics	0 (LOW)	0	Alternate Data	0	1	Stack	1 (HIGH)	0	Code or None	1	1	Data	S6 is 0 (LOW)		
S4	S3	Characteristics																			
0 (LOW)	0	Alternate Data																			
0	1	Stack																			
1 (HIGH)	0	Code or None																			
1	1	Data																			
S6 is 0 (LOW)																					
RD	32	O	READ: Read strobe indicates that the processor is performing a memory or I/O read cycle, depending on the state of the IO/M pin or S2. This signal is used to read devices which reside on the 8088 local bus. RD is active LOW during T2, T3 and Tw of any read cycle, and is guaranteed to remain HIGH in T2 until the 8088 local bus has floated. This signal floats to 3-state OFF in "hold acknowledge".																		
READY	22	I	READY: is the acknowledgement from the addressed memory or I/O device that it will complete the data transfer. The RDY signal from memory or I/O is synchronized by the 8284 clock generator to form READY. This signal is active HIGH. The 8088 READY input is not synchronized. Correct operation is not guaranteed if the set up and hold times are not met.																		
INTR	18	I	INTERRUPT REQUEST: is a level triggered input which is sampled during the last clock cycle of each instruction to determine if the processor should enter into an interrupt acknowledge operation. A subroutine is vectored to via an interrupt vector lookup table located in system memory. It can be internally masked by software resetting the interrupt enable bit. INTR is internally synchronized. This signal is active HIGH.																		
TEST	23	I	TEST: input is examined by the "wait for test" instruction. If the TEST input is LOW, execution continues, otherwise the processor waits in an "idle" state. This input is synchronized internally during each clock cycle on the leading edge of CLK.																		

Table 1. Pin Description (Continued)

Symbol	Pin No.	Type	Name and Function
NMI	17	I	NON-MASKABLE INTERRUPT: is an edge triggered input which causes a type 2 interrupt. A subroutine is vectored to via an interrupt vector lookup table located in system memory. NMI is not maskable internally by software. A transition from a LOW to HIGH initiates the interrupt at the end of the current instruction. This input is internally synchronized.
RESET	21	I	RESET: causes the processor to immediately terminate its present activity. The signal must be active HIGH for at least four clock cycles. It restarts execution, as described in the instruction set description, when RESET returns LOW. RESET is internally synchronized.
CLK	19	I	CLOCK: provides the basic timing for the processor and bus controller. It is asymmetric with a 33% duty cycle to provide optimized internal timing.
V _{CC}	40		V_{CC}: is the +5V ±10% power supply pin.
GND	1, 20		GND: are the ground pins.
MN/MX	33	I	MINIMUM/MAXIMUM: indicates what mode the processor is to operate in. The two modes are discussed in the following sections.

The following pin function descriptions are for the 8088 minimum mode (i.e., MN/MX = V_{CC}). Only the pin functions which are unique to minimum mode are described; all other pin functions are as described above.

Symbol	Pin No.	Type	Name and Function
IO/M	28	O	STATUS LINE: is an inverted maximum mode S ₂ . It is used to distinguish a memory access from an I/O access. IO/M becomes valid in the T4 preceding a bus cycle and remains valid until the final T4 of the cycle (I/O = HIGH, M = LOW). IO/M floats to 3-state OFF in local bus "hold acknowledge".
WR	29	O	WRITE: strobe indicates that the processor is performing a write memory or write I/O cycle, depending on the state of the IO/M signal. WR is active for T ₂ , T ₃ , and T _w of any write cycle. It is active LOW, and floats to 3-state OFF in local bus "hold acknowledge".
INTA	24	O	INTA: is used as a read strobe for interrupt acknowledge cycles. It is active LOW during T ₂ , T ₃ , and T _w of each interrupt acknowledge cycle.
ALE	25	O	ADDRESS LATCH ENABLE: is provided by the processor to latch the address into an address latch. It is a HIGH pulse active during clock low of T ₁ of any bus cycle. Note that ALE is never floated.
DT/R	27	O	DATA TRANSMIT/RECEIVE: is needed in a minimum system that desires to use a data bus transceiver. It is used to control the direction of data flow through the transceiver. Logically, DT/R is equivalent to S ₁ in the maximum mode, and its timing is the same as for IO/M (T = HIGH, R = LOW). This signal floats to 3-state OFF in local "hold acknowledge".
DEN	26	O	DATA ENABLE: is provided as an output enable for the data bus transceiver in a minimum system which uses the transceiver. DEN is active LOW during each memory and I/O access, and for INTA cycles. For a read or INTA cycle, it is active from the middle of T ₂ until the middle of T ₄ , while for a write cycle, it is active from the beginning of T ₂ until the middle of T ₄ . DEN floats to 3-state OFF during local bus "hold acknowledge".

Table 1. Pin Description (Continued)

Symbol	Pin No.	Type	Name and Function					
HOLD, HLDA	31, 30	I, O	HOLD: indicates that another master is requesting a local bus "hold". To be acknowledged, HOLD must be active HIGH. The processor receiving the "hold" request will issue HLDA (HIGH) as an acknowledgement, in the middle of a T4 or Ti clock cycle. Simultaneous with the issuance of HLDA the processor will float the local bus and control lines. After HOLD is detected as being LOW, the processor lowers HLDA, and when the processor needs to run another cycle, it will again drive the local bus and control lines. HOLD and HLDA have internal pull-up resistors. Hold is not an asynchronous input. External synchronization should be provided if the system cannot otherwise guarantee the set up time.					
SSO	34	O	STATUS LINE: is logically equivalent to $\overline{S_0}$ in the maximum mode. The combination of $\overline{S_0}$, $\overline{IO/M}$ and $\overline{DT/R}$ allows the system to completely decode the current bus cycle status.					
			IO/\overline{M}	DT/\overline{R}	$\overline{S_0}$	Characteristics		
			1(HIGH) 1 1 1 0(LOW) 0 0 0	0 0 1 1 0 0 1 1	0 1 0 1 0 1 0 1	Interrupt Acknowledge Read I/O Port Write I/O Port Halt Code Access Read Memory Write Memory Passive		

The following pin function descriptions are for the 8088/8288 system in maximum mode (i.e., $MN/MX = GND$). Only the pin functions which are unique to maximum mode are described; all other pin functions are as described above.

Symbol	Pin No.	Type	Name and Function					
S2, S1, S0	26–28	O	STATUS: is active during clock high of T4, T1, and T2, and is returned to the passive state (1,1,1) during T3 or during Tw when READY is HIGH. This status is used by the 8288 bus controller to generate all memory and I/O access control signals. Any change by $\overline{S_2}$, $\overline{S_1}$, or $\overline{S_0}$ during T4 is used to indicate the beginning of a bus cycle, and the return to the passive state in T3 and Tw is used to indicate the end of a bus cycle. These signals float to 3-state OFF during "hold acknowledge". During the first clock cycle after RESET becomes active, these signals are active HIGH. After this first clock, they float to 3-state OFF.					
			S2 S1 S0 Characteristics					
			0(LOW) 0 0 0 1(HIGH) 1 1 1	0 0 1 1 0 0 1 1	0 1 0 1 0 1 0 1	Interrupt Acknowledge Read I/O Port Write I/O Port Halt Code Access Read Memory Write Memory Passive		

Table 1. Pin Description (Continued)

Symbol	Pin No.	Type	Name and Function															
RQ/GT0, RQ/GT1	30, 31	I/O	<p>REQUEST/GANT: pins are used by other local bus masters to force the processor to release the local bus at the end of the processor's current bus cycle. Each pin is bidirectional with $\overline{RQ}/\overline{GT0}$ having higher priority than $\overline{RQ}/\overline{GT1}$. RQ/GT has an internal pull-up resistor, so may be left unconnected. The request/grant sequence is as follows (See Figure 8):</p> <ol style="list-style-type: none"> 1. A pulse of one CLK wide from another local bus master indicates a local bus request ("hold") to the 8088 (pulse 1). 2. During a T4 or T1 clock cycle, a pulse one clock wide from the 8088 to the requesting master (pulse 2), indicates that the 8088 has allowed the local bus to float and that it will enter the "hold acknowledge" state at the next CLK. The CPU's bus interface unit is disconnected logically from the local bus during "hold acknowledge". The same rules as for HOLD/HOLDA apply as for when the bus is released. 3. A pulse one CLK wide from the requesting master indicates to the 8088 (pulse 3) that the "hold" request is about to end and that the 8088 can reclaim the local bus at the next CLK. The CPU then enters T4. <p>Each master-master exchange of the local bus is a sequence of three pulses. There must be one idle CLK cycle after each bus exchange. Pulses are active LOW.</p> <p>If the request is made while the CPU is performing a memory cycle, it will release the local bus during T4 of the cycle when all the following conditions are met:</p> <ol style="list-style-type: none"> 1. Request occurs on or before T2. 2. Current cycle is not the low bit of a word. 3. Current cycle is not the first acknowledge of an interrupt acknowledge sequence. 4. A locked instruction is not currently executing. <p>If the local bus is idle when the request is made the two possible events will follow:</p> <ol style="list-style-type: none"> 1. Local bus will be released during the next clock. 2. A memory cycle will start within 3 clocks. Now the four rules for a currently active memory cycle apply with condition number 1 already satisfied. 															
LOCK	29	O	<p>LOCK: indicates that other system bus masters are not to gain control of the system bus while LOCK is active (LOW). The LOCK signal is activated by the "LOCK" prefix instruction and remains active until the completion of the next instruction. This signal is active LOW, and floats to 3-state off in "hold acknowledge".</p>															
QS1, QS0	24, 25	O	<p>QUEUE STATUS: provide status to allow external tracking of the internal 8088 instruction queue.</p> <p>The queue status is valid during the CLK cycle after which the queue operation is performed.</p> <table border="1"> <thead> <tr> <th>QS1</th> <th>QS0</th> <th>Characteristics</th> </tr> </thead> <tbody> <tr> <td>0(LOW)</td> <td>0</td> <td>No Operation</td> </tr> <tr> <td>0</td> <td>1</td> <td>First Byte of Opcode from Queue</td> </tr> <tr> <td>1(HIGH)</td> <td>0</td> <td>Empty the Queue</td> </tr> <tr> <td>1</td> <td>1</td> <td>Subsequent Byte from Queue</td> </tr> </tbody> </table>	QS1	QS0	Characteristics	0(LOW)	0	No Operation	0	1	First Byte of Opcode from Queue	1(HIGH)	0	Empty the Queue	1	1	Subsequent Byte from Queue
QS1	QS0	Characteristics																
0(LOW)	0	No Operation																
0	1	First Byte of Opcode from Queue																
1(HIGH)	0	Empty the Queue																
1	1	Subsequent Byte from Queue																
—	34	O	Pin 34 is always high in the maximum mode.															

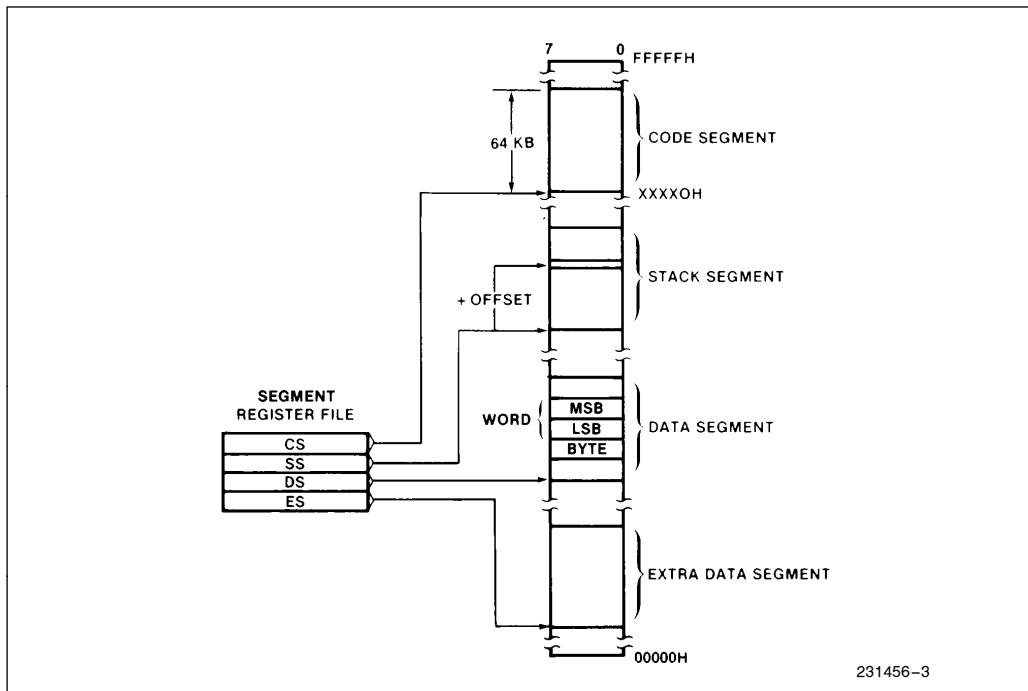


Figure 3. Memory Organization

FUNCTIONAL DESCRIPTION

Memory Organization

The processor provides a 20-bit address to memory which locates the byte being referenced. The memory is organized as a linear array of up to 1 million bytes, addressed as 00000(H) to FFFFF(H). The memory is logically divided into code, data, extra data, and stack segments of up to 64K bytes each, with each segment falling on 16-byte boundaries (See Figure 3).

All memory references are made relative to base addresses contained in high speed segment registers. The segment types were chosen based on the ad-

dressing needs of programs. The segment register to be selected is automatically chosen according to the rules of the following table. All information in one segment type share the same logical attributes (e.g. code or data). By structuring memory into relocatable areas of similar characteristics and by automatically selecting segment registers, programs are shorter, faster, and more structured.

Word (16-bit) operands can be located on even or odd address boundaries. For address and data operands, the least significant byte of the word is stored in the lower valued address location and the most significant byte in the next higher address location. The BIU will automatically execute two fetch or write cycles for 16-bit operands.

Memory Reference Used	Segment Register Used	Segment Selection Rule
Instructions	CODE (CS)	Automatic with all instruction prefetch.
Stack	STACK (SS)	All stack pushes and pops. Memory references relative to BP base register except data references.
Local Data	DATA (DS)	Data references when: relative to stack, destination of string operation, or explicitly overridden.
External (Global) Data	EXTRA (ES)	Destination of string operations: Explicitly selected using a segment override.

Certain locations in memory are reserved for specific CPU operations (See Figure 4). Locations from addresses FFFF0H through FFFFFH are reserved for operations including a jump to the initial system initialization routine. Following RESET, the CPU will always begin execution at location FFFF0H where the jump must be located. Locations 00000H through 003FFH are reserved for interrupt operations. Four-byte pointers consisting of a 16-bit segment address and a 16-bit offset address direct program flow to one of the 256 possible interrupt service routines. The pointer elements are assumed to have been stored at their respective places in reserved memory prior to the occurrence of interrupts.

Minimum and Maximum Modes

The requirements for supporting minimum and maximum 8088 systems are sufficiently different that they cannot be done efficiently with 40 uniquely defined pins. Consequently, the 8088 is equipped with a strap pin (MN/M_X) which defines the system con-

figuration. The definition of a certain subset of the pins changes, dependent on the condition of the strap pin. When the MN/M_X pin is strapped to GND, the 8088 defines pins 24 through 31 and 34 in maximum mode. When the MN/M_X pin is strapped to V_{CC}, the 8088 generates bus control signals itself on pins 24 through 31 and 34.

The minimum mode 8088 can be used with either a multiplexed or demultiplexed bus. The multiplexed bus configuration is compatible with the MCS-85 multiplexed bus peripherals. This configuration (See Figure 5) provides the user with a minimum chip count system. This architecture provides the 8088 processing power in a highly integrated form.

The demultiplexed mode requires one latch (for 64K addressability) or two latches (for a full megabyte of addressing). A third latch can be used for buffering if the address bus loading requires it. A transceiver can also be used if data bus buffering is required (See Figure 6). The 8088 provides DEN and DT/R to control the transceiver, and ALE to latch the addresses. This configuration of the minimum mode provides the standard demultiplexed bus structure with heavy bus buffering and relaxed bus timing requirements.

The maximum mode employs the 8288 bus controller (See Figure 7). The 8288 decodes status lines S₀, S₁, and S₂, and provides the system with all bus control signals. Moving the bus control to the 8288 provides better source and sink current capability to the control lines, and frees the 8088 pins for extended large system features. Hardware lock, queue status, and two request/grant interfaces are provided by the 8088 in maximum mode. These features allow co-processors in local bus and remote bus configurations.

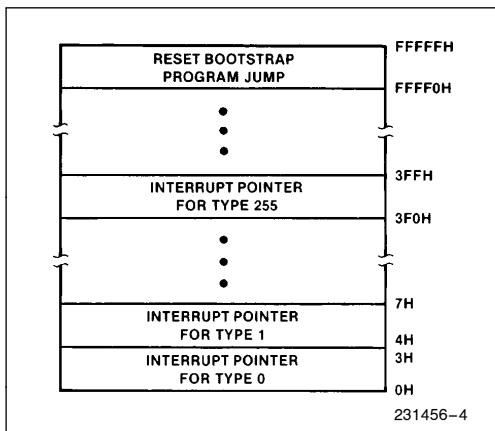


Figure 4. Reserved Memory Locations

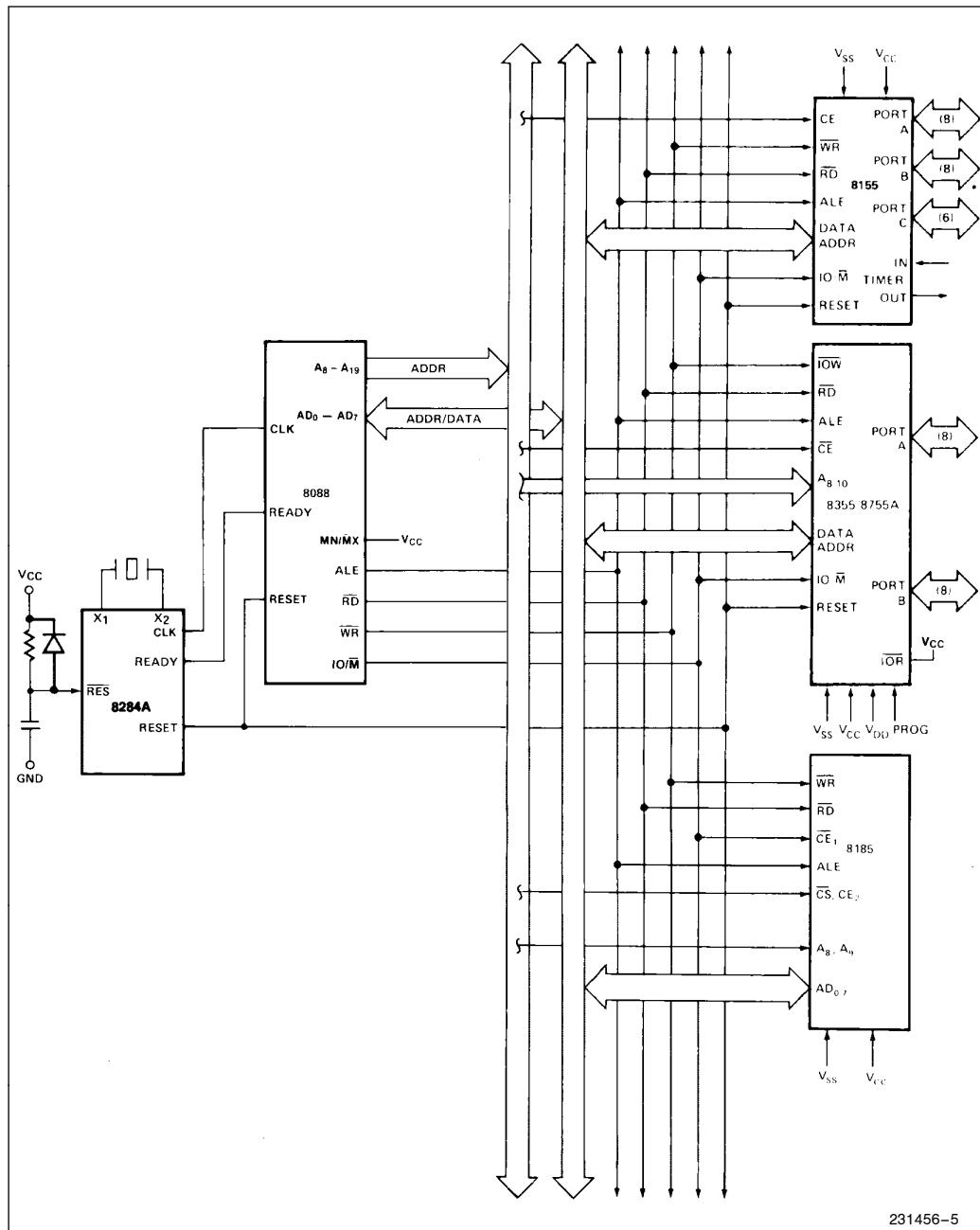


Figure 5. Multiplexed Bus Configuration

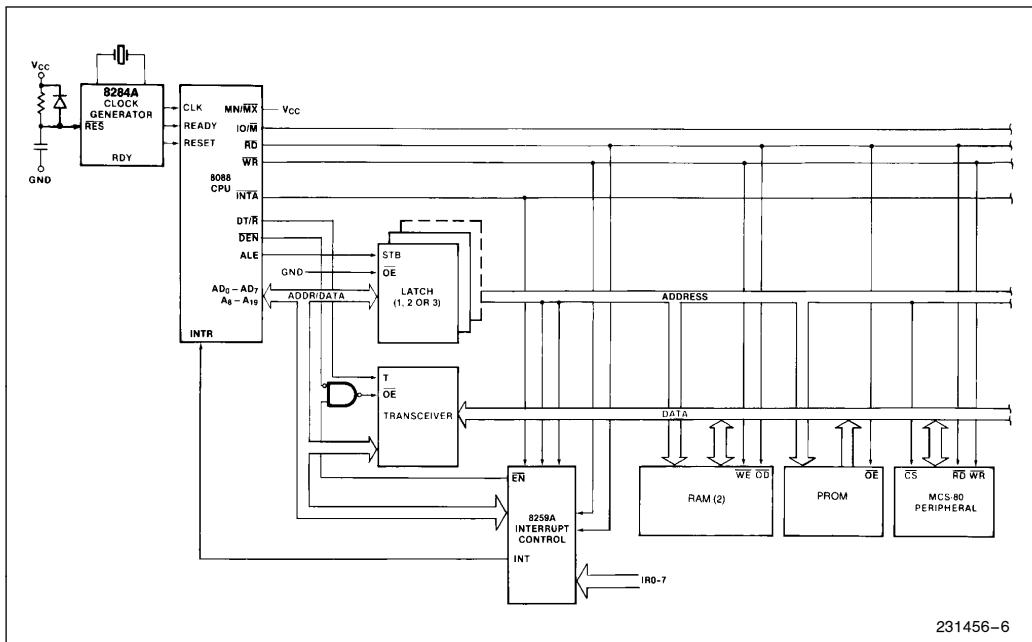


Figure 6. Demultiplexed Bus Configuration

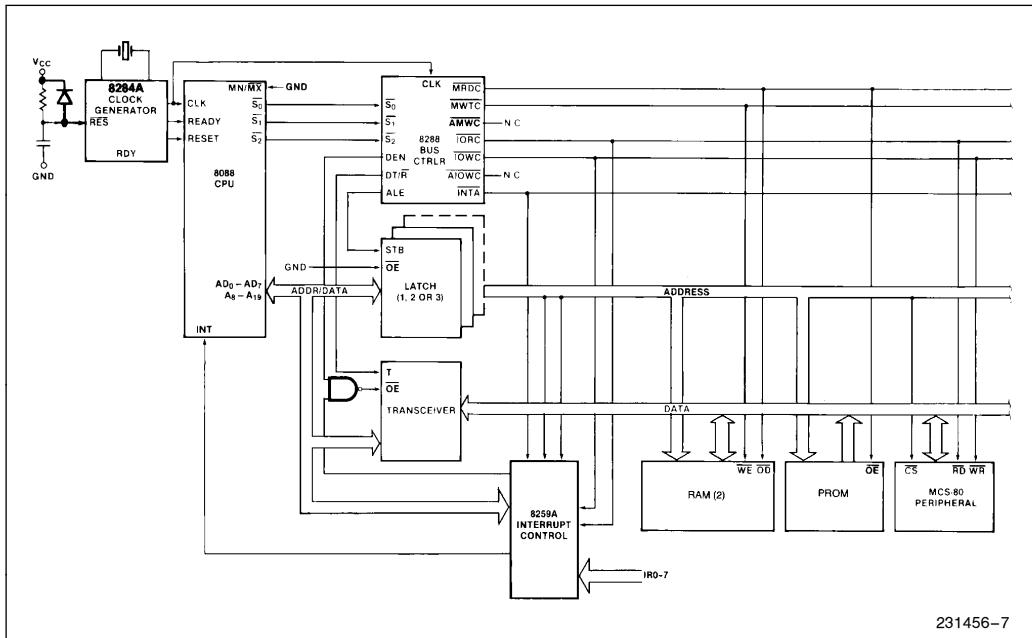


Figure 7. Fully Buffered System Using Bus Controller

Bus Operation

The 8088 address/data bus is broken into three parts—the lower eight address/data bits (AD0–AD7), the middle eight address bits (A8–A15), and the upper four address bits (A16–A19). The address/data bits and the highest four address bits are time multiplexed. This technique provides the most efficient use of pins on the processor, permitting the use of a standard 40 lead package. The middle eight address bits are not multiplexed, i.e. they remain val-

id throughout each bus cycle. In addition, the bus can be demultiplexed at the processor with a single address latch if a standard, non-multiplexed bus is desired for the system.

Each processor bus cycle consists of at least four CLK cycles. These are referred to as T1, T2, T3, and T4 (See Figure 8). The address is emitted from the processor during T1 and data transfer occurs on the bus during T3 and T4. T2 is used primarily for chang-

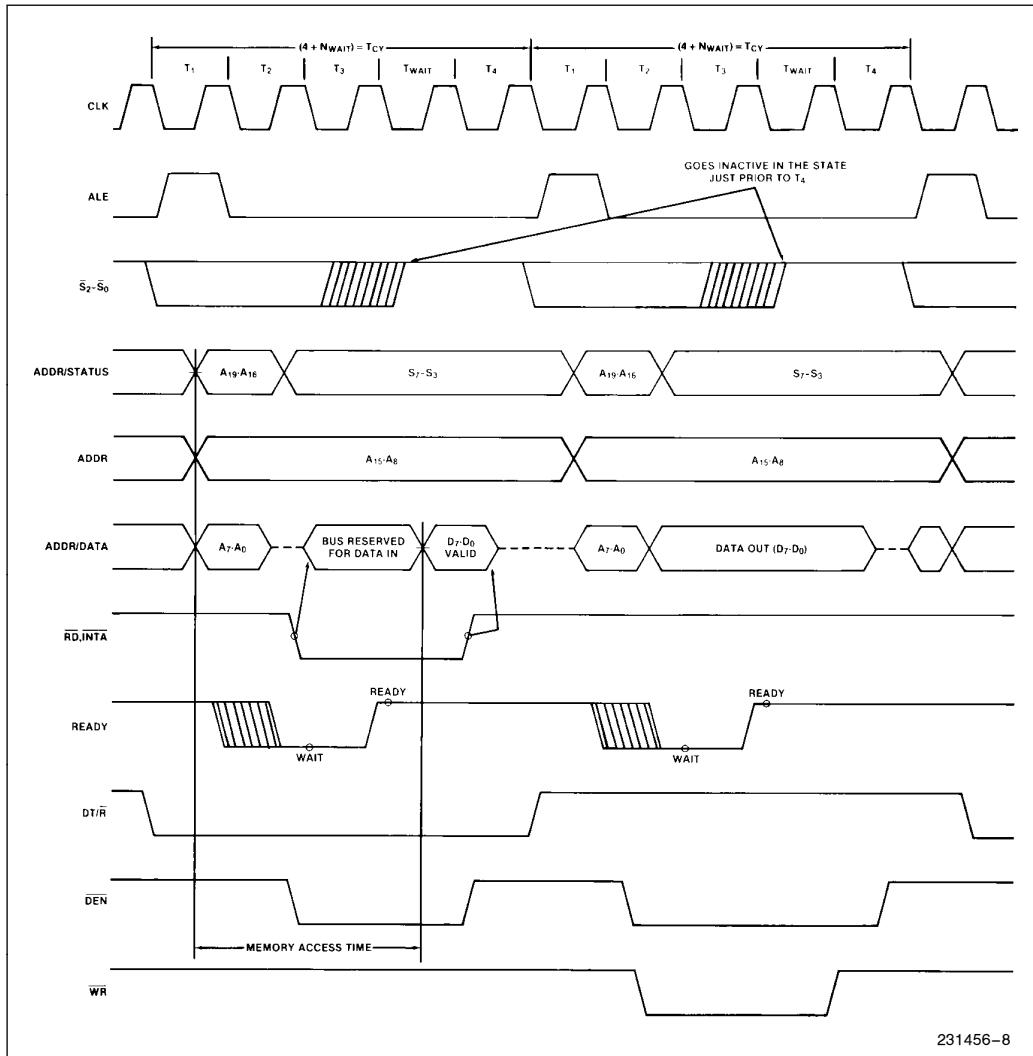


Figure 8. Basic System Timing

ing the direction of the bus during read operations. In the event that a "NOT READY" indication is given by the addressed device, "wait" states (T_w) are inserted between T_3 and T_4 . Each inserted "wait" state is of the same duration as a CLK cycle. Periods can occur between 8088 driven bus cycles. These are referred to as "idle" states (T_i), or inactive CLK cycles. The processor uses these cycles for internal housekeeping.

During T_1 of any bus cycle, the ALE (address latch enable) signal is emitted (by either the processor or the 8288 bus controller, depending on the MN/MX strap). At the trailing edge of this pulse, a valid address and certain status information for the cycle may be latched.

Status bits S_0 , S_1 , and S_2 are used by the bus controller, in maximum mode, to identify the type of bus transaction according to the following table:

S_2	S_1	S_0	Characteristics
0(LOW)	0	0	Interrupt Acknowledge
0	0	1	Read I/O
0	1	0	Write I/O
0	1	1	Halt
1(HIGH)	0	0	Instruction Fetch
1	0	1	Read Data from Memory
1	1	0	Write Data to Memory
1	1	1	Passive (No Bus Cycle)

Status bits S_3 through S_6 are multiplexed with high order address bits and are therefore valid during T_2 through T_4 . S_3 and S_4 indicate which segment register was used for this bus cycle in forming the address according to the following table:

S_4	S_3	Characteristics
0(LOW)	0	Alternate Data (Extra Segment)
0	1	Stack
1(HIGH)	0	Code or None
1	1	Data

S_5 is a reflection of the PSW interrupt enable bit. S_6 is always equal to 0.

I/O Addressing

In the 8088, I/O operations can address up to a maximum of 64K I/O registers. The I/O address appears in the same format as the memory address on bus lines A15–A0. The address lines A19–A16 are zero in I/O operations. The variable I/O instructions,

which use register DX as a pointer, have full address capability, while the direct I/O instructions directly address one or two of the 256 I/O byte locations in page 0 of the I/O address space. I/O ports are addressed in the same manner as memory locations.

Designers familiar with the 8085 or upgrading an 8085 design should note that the 8085 addresses I/O with an 8-bit address on both halves of the 16-bit address bus. The 8088 uses a full 16-bit address on its lower 16 address lines.

EXTERNAL INTERFACE

Processor Reset and Initialization

Processor initialization or start up is accomplished with activation (HIGH) of the RESET pin. The 8088 RESET is required to be HIGH for greater than four clock cycles. The 8088 will terminate operations on the high-going edge of RESET and will remain dormant as long as RESET is HIGH. The low-going transition of RESET triggers an internal reset sequence for approximately 7 clock cycles. After this interval the 8088 operates normally, beginning with the instruction in absolute locations FFFF0H (See Figure 4). The RESET input is internally synchronized to the processor clock. At initialization, the HIGH to LOW transition of RESET must occur no sooner than 50 μ s after power up, to allow complete initialization of the 8088.

NMI asserted prior to the 2nd clock after the end of RESET will not be honored. If NMI is asserted after that point and during the internal reset sequence, the processor may execute one instruction before responding to the interrupt. A hold request active immediately after RESET will be honored before the first instruction fetch.

All 3-state outputs float to 3-state OFF during RESET. Status is active in the idle state for the first clock after RESET becomes active and then floats to 3-state OFF. ALE and HLDA are driven low.

Interrupt Operations

Interrupt operations fall into two classes: software or hardware initiated. The software initiated interrupts and software aspects of hardware interrupts are specified in the instruction set description in the iAPX 88 book or the iAPX 86,88 User's Manual. Hardware interrupts can be classified as nonmaskable or maskable.



Interrupts result in a transfer of control to a new program location. A 256 element table containing address pointers to the interrupt service program locations resides in absolute locations 0 through 3FFH (See Figure 4), which are reserved for this purpose. Each element in the table is 4 bytes in size and corresponds to an interrupt "type." An interrupting device supplies an 8-bit type number, during the interrupt acknowledge sequence, which is used to vector through the appropriate element to the new interrupt service program location.

Non-Maskable Interrupt (NMI)

The processor provides a single non-maskable interrupt (NMI) pin which has higher priority than the maskable interrupt request (INTR) pin. A typical use would be to activate a power failure routine. The NMI is edge-triggered on a LOW to HIGH transition. The activation of this pin causes a type 2 interrupt.

NMI is required to have a duration in the HIGH state of greater than two clock cycles, but is not required to be synchronized to the clock. Any higher going transition of NMI is latched on-chip and will be serviced at the end of the current instruction or between whole moves (2 bytes in the case of word moves) of a block type instruction. Worst case response to NMI would be for multiply, divide, and variable shift instructions. There is no specification on the occurrence of the low-going edge; it may occur before, during, or after the servicing of NMI. Another high-going edge triggers another response if it occurs after the start of the NMI procedure. The signal must be free of logical spikes in general and be free of bounces on the low-going edge to avoid triggering extraneous responses.

Maskable Interrupt (INTR)

The 8088 provides a single interrupt request input (INTR) which can be masked internally by software with the resetting of the interrupt enable (IF) flag bit. The interrupt request signal is level triggered. It is internally synchronized during each clock cycle on the high-going edge of CLK. To be responded to, INTR must be present (HIGH) during the clock period preceding the end of the current instruction or the end of a whole move for a block type instruction. During interrupt response sequence, further interrupts are disabled. The enable bit is reset as part of the response to any interrupt (INTR, NMI, software interrupt, or single step), although the FLAGS register which is automatically pushed onto the stack reflects the state of the processor prior to the interrupt. Until the old FLAGS register is restored, the

enable bit will be zero unless specifically set by an instruction.

During the response sequence (See Figure 9), the processor executes two successive (back to back) interrupt acknowledge cycles. The 8088 emits the LOCK signal (maximum mode only) from T2 of the first bus cycle until T2 of the second. A local bus "hold" request will not be honored until the end of the second bus cycle. In the second bus cycle, a byte is fetched from the external interrupt system (e.g., 8259A PIC) which identifies the source (type) of the interrupt. This byte is multiplied by four and used as a pointer into the interrupt vector lookup table. An INTR signal left HIGH will be continually responded to within the limitations of the enable bit and sample period. The interrupt return instruction includes a flags pop which returns the status of the original interrupt enable bit when it restores the flags.

HALT

When a software HALT instruction is executed, the processor indicates that it is entering the HALT state in one of two ways, depending upon which mode is strapped. In minimum mode, the processor issues ALE, delayed by one clock cycle, to allow the system to latch the halt status. Halt status is available on IO/M, DT/R, and SSO. In maximum mode, the processor issues appropriate HALT status on \overline{S}_2 , \overline{S}_1 , and \overline{S}_0 , and the 8288 bus controller issues one ALE. The 8088 will not leave the HALT state when a local bus hold is entered while in HALT. In this case, the processor reissues the HALT indicator at the end of the local bus hold. An interrupt request or RESET will force the 8088 out of the HALT state.

Read/Modify/Write (Semaphore) Operations via LOCK

The LOCK status information is provided by the processor when consecutive bus cycles are required during the execution of an instruction. This allows the processor to perform read/modify/write operations on memory (via the "exchange register with memory" instruction), without another system bus master receiving intervening memory cycles. This is useful in multiprocessor system configurations to accomplish "test and set lock" operations. The LOCK signal is activated (LOW) in the clock cycle following decoding of the LOCK prefix instruction. It is deactivated at the end of the last bus cycle of the instruction following the LOCK prefix. While LOCK is active, a request on a RQ/GT pin will be recorded, and then honored at the end of the LOCK.

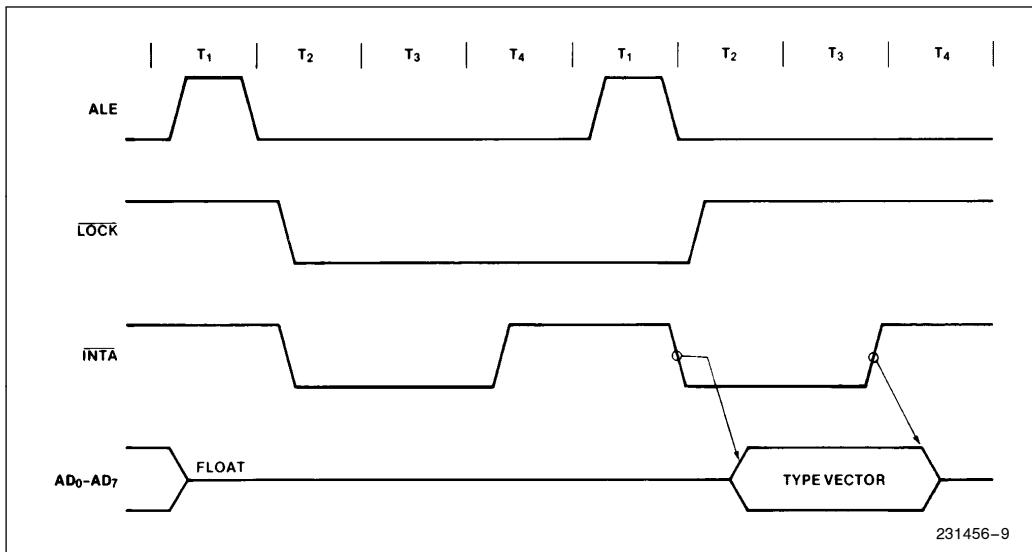


Figure 9. Interrupt Acknowledge Sequence

External Synchronization via TEST

As an alternative to interrupts, the 8088 provides a single software-testable input pin (TEST). This input is utilized by executing a WAIT instruction. The single WAIT instruction is repeatedly executed until the TEST input goes active (LOW). The execution of WAIT does not consume bus cycles once the queue is full.

If a local bus request occurs during WAIT execution, the 8088 3-states all output drivers. If interrupts are enabled, the 8088 will recognize interrupts and process them. The WAIT instruction is then refetched, and reexecuted.

Basic System Timing

In minimum mode, the MN/MX pin is strapped to V_{CC} and the processor emits bus control signals compatible with the 8085 bus structure. In maximum mode, the MN/MX pin is strapped to GND and the processor emits coded status information which the 8288 bus controller uses to generate MULTIBUS compatible bus control signals.

System Timing—Minimum System

(See Figure 8)

The read cycle begins in T₁ with the assertion of the address latch enable (ALE) signal. The trailing (low

going) edge of this signal is used to latch the address information, which is valid on the address/data bus (AD₀-AD₇) at this time, into the 8282/8283 latch. Address lines A₈ through A₁₅ do not need to be latched because they remain valid throughout the bus cycle. From T₁ to T₄ the IO/M signal indicates a memory or I/O operation. At T₂ the address is removed from the address/data bus and the bus goes to a high impedance state. The read control signal is also asserted at T₂. The read (\bar{RD}) signal causes the addressed device to enable its data bus drivers to the local bus. Some time later, valid data will be available on the bus and the addressed device will drive the READY line HIGH. When the processor returns the read signal to a HIGH level, the addressed device will again 3-state its bus drivers. If a transceiver is required to buffer the 8088 local bus, signals DT/R and DEN are provided by the 8088.

A write cycle also begins with the assertion of ALE and the emission of the address. The IO/M signal is again asserted to indicate a memory or I/O write operation. In T₂, immediately following the address emission, the processor emits the data to be written into the addressed location. This data remains valid until at least the middle of T₄. During T₂, T₃, and T₄, the processor asserts the write control signal. The write (\bar{WR}) signal becomes active at the beginning of T₂, as opposed to the read, which is delayed somewhat into T₂ to provide time for the bus to float.

The basic difference between the interrupt acknowledge cycle and a read cycle is that the interrupt acknowledge (INTA) signal is asserted in place of the read (\overline{RD}) signal and the address bus is floated. (See Figure 9) In the second of two successive INTA cycles, a byte of information is read from the data bus, as supplied by the interrupt system logic (i.e. 8259A priority interrupt controller). This byte identifies the source (type) of the interrupt. It is multiplied by four and used as a pointer into the interrupt vector lookup table, as described earlier.

Bus Timing—Medium Complexity Systems

(See Figure 10)

For medium complexity systems, the MN/MX pin is connected to GND and the 8288 bus controller is added to the system, as well as a latch for latching the system address, and a transceiver to allow for bus loading greater than the 8088 is capable of handling. Signals ALE, \overline{DEN} , and DT/ \overline{R} are generated by the 8288 instead of the processor in this configuration, although their timing remains relatively the same. The 8088 status outputs (S_2 , S_1 , and S_0) provide type of cycle information and become 8288 inputs. This bus cycle information specifies read (code, data, or I/O), write (data or I/O), interrupt acknowledge, or software halt. The 8288 thus issues control signals specifying memory read or write, I/O read or write, or interrupt acknowledge. The 8288 provides two types of write strobes, normal and advanced, to be applied as required. The normal write strobes have data valid at the leading edge of write. The advanced write strobes have the same timing as read strobes, and hence, data is not valid at the leading edge of write. The transceiver receives the usual T and \overline{OE} inputs from the 8288's DT/ \overline{R} and \overline{DEN} outputs.

The pointer into the interrupt vector table, which is passed during the second INTA cycle, can derive from an 8259A located on either the local bus or the system bus. If the master 8289A priority interrupt controller is positioned on the local bus, a TTL gate is required to disable the transceiver when reading from the master 8259A during the interrupt acknowledge sequence and software "pol".

The 8088 Compared to the 8086

The 8088 CPU is an 8-bit processor designed around the 8086 internal structure. Most internal functions of the 8088 are identical to the equivalent 8086 functions. The 8088 handles the external bus

the same way the 8086 does with the distinction of handling only 8 bits at a time. Sixteen-bit operands are fetched or written in two consecutive bus cycles. Both processors will appear identical to the software engineer, with the exception of execution time. The internal register structure is identical and all instructions have the same end result. The differences between the 8088 and 8086 are outlined below. The engineer who is unfamiliar with the 8086 is referred to the iAPX 86, 88 User's Manual, Chapters 2 and 4, for function description and instruction set information. Internally, there are three differences between the 8088 and the 8086. All changes are related to the 8-bit bus interface.

- The queue length is 4 bytes in the 8088, whereas the 8086 queue contains 6 bytes, or three words. The queue was shortened to prevent overuse of the bus by the BIU when prefetching instructions. This was required because of the additional time necessary to fetch instructions 8 bits at a time.
- To further optimize the queue, the prefetching algorithm was changed. The 8088 BIU will fetch a new instruction to load into the queue each time there is a 1 byte hole (space available) in the queue. The 8086 waits until a 2-byte space is available.
- The internal execution time of the instruction set is affected by the 8-bit interface. All 16-bit fetches and writes from/to memory take an additional four clock cycles. The CPU is also limited by the speed of instruction fetches. This latter problem only occurs when a series of simple operations occur. When the more sophisticated instructions of the 8088 are being used, the queue has time to fill and the execution proceeds as fast as the execution unit will allow.

The 8088 and 8086 are completely software compatible by virtue of their identical execution units. Software that is system dependent may not be completely transferable, but software that is not system dependent will operate equally as well on an 8088 and an 8086.

The hardware interface of the 8088 contains the major differences between the two CPUs. The pin assignments are nearly identical, however, with the following functional changes:

- A8–A15—These pins are only address outputs on the 8088. These address lines are latched internally and remain valid throughout a bus cycle in a manner similar to the 8085 upper address lines.
- \overline{BHE} has no meaning on the 8088 and has been eliminated.

- \overline{SSO} provides the \overline{SO} status information in the minimum mode. This output occurs on pin 34 in minimum mode only. DT/R, IO/M, and \overline{SSO} provide the complete bus status in minimum mode.
- IO/M has been inverted to be compatible with the MCS-85 bus structure.
- ALE is delayed by one clock cycle in the minimum mode when entering HALT, to allow the status to be latched with ALE.

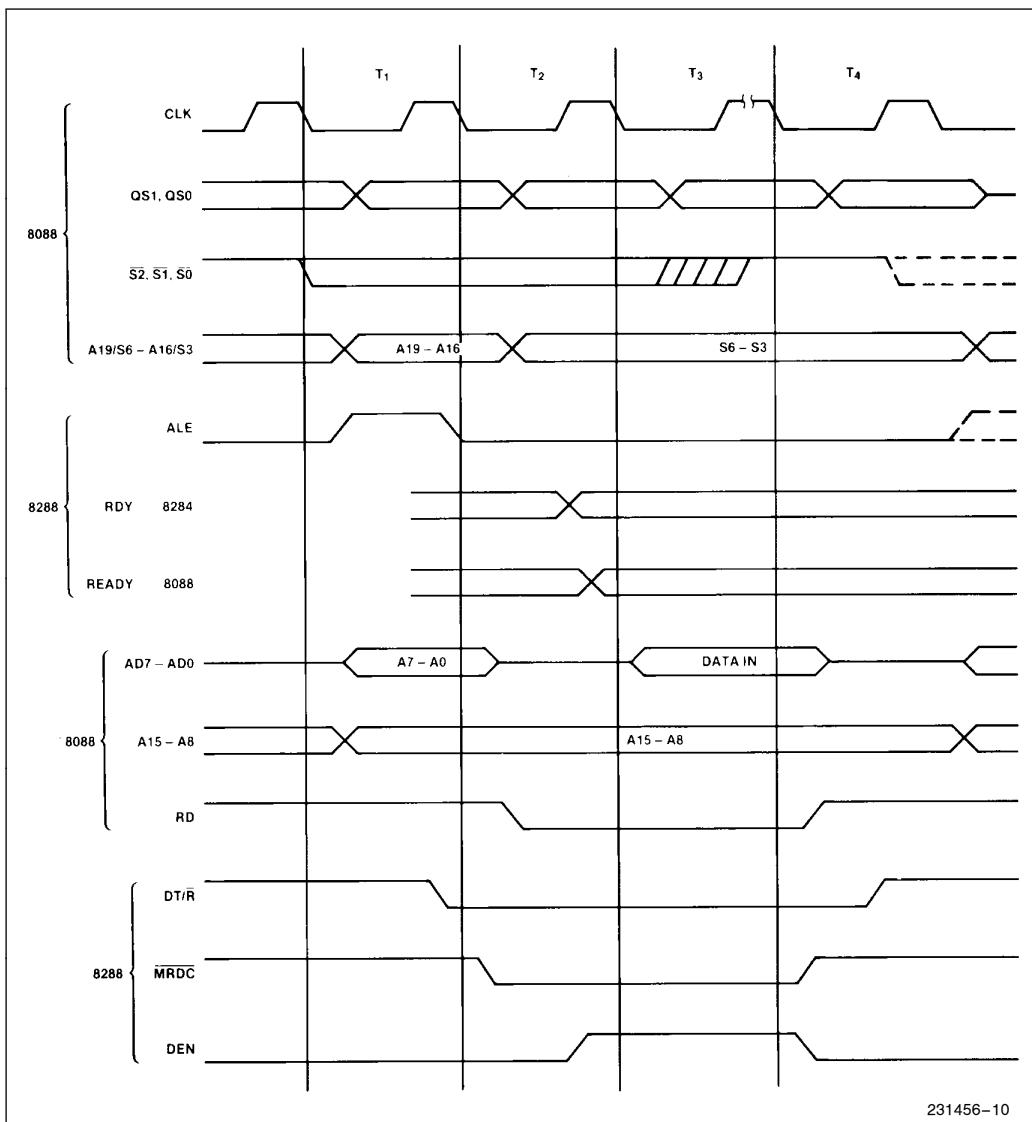


Figure 10. Medium Complexity System Timing

ABSOLUTE MAXIMUM RATINGS*

Ambient Temperature Under Bias 0°C to + 70°C
 Case Temperature (Plastic) 0°C to + 95°C
 Case Temperature (CERDIP) 0°C to + 75°C
 Storage Temperature - 65°C to + 150°C
 Voltage on Any Pin with
 Respect to Ground - 1.0 to + 7V
 Power Dissipation 2.5 Watt

NOTICE: This is a production data sheet. The specifications are subject to change without notice.

**WARNING: Stressing the device beyond the "Absolute Maximum Ratings" may cause permanent damage. These are stress ratings only. Operation beyond the "Operating Conditions" is not recommended and extended exposure beyond the "Operating Conditions" may affect device reliability.*

D.C. CHARACTERISTICS

($T_A = 0^\circ\text{C}$ to 70°C , $T_{\text{CASE}} (\text{Plastic}) = 0^\circ\text{C}$ to 95°C , $T_{\text{CASE}} (\text{CERDIP}) = 0^\circ\text{C}$ to 75°C ,
 $T_A = 0^\circ\text{C}$ to 55°C and $T_{\text{CASE}} = 0^\circ\text{C}$ to 75°C for P8088-2 only
 T_A is guaranteed as long as T_{CASE} is not exceeded)

($V_{CC} = 5\text{ V} \pm 10\%$ for 8088, $V_{CC} = 5\text{ V} \pm 5\%$ for 8088-2 and Extended Temperature EXPRESS)

Symbol	Parameter	Min	Max	Units	Test Conditions
V_{IL}	Input Low Voltage	-0.5	+0.8	V	(Note 1)
V_{IH}	Input High Voltage	2.0	$V_{CC} + 0.5$	V	(Notes 1, 2)
V_{OL}	Output Low Voltage		0.45	V	$I_{OL} = 2.0\text{ mA}$
V_{OH}	Output High Voltage	2.4		V	$I_{OH} = -400\text{ }\mu\text{A}$
I_{CC}	8088 Power Supply Current: 8088-2 P8088	340 350 250		mA	$T_A = 25^\circ\text{C}$
I_{LI}	Input Leakage Current		± 10	μA	$0\text{V} \leq V_{IN} \leq V_{CC}$ (Note 3)
I_{LO}	Output and I/O Leakage Current		± 10	μA	$0.45\text{V} \leq V_{OUT} \leq V_{CC}$
V_{CL}	Clock Input Low Voltage	-0.5	+0.6	V	
V_{CH}	Clock Input High Voltage	3.9	$V_{CC} + 1.0$	V	
C_{IN}	Capacitance If Input Buffer (All Input Except AD_0 - AD_7 , RQ/GT)		15	pF	$f_c = 1\text{ MHz}$
C_{IO}	Capacitance of I/O Buffer AD_0 - AD_7 , RQ/GT)		15	pF	$f_c = 1\text{ MHz}$

NOTES:

1. V_{IL} tested with MN/\overline{MX} Pin = 0V
 V_{IH} tested with MN/MX Pin = 5V
 MN/\overline{MX} Pin is a strap Pin
2. Not applicable to RQ/GT0 and $\overline{RQ}/\overline{GT1}$ Pins (Pins 30 and 31)
3. HOLD and HLDA I_{LI} Min = 30 μA , Max = 500 μA

A.C. CHARACTERISTICS

($T_A = 0^\circ\text{C}$ to 70°C , $T_{\text{CASE}} (\text{Plastic}) = 0^\circ\text{C}$ to 95°C , $T_{\text{CASE}} (\text{CERDIP}) = 0^\circ\text{C}$ to 75°C ,
 $T_A = 0^\circ\text{C}$ to 55°C and $T_{\text{CASE}} = 0^\circ\text{C}$ to 80°C for P8088-2 only
 T_A is guaranteed as long as T_{CASE} is not exceeded)

($V_{\text{CC}} = 5\text{V} \pm 10\%$ for 8088, $V_{\text{CC}} = 5\text{V} \pm 5\%$ for 8088-2 and Extended Temperature EXPRESS)

MINIMUM COMPLEXITY SYSTEM TIMING REQUIREMENTS

Symbol	Parameter	8088		8088-2		Units	Test Conditions
		Min	Max	Min	Max		
TCLCL	CLK Cycle Period	200	500	125	500	ns	
TCLCH	CLK Low Time	118		68		ns	
TCHCL	CLK High Time	69		44		ns	
TCH1CH2	CLK Rise Time		10		10	ns	From 1.0V to 3.5V
TCL2CL2	CLK Fall Time		10		10	ns	From 3.5V to 1.0V
TDVCL	Data in Setup Time	30		20		ns	
TCLDX	Data in Hold Time	10		10		ns	
TR1VCL	RDY Setup Time into 8284 (Notes 1, 2)	35		35		ns	
TCLR1X	RDY Hold Time into 8284 (Notes 1, 2)	0		0		ns	
TRYHCH	READY Setup Time into 8088	118		68		ns	
TCHRYX	READY Hold Time into 8088	30		20		ns	
TRYLCL	READY Inactive to CLK (Note 3)	-8		-8		ns	
THVCH	HOLD Setup Time	35		20		ns	
TINVCH	INTR, NMI, TEST Setup Time (Note 2)	30		15		ns	
TILIH	Input Rise Time (Except CLK)		20		20	ns	From 0.8V to 2.0V
TIHIL	Input Fall Time (Except CLK)		12		12	ns	From 2.0V to 0.8V

A.C. CHARACTERISTICS (Continued)

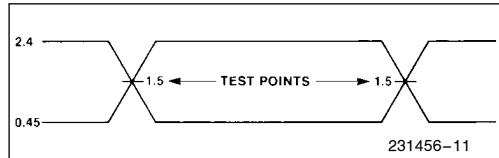
TIMING RESPONSES

Symbol	Parameter	8088		8088-2		Units	Test Conditions
		Min	Max	Min	Max		
TCLAV	Address Valid Delay	10	110	10	60	ns	
TCLAX	Address Hold Time	10		10		ns	
TCLAZ	Address Float Delay	TCLAX	80	TCLAX	50	ns	
TLHLL	ALE Width	TCLCH-20		TCLCH-10		ns	
TCLLH	ALE Active Delay		80		50	ns	
TCHLL	ALE Inactive Delay		85		55	ns	
TLLAX	Address Hold Time to ALE Inactive	TCHCL-10		TCHCL-10		ns	
TCLDV	Data Valid Delay	10	110	10	60	ns	
TCHDX	Data Hold Time	10		10		ns	
TWHDX	Data Hold Time after WR	TCLCH-30		TCLCH-30		ns	
TCVCTV	Control Active Delay 1	10	110	10	70	ns	
TCHCTV	Control Active Delay 2	10	110	10	60	ns	
TCVCTX	Control Inactive Delay	10	110	10	70	ns	
TAZRL	Address Float to READ Active	0		0		ns	
TCLRL	RD Active Delay	10	165	10	100	ns	
TCLRH	RD Inactive Delay	10	150	10	80	ns	
TRHAV	RD Inactive to Next Address Active	TCLCL-45		TCLCL-40		ns	
TCLHAV	HLDA Valid Delay	10	160	10	100	ns	
TRLRH	RD Width	2TCLCL-75		2TCLCL-50		ns	
TWLWH	WR Width	2TCLCL-60		2TCLCL-40		ns	
TAVAL	Address Valid to ALE Low	TCLCH-60		TCLCH-40		ns	
TOLOH	Output Rise Time		20		20	ns	From 0.8V to 2.0V
TOHOL	Output Fall Time		12		12	ns	From 2.0V to 0.8V

NOTES:

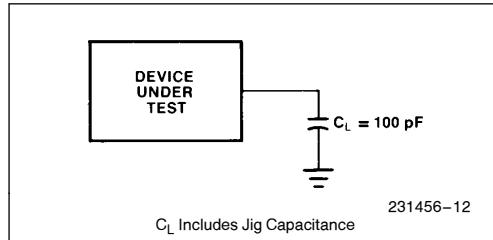
1. Signal at 8284A shown for reference only. See 8284A data sheet for the most recent specifications.
2. Set up requirement for asynchronous signal only to guarantee recognition at next CLK.
3. Applies only to T2 state (8 ns into T3 state).

A.C. TESTING INPUT, OUTPUT WAVEFORM



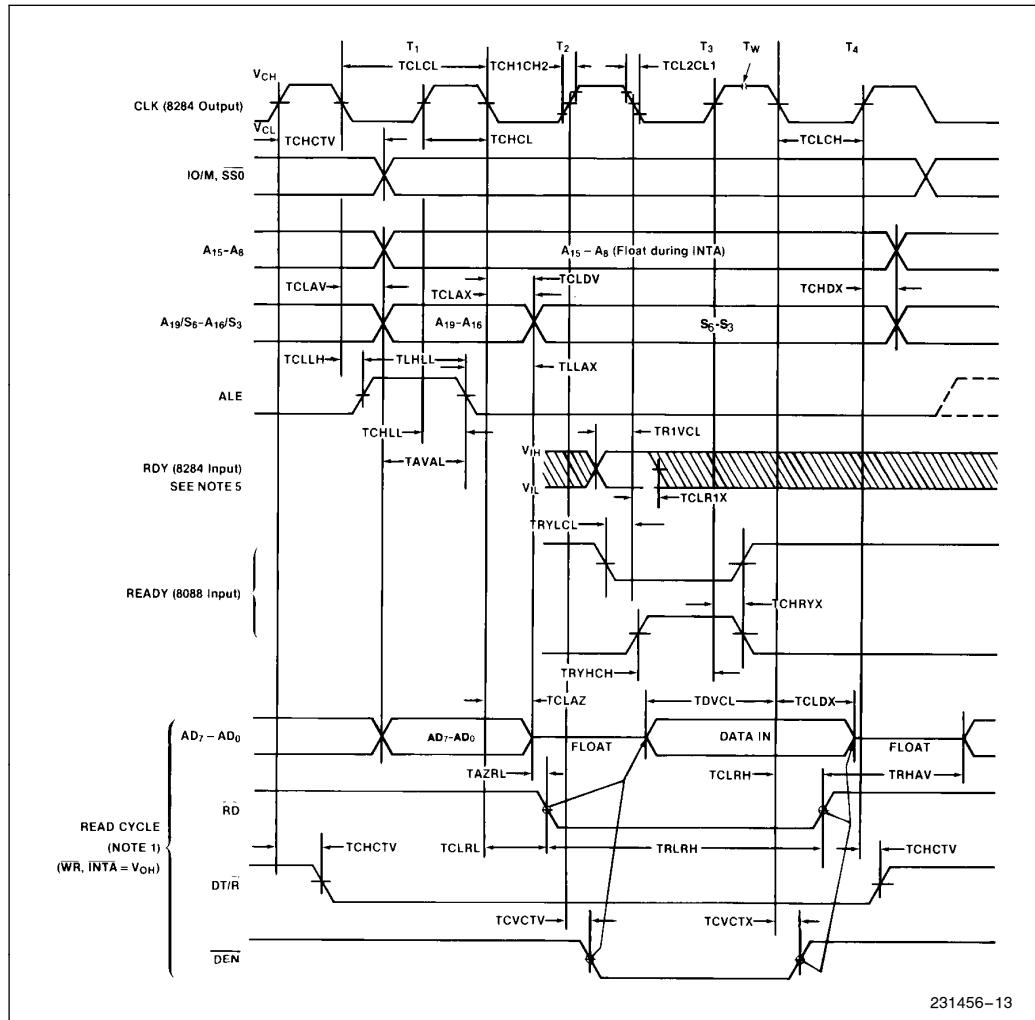
A.C. Testing; Inputs are driven at 2.4V for a logic “1” and 0.45V for a logic “0”. Timing measurements are made at 1.5V for both a logic “1” and logic “0”.

A.C. TESTING LOAD CIRCUIT



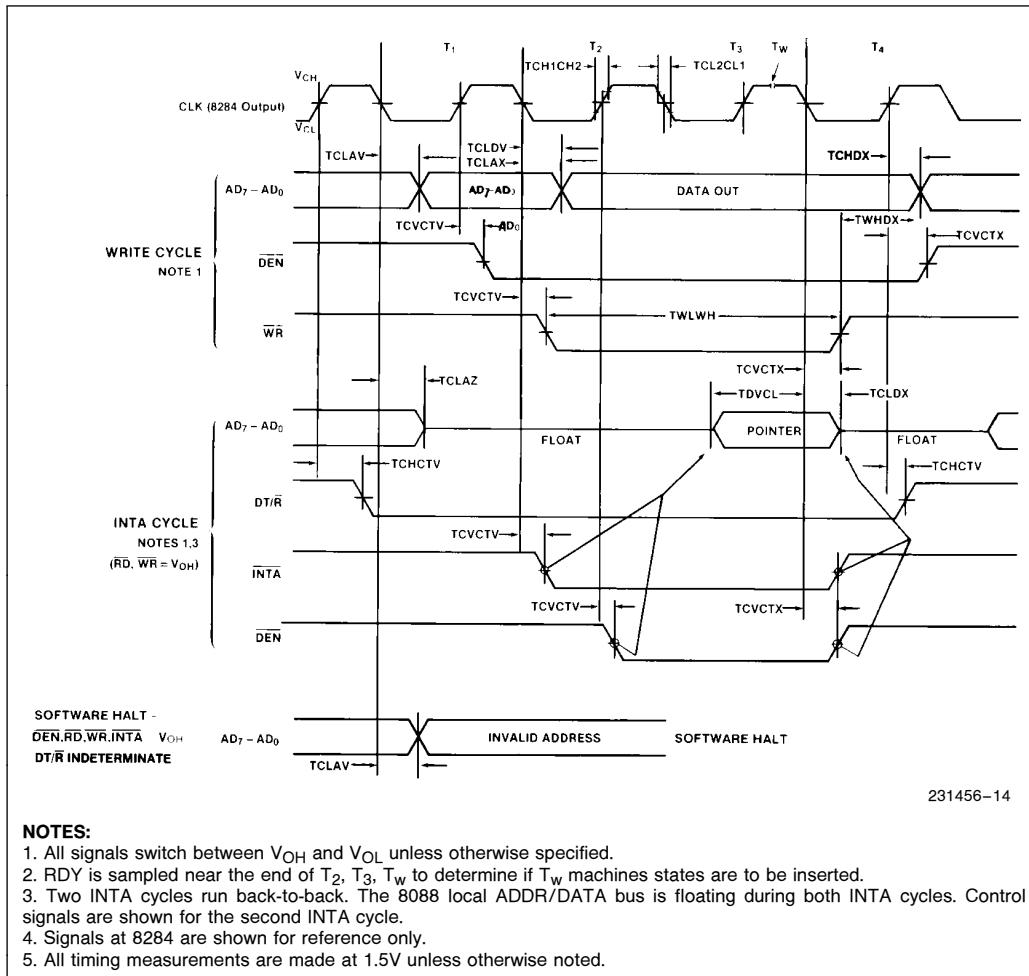
WAVEFORMS

BUS TIMING—MINIMUM MODE SYSTEM



WAVEFORMS (Continued)

BUS TIMING—MINIMUM MODE SYSTEM (Continued)



A.C. CHARACTERISTICS

MAX MODE SYSTEM (USING 8288 BUS CONTROLLER)

TIMING REQUIREMENTS

Symbol	Parameter	8088		8088-2		Units	Test Conditions
		Min	Max	Min	Max		
TCLCL	CLK Cycle Period	200	500	125	500	ns	
TCLCH	CLK Low Time	118		68		ns	
TCHCL	CLK High Time	69		44		ns	
TCH1CH2	CLK Rise Time		10		10	ns	From 1.0V to 3.5V
TCL2CL1	CLK Fall Time		10		10	ns	From 3.5V to 1.0V
TDVCL	Data in Setup Time	30		20		ns	
TCLDX	Data in Hold Time	10		10		ns	
TR1VCL	RDY Setup Time into 8284 (Notes 1, 2)	35		35		ns	
TCLR1X	RDY Hold Time into 8284 (Notes 1, 2)	0		0		ns	
TRYHCH	READY Setup Time into 8088	118		68		ns	
TCHRYX	READY Hold Time into 8088	30		20		ns	
TRYLCL	READY Inactive to CLK (Note 4)	-8		-8		ns	
TINVCH	Setup Time for Recognition (INTR, NMI, TEST) (Note 2)	30		15		ns	
TGVCH	RQ/GT Setup Time	30		15		ns	
TCHGX	RQ Hold Time into 8088	40		30		ns	
TILIH	Input Rise Time (Except CLK)		20		20	ns	From 0.8V to 2.0V
TIHIL	Input Fall Time (Except CLK)		12		12	ns	From 2.0V to 0.8V

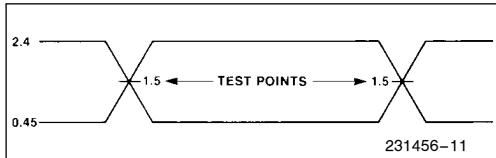
A.C. CHARACTERISTICS (Continued)

TIMING RESPONSES

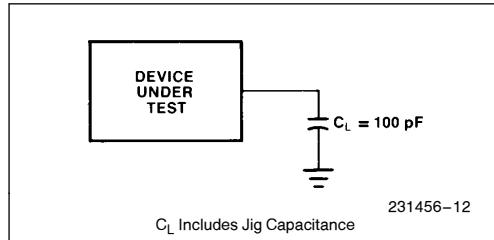
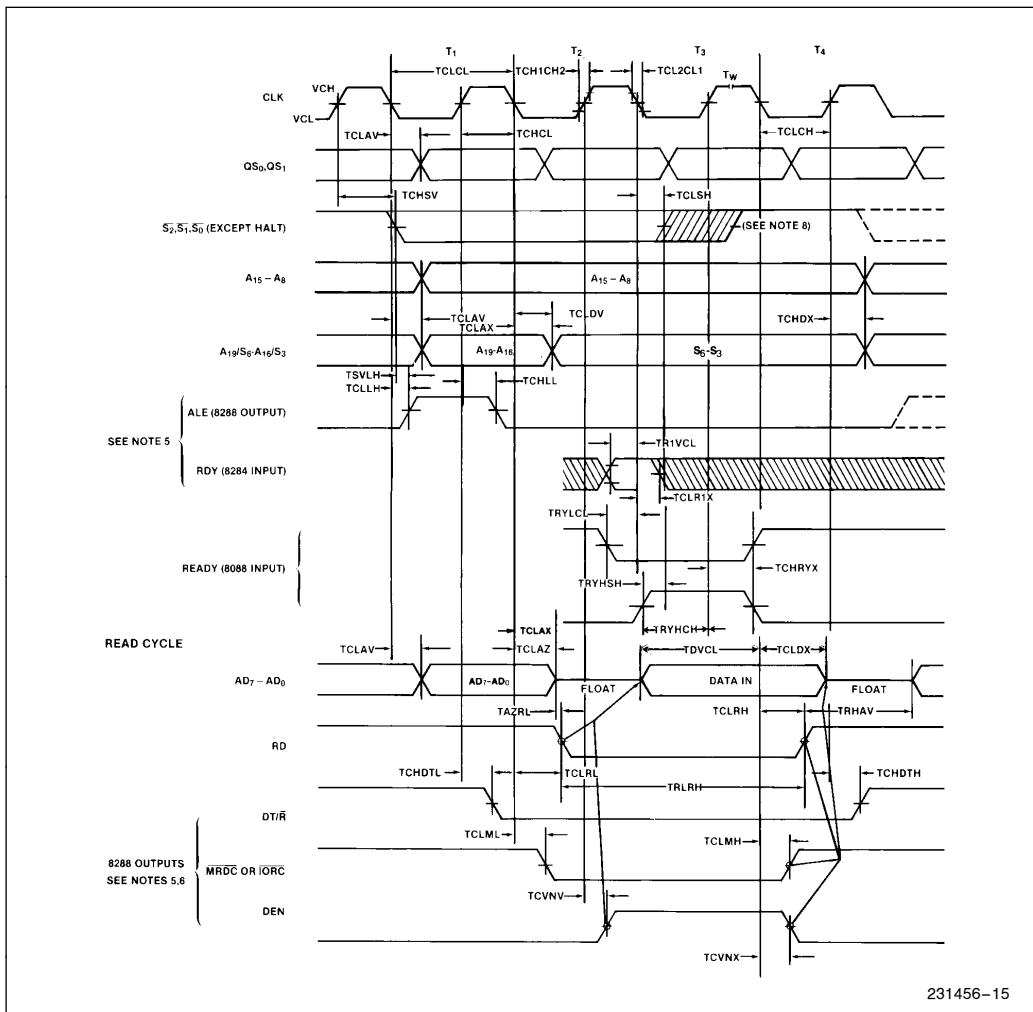
Symbol	Parameter	8088		8088-2		Units	Test Conditions
		Min	Max	Min	Max		
TCLML	Command Active Delay (Note 1)	10	35	10	35	ns	$C_L = 20\text{--}100 \text{ pF}$ for All 8088 Outputs in Addition to Internal Loads
TCLMH	Command Inactive Delay (Note 1)	10	35	10	35	ns	
TRYHSH	READY Active to Status Passive (Note 3)		110		65	ns	
TCHSV	Status Active Delay	10	110	10	60	ns	
TCLSH	Status Inactive Delay	10	130	10	70	ns	
TCLAV	Address Valid Delay	10	110	10	60	ns	
TCLAX	Address Hold Time	10		10		ns	
TCLAZ	Address Float Delay	TCLAX	80	TCLAX	50	ns	
TSVLH	Status Valid to ALE High (Note 1)		15		15	ns	
TSVMCH	Status Valid to MCE High (Note 1)		15		15	ns	
TCLLH	CLK Low to ALE Valid (Note 1)		15		15	ns	
TCLMCH	CLK Low to MCE (Note 1)		15		15	ns	
TCHLL	ALE Inactive Delay (Note 1)		15		15	ns	
TCLMCL	MCE Inactive Delay (Note 1)		15		15	ns	
TCLDV	Data Valid Delay	10	110	10	60	ns	
TCHDX	Data Hold Time	10		10		ns	
TCVNV	Control Active Delay (Note 1)	5	45	5	45	ns	
TCVNX	Control Inactive Delay (Note 1)	10	45	10	45	ns	
TAZRL	Address Float to Read Active	0		0		ns	
TCLRL	RD Active Delay	10	165	10	100	ns	
TCLRH	RD Inactive Delay	10	150	10	80	ns	
TRHAV	RD Inactive to Next Address Active	TCLCL-45		TCLCL-40		ns	
TCHDTL	Direction Control Active Delay (Note 1)		50		50	ns	
TCHDTH	Direction Control Inactive Delay (Note 1)		30		30	ns	
TCLGL	GT Active Delay		85		50	ns	
TCLGH	GT Inactive Delay		85		50	ns	
TRLRH	RD Width	2TCLCL-75		2TCLCL-50		ns	
TOLOH	Output Rise Time		20		20	ns	From 0.8V to 2.0V
TOHOL	Output Fall Time		12		12	ns	From 2.0V to 0.8V

NOTES:

1. Signal at 8284 or 8288 shown for reference only.
2. Setup requirement for asynchronous signal only to guarantee recognition at next CLK.
3. Applies only to T3 and wait states.
4. Applies only to T2 state (8 ns into T3 state).

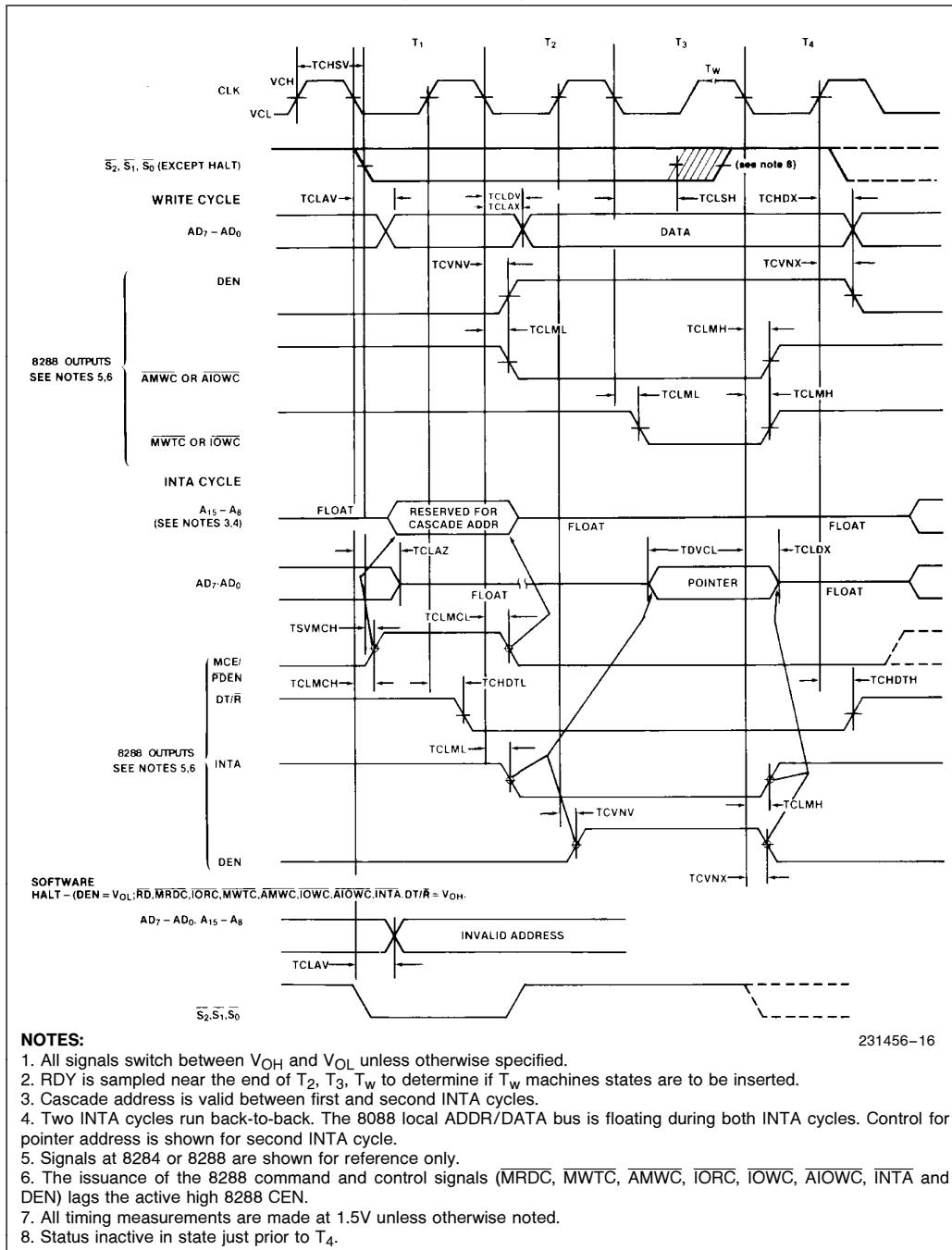
A.C. TESTING INPUT, OUTPUT WAVEFORM


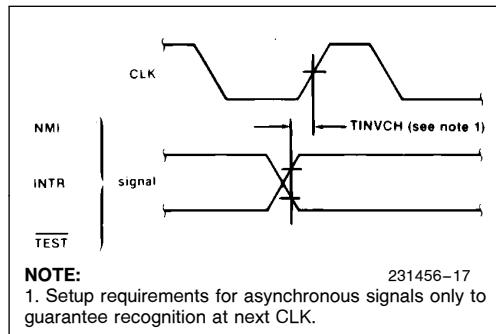
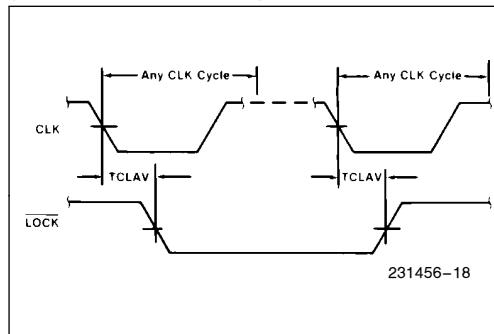
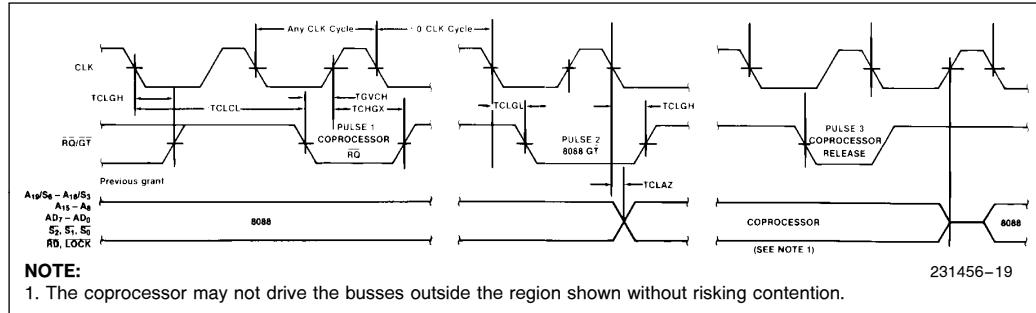
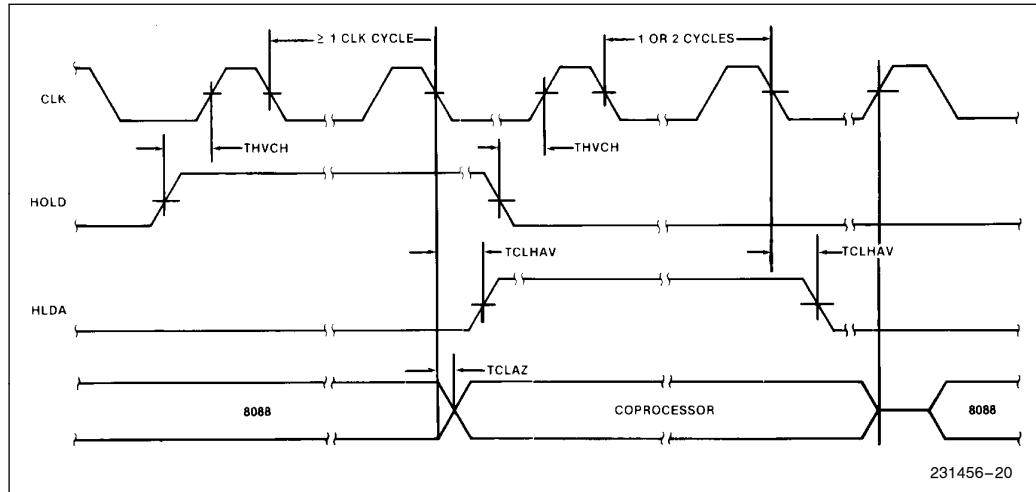
A.C. Testing; Inputs are driven at 2.4V for a logic "1" and 0.45V for a logic "0". Timing measurements are made at 1.5V for both a logic "1" and logic "0".

A.C. TESTING LOAD CIRCUIT

WAVEFORMS (Continued)
BUS TIMING—MAXIMUM MODE SYSTEM


WAVEFORMS (Continued)

BUS TIMING—MAXIMUM MODE SYSTEM (USING 8288)



WAVEFORMS (Continued)
ASYNCHRONOUS SIGNAL RECOGNITION

**BUS LOCK SIGNAL TIMING
(MAXIMUM MODE ONLY)**

REQUEST/GRANT SEQUENCE TIMING (MAXIMUM MODE ONLY)

HOLD/HOLD ACKNOWLEDGE TIMING (MINIMUM MODE ONLY)


8086/8088 Instruction Set Summary

Mnemonic and Description	Instruction Code			
DATA TRANSFER				
MOV = Move:	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0
Register/Memory to/from Register	1 0 0 0 1 0 d w	mod reg r/m		
Immediate to Register/Memory	1 1 0 0 0 1 1 w	mod 0 0 0 r/m	data	data if w = 1
Immediate to Register	1 0 1 1 w reg	data	data if w = 1	
Memory to Accumulator	1 0 1 0 0 0 w	addr-low	addr-high	
Accumulator to Memory	1 0 1 0 0 1 w	addr-low	addr-high	
Register/Memory to Segment Register	1 0 0 0 1 1 0	mod 0 reg r/m		
Segment Register to Register/Memory	1 0 0 0 1 1 0 0	mod 0 reg r/m		
PUSH = Push:				
Register/Memory	1 1 1 1 1 1 1 1	mod 1 1 0 r/m		
Register	0 1 0 1 0 reg			
Segment Register	0 0 0 reg 1 1 0			
POP = Pop:				
Register/Memory	1 0 0 0 1 1 1 1	mod 0 0 0 r/m		
Register	0 1 0 1 1 reg			
Segment Register	0 0 0 reg 1 1 1			
XCHG = Exchange:				
Register/Memory with Register	1 0 0 0 0 1 1 w	mod reg r/m		
Register with Accumulator	1 0 0 1 0 reg			
IN = Input from:				
Fixed Port	1 1 1 0 0 1 0 w	port		
Variable Port	1 1 1 0 1 1 0 w			
OUT = Output to:				
Fixed Port	1 1 1 0 0 1 1 w	port		
Variable Port	1 1 1 0 1 1 1 w			
XLAT = Translate Byte to AL	1 1 0 1 0 1 1 1			
LEA = Load EA to Register	1 0 0 0 1 1 0 1	mod reg r/m		
LDS = Load Pointer to DS	1 1 0 0 0 1 0 1	mod reg r/m		
LES = Load Pointer to ES	1 1 0 0 0 1 0 0	mod reg r/m		
LAHF = Load AH with Flags	1 0 0 1 1 1 1 1			
SAHF = Store AH into Flags	1 0 0 1 1 1 1 0			
PUSHF = Push Flags	1 0 0 1 1 1 0 0			
POPF = Pop Flags	1 0 0 1 1 1 0 1			

8086/8088 Instruction Set Summary (Continued)

Mnemonic and Description	Instruction Code			
ARITHMETIC	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0
ADD = Add:				
Reg./Memory with Register to Either	0 0 0 0 0 0 d w	mod reg r/m		
Immediate to Register/Memory	1 0 0 0 0 0 s w	mod 0 0 0 r/m	data	data if s:w = 01
Immediate to Accumulator	0 0 0 0 0 1 0 w	data	data if w = 1	
ADC = Add with Carry:				
Reg./Memory with Register to Either	0 0 0 1 0 0 d w	mod reg r/m		
Immediate to Register/Memory	1 0 0 0 0 0 s w	mod 0 1 0 r/m	data	data if s:w = 01
Immediate to Accumulator	0 0 0 1 0 1 0 w	data	data if w = 1	
INC = Increment:				
Register/Memory	1 1 1 1 1 1 1 w	mod 0 0 0 r/m		
Register	0 1 0 0 0 reg			
AAA = ASCII Adjust for Add	0 0 1 1 0 1 1 1			
BAA = Decimal Adjust for Add	0 0 1 0 0 1 1 1			
SUB = Subtract:				
Reg./Memory and Register to Either	0 0 1 0 1 0 d w	mod reg r/m		
Immediate from Register/Memory	1 0 0 0 0 0 s w	mod 1 0 1 r/m	data	data if s:w = 01
Immediate from Accumulator	0 0 1 0 1 1 0 w	data	data if w = 1	
SSB = Subtract with Borrow				
Reg./Memory and Register to Either	0 0 0 1 1 0 d w	mod reg r/m		
Immediate from Register/Memory	1 0 0 0 0 0 s w	mod 0 1 1 r/m	data	data if s:w = 01
Immediate from Accumulator	0 0 0 1 1 1 w	data	data if w = 1	
DEC = Decrement:				
Register/memory	1 1 1 1 1 1 1 w	mod 0 0 1 r/m		
Register	0 1 0 0 1 reg			
NEG = Change sign	1 1 1 1 0 1 1 w	mod 0 1 1 r/m		
CMP = Compare:				
Register/Memory and Register	0 0 1 1 1 0 d w	mod reg r/m		
Immediate with Register/Memory	1 0 0 0 0 0 s w	mod 1 1 1 r/m	data	data if s:w = 01
Immediate with Accumulator	0 0 1 1 1 1 0 w	data	data if w = 1	
AAS = ASCII Adjust for Subtract	0 0 1 1 1 1 1 1			
DAS = Decimal Adjust for Subtract	0 0 1 0 1 1 1 1			
MUL = Multiply (Unsigned)	1 1 1 1 0 1 1 w	mod 1 0 0 r/m		
IMUL = Integer Multiply (Signed)	1 1 1 1 0 1 1 w	mod 1 0 1 r/m		
AAM = ASCII Adjust for Multiply	1 1 0 1 0 1 0 0	0 0 0 0 1 0 1 0		
DIV = Divide (Unsigned)	1 1 1 1 0 1 1 w	mod 1 1 0 r/m		
IDIV = Integer Divide (Signed)	1 1 1 1 0 1 1 w	mod 1 1 1 r/m		
AAD = ASCII Adjust for Divide	1 1 0 1 0 1 0 1	0 0 0 0 1 0 1 0		
CBW = Convert Byte to Word	1 0 0 1 1 0 0 0			
CWD = Convert Word to Double Word	1 0 0 1 1 0 0 1			

8086/8088 Instruction Set Summary (Continued)

Mnemonic and Description	Instruction Code			
LOGIC	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0
NOT = Invert	1 1 1 1 0 1 1 w	mod 0 1 0 r/m		
SHL/SAL = Shift Logical/Arithmetic Left	1 1 0 1 0 0 v w	mod 1 0 0 r/m		
SHR = Shift Logical Right	1 1 0 1 0 0 v w	mod 1 0 1 r/m		
SAR = Shift Arithmetic Right	1 1 0 1 0 0 v w	mod 1 1 1 r/m		
ROL = Rotate Left	1 1 0 1 0 0 v w	mod 0 0 0 r/m		
ROR = Rotate Right	1 1 0 1 0 0 v w	mod 0 0 1 r/m		
RCL = Rotate Through Carry Flag Left	1 1 0 1 0 0 v w	mod 0 1 0 r/m		
RCR = Rotate Through Carry Right	1 1 0 1 0 0 v w	mod 0 1 1 r/m		
AND = And:				
Reg./Memory and Register to Either	0 0 1 0 0 0 d w	mod reg r/m		
Immediate to Register/Memory	1 0 0 0 0 0 0 w	mod 1 0 0 r/m	data	data if w = 1
Immediate to Accumulator	0 0 1 0 0 1 0 w	data	data if w = 1	
TEST = And Function to Flags. No Result:				
Register/Memory and Register	1 0 0 0 0 1 0 w	mod reg r/m		
Immediate Data and Register/Memory	1 1 1 1 0 1 1 w	mod 0 0 0 r/m	data	data if w = 1
Immediate Data and Accumulator	1 0 1 0 1 0 0 w	data	data if w = 1	
OR = Or:				
Reg./Memory and Register to Either	0 0 0 0 1 0 d w	mod reg r/m		
Immediate to Register/Memory	1 0 0 0 0 0 0 w	mod 0 0 1 r/m	data	data if w = 1
Immediate to Accumulator	0 0 0 0 1 1 0 w	data	data if w = 1	
XOR = Exclusive or:				
Reg./Memory and Register to Either	0 0 1 1 0 0 d w	mod reg r/m		
Immediate to Register/Memory	1 0 0 0 0 0 0 w	mod 1 1 0 r/m	data	data if w = 1
Immediate to Accumulator	0 0 1 1 0 1 0 w	data	data if w = 1	
STRING MANIPULATION				
REP = Repeat	1 1 1 1 0 0 1 z			
MOVS = Move Byte/Word	1 0 1 0 0 1 0 w			
CMPS = Compare Byte/Word	1 0 1 0 0 1 1 w			
SCAS = Scan Byte/Word	1 0 1 0 1 1 1 w			
LODS = Load Byte/Wd to AL/AX	1 0 1 0 1 1 0 w			
STOS = Stor Byte/Wd from AL/A	1 0 1 0 1 0 1 w			
CONTROL TRANSFER				
CALL = Call:				
Direct Within Segment	1 1 1 0 1 0 0 0	disp-low	disp-high	
Indirect Within Segment	1 1 1 1 1 1 1 1	mod 0 1 0 r/m		
Direct Intersegment	1 0 0 1 1 0 1 0	offset-low	offset-high	
		seg-low	seg-high	
Indirect Intersegment	1 1 1 1 1 1 1 1	mod 0 1 1 r/m		

8086/8088 Instruction Set Summary (Continued)

Mnemonic and Description	Instruction Code		
JMP = Unconditional Jump:	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0
Direct Within Segment	1 1 1 0 1 0 0 1	disp-low	disp-high
Direct Within Segment-Short	1 1 1 0 1 0 1 1	disp	
Indirect Within Segment	1 1 1 1 1 1 1 1	mod 1 0 0 r/m	
Direct Intersegment	1 1 1 0 1 0 1 0	offset-low	offset-high
		seg-low	seg-high
Indirect Intersegment	1 1 1 1 1 1 1 1	mod 1 0 1 r/m	
RET = Return from CALL:			
Within Segment	1 1 0 0 0 0 1 1		
Within Seg Adding Immed to SP	1 1 0 0 0 0 1 0	data-low	data-high
Intersegment	1 1 0 0 1 0 1 1		
Intersegment Adding Immediate to SP	1 1 0 0 1 0 1 0	data-low	data-high
JE/JZ = Jump on Equal/Zero	0 1 1 1 0 1 0 0	disp	
JL/JNGE = Jump on Less/Not Greater or Equal	0 1 1 1 1 1 0 0	disp	
JLE/JNG = Jump on Less or Equal/Not Greater	0 1 1 1 1 1 1 0	disp	
JB/JNAE = Jump on Below/Not Above or Equal	0 1 1 1 0 0 1 0	disp	
JBE/JNA = Jump on Below or Equal/Not Above	0 1 1 1 0 1 1 0	disp	
JP/JPE = Jump on Parity/Parity Even	0 1 1 1 1 0 1 0	disp	
JO = Jump on Overflow	0 1 1 1 0 0 0 0	disp	
JS = Jump on Sign	0 1 1 1 1 0 0 0	disp	
JNE/JNZ = Jump on Not Equal/Not Zero	0 1 1 1 0 1 0 1	disp	
JNL/JGE = Jump on Not Less/Greater or Equal	0 1 1 1 1 1 0 1	disp	
JNLE/JG = Jump on Not Less or Equal/Greater	0 1 1 1 1 1 1 1	disp	
JNB/JAE = Jump on Not Below/Above or Equal	0 1 1 1 0 0 1 1	disp	
JNBE/JA = Jump on Not Below or Equal/Above	0 1 1 1 0 1 1 1	disp	
JNP/JPO = Jump on Not Par/Par Odd	0 1 1 1 1 0 1 1	disp	
JNO = Jump on Not Overflow	0 1 1 1 0 0 0 1	disp	
JNS = Jump on Not Sign	0 1 1 1 1 0 0 1	disp	
LOOP = Loop CX Times	1 1 1 0 0 0 1 0	disp	
LOOPZ/LOOPE = Loop While Zero/Equal	1 1 1 0 0 0 0 1	disp	
LOOPNZ/LOOPNE = Loop While Not Zero/Equal	1 1 1 0 0 0 0 0	disp	
JCXZ = Jump on CX Zero	1 1 1 0 0 0 1 1	disp	
INT = Interrupt			
Type Specified	1 1 0 0 1 1 0 1	type	
Type 3	1 1 0 0 1 1 0 0		
INTO = Interrupt on Overflow	1 1 0 0 1 1 1 0		
IRET = Interrupt Return	1 1 0 0 1 1 1 1		

8086/8088 Instruction Set Summary (Continued)

Mnemonic and Description	Instruction Code	
	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0
PROCESSOR CONTROL		
CLC = Clear Carry	1 1 1 1 1 0 0 0	
CMC = Complement Carry	1 1 1 1 0 1 0 1	
STC = Set Carry	1 1 1 1 1 0 0 1	
CLD = Clear Direction	1 1 1 1 1 1 0 0	
STD = Set Direction	1 1 1 1 1 1 0 1	
CLI = Clear Interrupt	1 1 1 1 1 0 1 0	
STI = Set Interrupt	1 1 1 1 1 0 1 1	
HLT = Halt	1 1 1 1 0 1 0 0	
WAIT = Wait	1 0 0 1 1 0 1 1	
ESC = Escape (to External Device)	1 1 0 1 1 x x x	mod x x x r/m
LOCK = Bus Lock Prefix	1 1 1 1 0 0 0 0	

NOTES:

AL = 8-bit accumulator
 AX = 16-bit accumulator
 CX = Count register
 DS = Data segment
 ES = Extra segment
 Above/below refers to unsigned value
 Greater = more positive:
 Less = less positive (more negative) signed values
 if d = 1 then "to" reg; if d = 0 then "from" reg
 if w = 1 then word instruction; if w = 0 then byte instruction
 if mod = 11 then r/m is treated as a REG field
 if mod = 00 then DISP = 0*, disp-low and disp-high are absent
 if mod = 01 then DISP = disp-low sign-extended to 16 bits, disp-high is absent
 if mod = 10 then DISP = disp-high; disp-low
 if r/m = 000 then EA = (BX) + (SI) + DISP
 if r/m = 001 then EA = (BX) + (DI) + DISP
 if r/m = 010 then EA = (BP) + (SI) + DISP
 if r/m = 011 then EA = (BP) + (DI) + DISP
 if r/m = 100 then EA = (SI) + DISP
 if r/m = 101 then EA = (DI) + DISP
 if r/m = 110 then EA = (BP) + DISP*
 if r/m = 111 then EA = (BX) + DISP
 DISP follows 2nd byte of instruction (before data if required)
 *except if mod = 00 and r/m = then EA = disp-high: disp-low.
 if s:w = 01 then 16 bits of immediate data form the operand
 if s:w = 11 then an immediate data byte is sign extended to form the 16-bit operand
 if v = 0 then "count" = 1; if v = 1 then "count" in (CL) register
 x = don't care
 z is used for string primitives for comparison with ZF FLAG
 SEGMENT OVERRIDE PREFIX

0 0 1 reg 1 1 0

REG is assigned according to the following table:

16-Bit (w = 1)	8-Bit (w = 0)	Segment
000 AX	000 AL	00 ES
001 CX	001 CL	01 CS
010 DX	010 DL	10 SS
011 BX	011 BL	11 DS
100 SP	100 AH	
101 BP	101 CH	
110 SI	110 DH	
111 DI	111 BH	

Instructions which reference the flag register file as a 16-bit object use the symbol FLAGS to represent the file:

FLAGS =

X:X:X:(OF):(DF):(IF):(TF):(SF):(ZF):X:(AF):X:(PF):X:(CF)

Mnemonics © Intel, 1978

DATA SHEET REVISION REVIEW

The following list represents key differences between this and the -005 data sheet. Please review this summary carefully.

1. The Intel 8088 implementation technology (HMOS) has been changed to (HMOS-II).

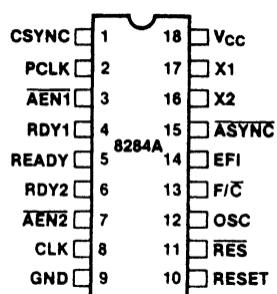
Generador de Reloj 8284A

Arquitectura de Computadoras

Generador de Reloj 8284A

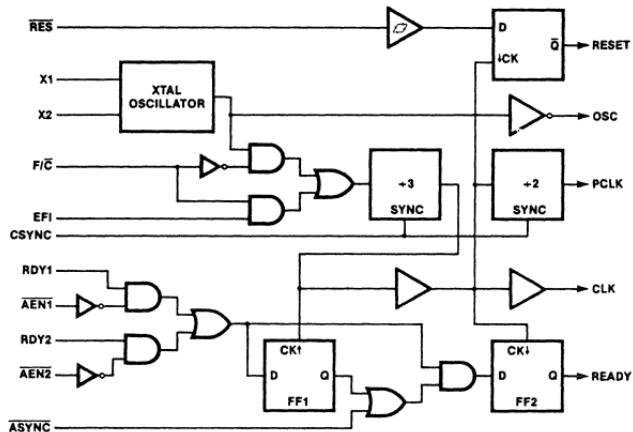
Generador de Reloj 8284A

- Terminales



Generador de Reloj 8284A

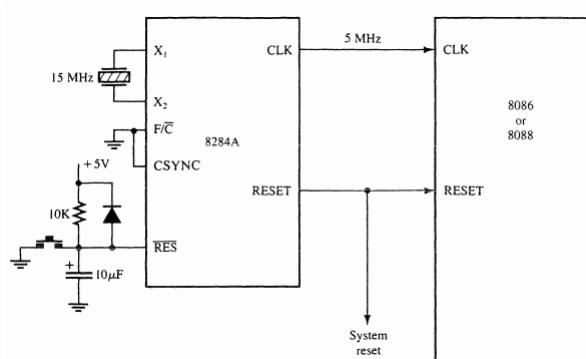
- Diagrama de bloques interno



3

Generador de Reloj 8284A

- Conexión con el microprocesador 8086



4

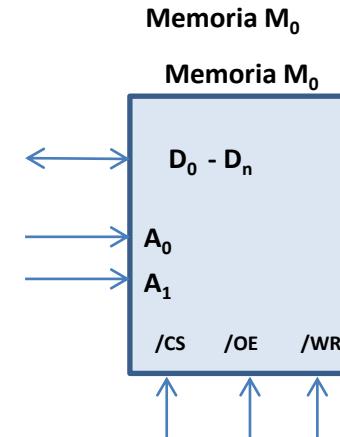
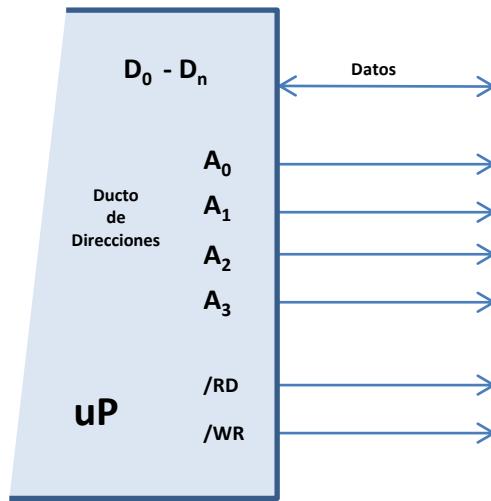
Decodificadores

Decodificadores de Memoria

Con fines de explicación considere lo siguiente:

- Un procesador (uP) con un ducto de datos de n bits y un ducto direcciones de 4 bits ($A_3 - A_0$)
- Una memoria (M_0) de un ducto de datos de n bits y un ducto direcciones de 2 bits (A_1, A_0)

Nota: para este caso explicativo no importa el número de bits del ducto de datos.



1

El uP tiene un espacio de direcciones de 2^4 (16 localidades) pues su ducto de direcciones es de 4 bits

2

La memoria tiene un espacio de direcciones de 2^2 (4 localidades) pues su ducto de direcciones es de 2 bits

3

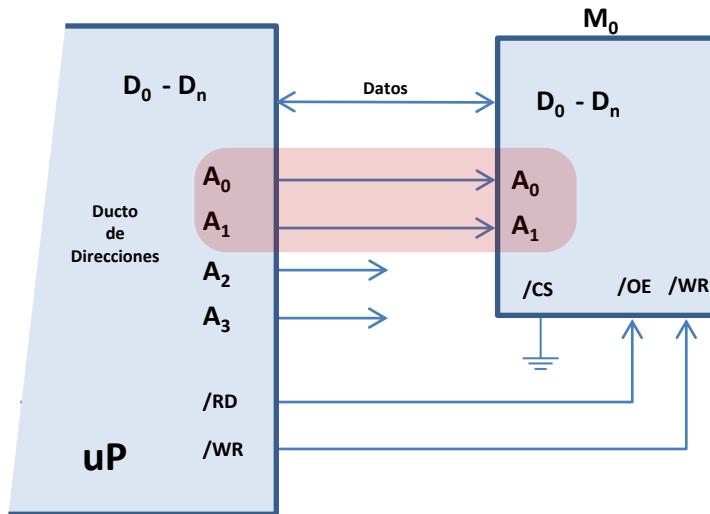
Si se desea conectar la memoria al procesador puede verse que el procesador tiene más líneas de direcciones que la memoria; por tanto tiene más localidades diferentes para acceder (16 en total) que las localidades que tiene la memoria (solamente 4).

Decodificadores de Memoria

4

Para conectar la memoria al procesador lo lógico es conectar los correspondientes ductos (direcciones, datos y control) con se muestra en la siguiente figura.

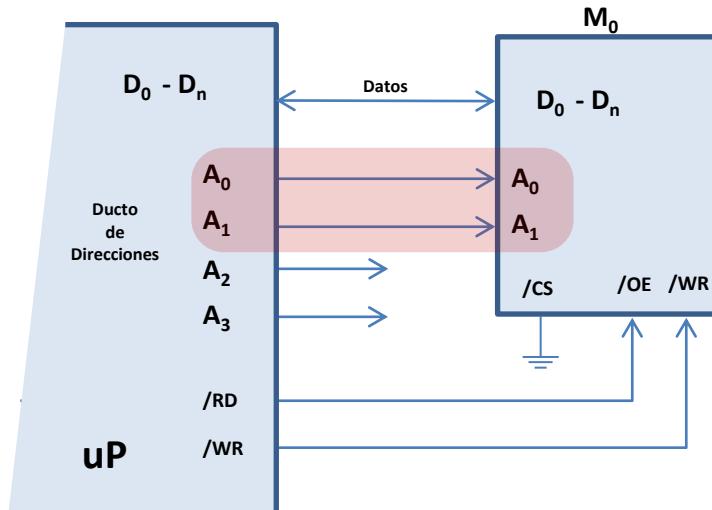
Nota: dado que se tiene una única memoria podemos tener el selector de la memoria (/CS) siempre activo (conectado a tierra)



Decodificadores de Memoria

5

Bajo estas condiciones tenemos entonces la siguiente tabla de direcciones del uP con su correspondiente dirección de la localidad accedida de la memoria.

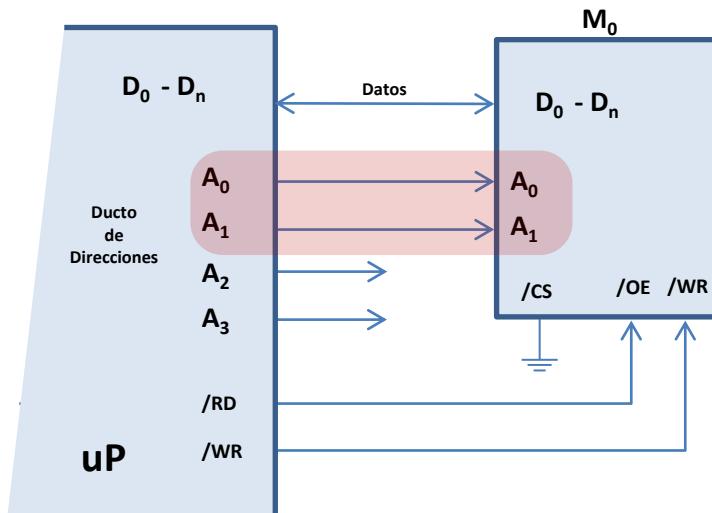


Dirección (uP)	Dir. Mem (Hex)	Localidad (Hex)
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	0	4
0101	1	5
0110	2	6
0111	3	7
1000	0	8
1001	1	9
1010	2	A
1011	3	B
1100	0	C
1101	1	D
1110	2	E
1111	3	F

Decodificadores de Memoria

6

Como puede observarse existen diferentes direcciones del uP para las cuales se acceda a la misma localidad de memoria y eso podemos verlo en la tabla donde se muestra que el espacio de memoria (de 4 localidades) se repite a lo largo del espacio de direcciones del uP.

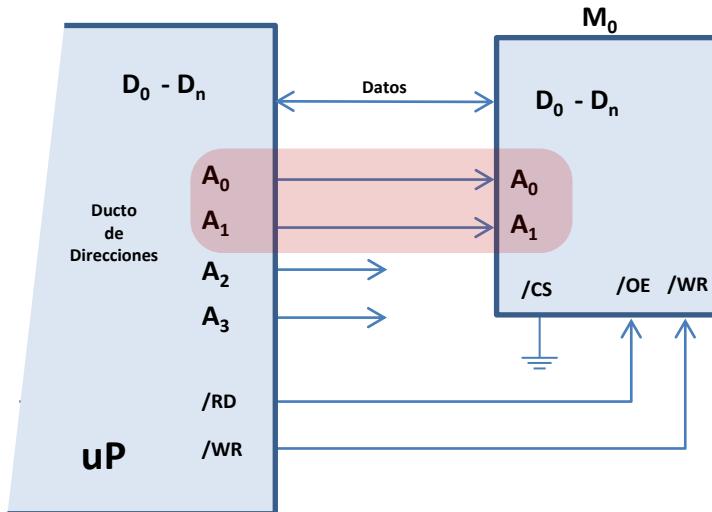


Dirección (uP)	Dir. Mem (Hex)	Localidad (Hex)
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	0	4
0101	1	5
0110	2	6
0111	3	7
1000	0	8
1001	1	9
1010	2	A
1011	3	B
1100	0	C
1101	1	D
1110	2	E
1111	3	F

Decodificadores de Memoria

7

Esto puede verse que es debido a que la memoria sólo puede detectar (digamos ver) los dos primeros bits (A_1 y A_0) del ducto de direcciones del uP causando entonces que existan **direcciones espejo**. Es decir una dirección de memoria tiene una correspondiente dirección del uP pero también puede verse reflejada en otra dirección diferente del uP.



Dirección (uP)	Dir. Mem (Hex)	Localidad (Hex)
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	0	4
0101	1	5
0110	2	6
0111	3	7
1000	0	8
1001	1	9
1010	2	A
1011	3	B
1100	0	C
1101	1	D
1110	2	E
1111	3	F

8

¿Es problema o no tener direcciones espejo? Depende...

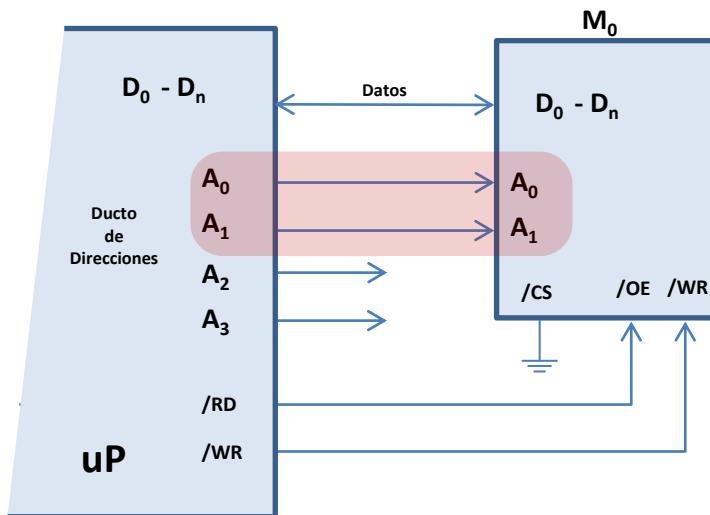
- Si el programador lo sabe entonces no existe problema alguno pues él deberá tener el suficiente cuidado de usar las direcciones correctas.
- Si el programador desconoce la existencia de zona espejos puede pensar que tiene más memoria de la que realmente existe y sobre escribir datos causando así problemas.

Decodificadores de Memoria

9

¿Cómo evitar que existan estas zonas espejo de la memoria?

Se tendría que forzar la asignación de un rango exclusivo de direcciones del uP a la memoria. Como ejemplo podemos decir que la memoria M_0 debería ser accedida exclusivamente en el rango de 0 a 3 de direcciones del uP.



Dirección (uP)	Dir. Mem (Hex)	Localidad (Hex)
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	0	4
0101	1	5
0110	2	6
0111	3	7
1000	0	8
1001	1	9
1010	2	A
1011	3	B
1100	0	C
1101	1	D
1110	2	E
1111	3	F

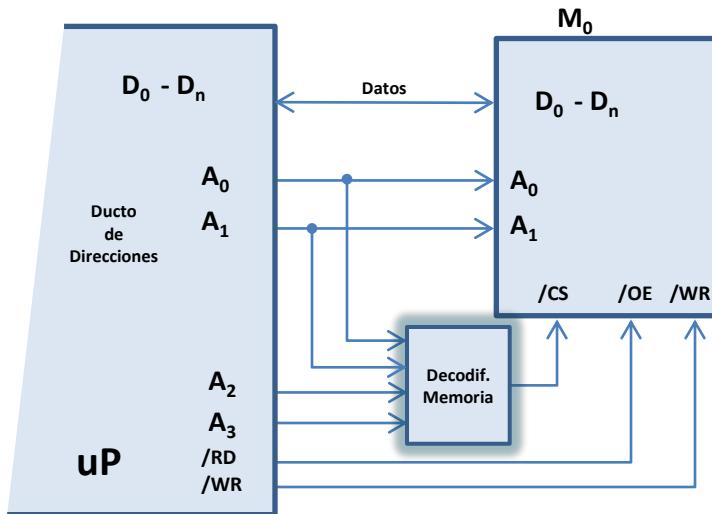
10

Para esto entonces es necesario un decodificador de memoria para que M_0 sólo se active en el rango de direcciones correspondientes.

Decodificadores de Memoria

11

Para esto entonces es necesario un decodificador de memoria para que M_0 sólo se active en el rango de direcciones correspondientes.



Dirección (uP)	Dir. Mem (Hex)	Localidad (Hex)	M ₀ /CS
0000	0	0	0
0001	1	1	0
0010	2	2	0
0011	3	3	0
0100	0	4	1
0101	1	5	1
0110	2	6	1
0111	3	7	1
1000	0	8	1
1001	1	9	1
1010	2	A	1
1011	3	B	1
1100	0	C	1
1101	1	D	1
1110	2	E	1
1111	3	F	1

Activar memoria

Desactivar memoria

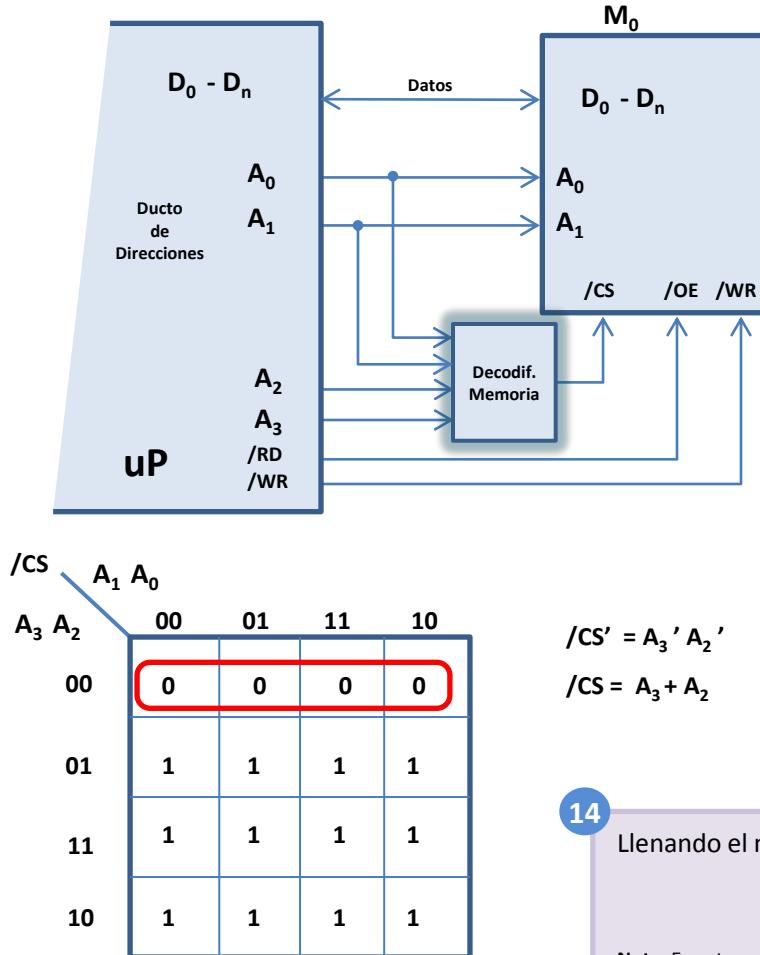
12

Por tanto el decodificador de memoria tendría como entradas las líneas de ducto de direcciones (A₃-A₀) y como salidas el selector de memoria (/CS). Así el decodificador determinará si la dirección presente corresponde al rango y entonces activar (hacer cero) la salida /CS; o desactivar (hacer uno) en cualquier otro caso en la que no corresponde al rango – ver tabla.

Decodificadores de Memoria

13

El decodificador es un circuito combinaciones donde su salida depende de la entrada (para al entrada, tal salida) y para el diseño se hace uso de mapas de Karnaugh.



Activar memoria

Dirección (uP)	Dir. Mem (Hex)	Localidad (Hex)	M_0 /CS
0000	0	0	0
0001	1	1	0
0010	2	2	0
0011	3	3	0
0100	0	4	1
0101	1	5	1
0110	2	6	1
0111	3	7	1
1000	0	8	1
1001	1	9	1
1010	2	A	1
1011	3	B	1
1100	0	C	1
1101	1	D	1
1110	2	E	1
1111	3	F	1

Desactivar memoria

14

Llenando el mapa y tomando ceros (son menos) tenemos como resultado:

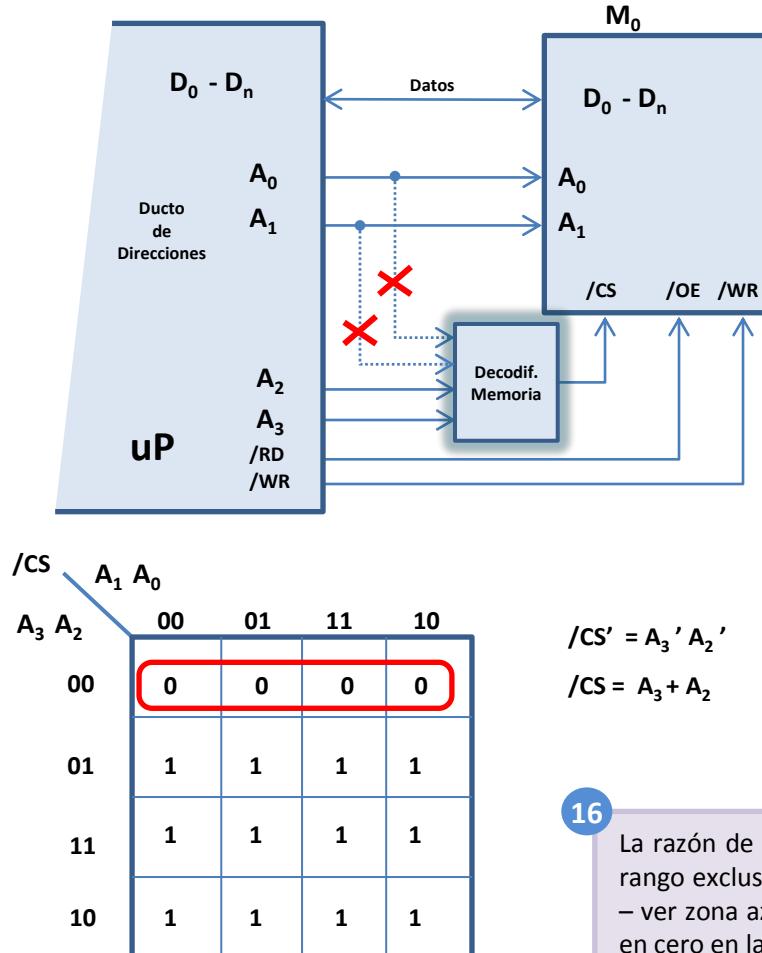
$$/CS = A_3 + A_2$$

Nota: En este caso particular resultaría la misma ecuación si tomamos los unos en lugar de los ceros.

Decodificadores de Memoria

15

Se puede observar que en el proceso desaparecen las líneas A_0 y A_1 – líneas que van directamente a la memoria.



16

La razón de que A_0 y A_1 se eliminan es porque esas líneas no nos ayudan a discriminar el rango exclusivo deseado pues en todo el espacio de direcciones del uP siempre cambiaron – ver zona azul. En cambio A_3 y A_2 si ayudan a discriminar la zona pues estás permanecen en cero en la zona exclusiva que se desea activar la memoria.

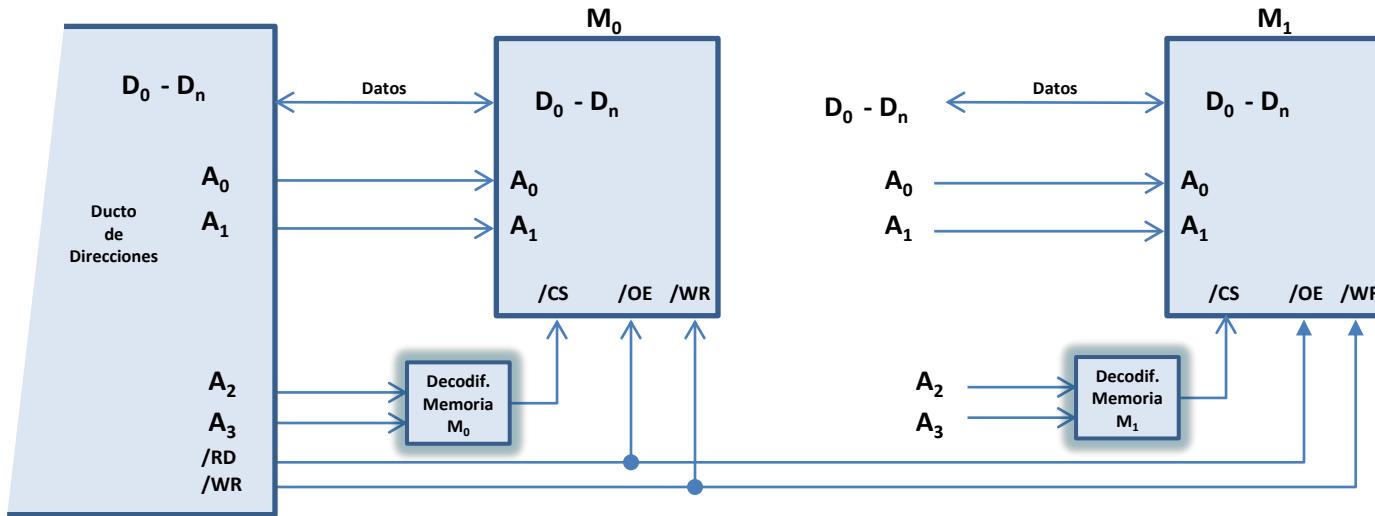
Activar memoria

Desactivar memoria

Decodificadores de Memoria

17

¿Qué sucede si ahora tenemos dos memorias?



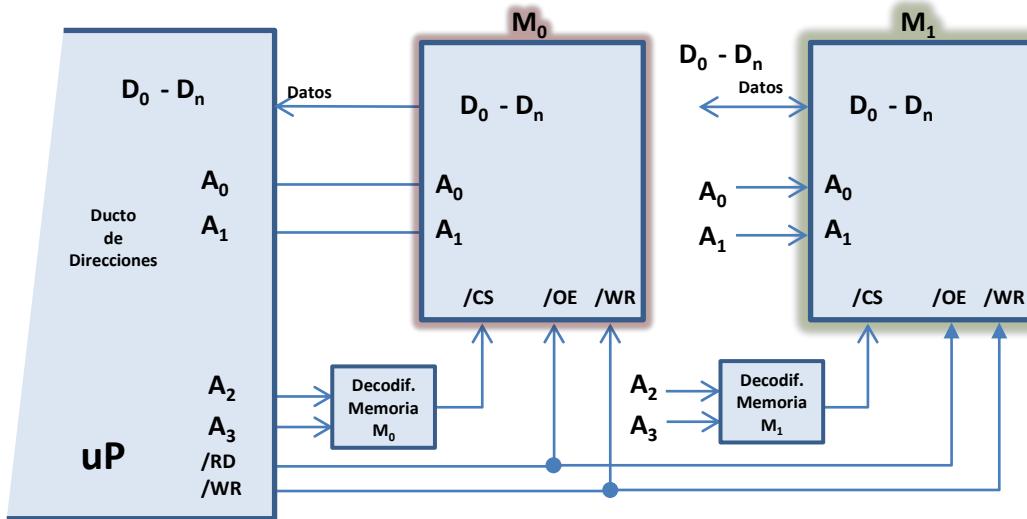
18

Para cada memoria debe existir un decodificador para que cada una de las memorias de active exclusivamente en su rango de direcciones válido.

Decodificadores de Memoria

19

Un ejemplo de esto sería el siguiente esquema de dos memoria de 4 localidades cada una.



Dirección (uP)	Dir. Mem 0 (Hex)	M_0 /CS	Dir. Mem 1 (Hex)	M_1 /CS
0000	0	0	-	1
0001	1	0	-	1
0010	2	0	-	1
0011	3	0	-	1
0100	-	1	0	0
0101	-	1	1	0
0110	-	1	2	0
0111	-	1	3	0
1000	-	1	-	1
1001	-	1	-	1
1010	-	1	-	1
1011	-	1	-	1
1100	-	1	-	1
1101	-	1	-	1
1110	-	1	-	1
1111	-	1	-	1

Activar M_0

Activar M_1

20

Cada memoria tiene su decodificador y para su diseño para cada uno se realiza su propio proceso según el rango a activar

EJERCICIO EJEMPLO DECODIFICADORES

Ejemplo Decodificadores

Conectar los elementos básicos de un sistema mínimo basado en 8088 con 32Kb de RAM y 32Kb de EPROM.

- Diseñe el decodificador de direcciones, y control para memoria. **Nota:** No existen direcciones espejo
 - a) RAM inicia en la dirección 00000h, EPROM inicia en 80000H
 - b) Mostrar las interconexiones entre los dispositivos. (8088, latch,ducto de direcciones, datos y control etc.)



Rango de la RAM, incluyendo a IO/_M para asegurar que el acceso que esta haciendo el procesador es a memoria.

IO/_M	A ₁₉	A ₁₈	A ₁₇	A ₁₆	A ₁₅	A ₁₄	A ₁₃	A ₁₂	A ₁₁	A ₁₀	A ₉	A ₈	A ₇	A ₆	A ₅	A ₄	A ₃	A ₂	A ₁	A ₀	HEX	_CS _{RAM}	_CS _{EPROM}	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	00000H	0	1
:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:
0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	07FFFFH	0	1
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	80000H	1	0
:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	:	
0	1	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	87FFFFH	1	0

Estas líneas permanecen constantes en el rango de la memoria EPROM

Estas líneas permanecen constantes en el rango de la memoria RAM

Estas líneas cambian el todo el rango, por lo que no ayudan a determinar si se está en el rango de la memoria uno o dos por tanto no intervienen en las ecuaciones de los selectores (SE CONECTAN DIRECTOS A LA MEMORIA)

De la tabla anterior _CS_{RAM} debe ser cero (0) cuando IO/_M=0 y A₁₉=0 y A₁₈=0 y A₁₇=0 y A₁₆=0 y A₁₅=0.

En forma de ecuación esto es:

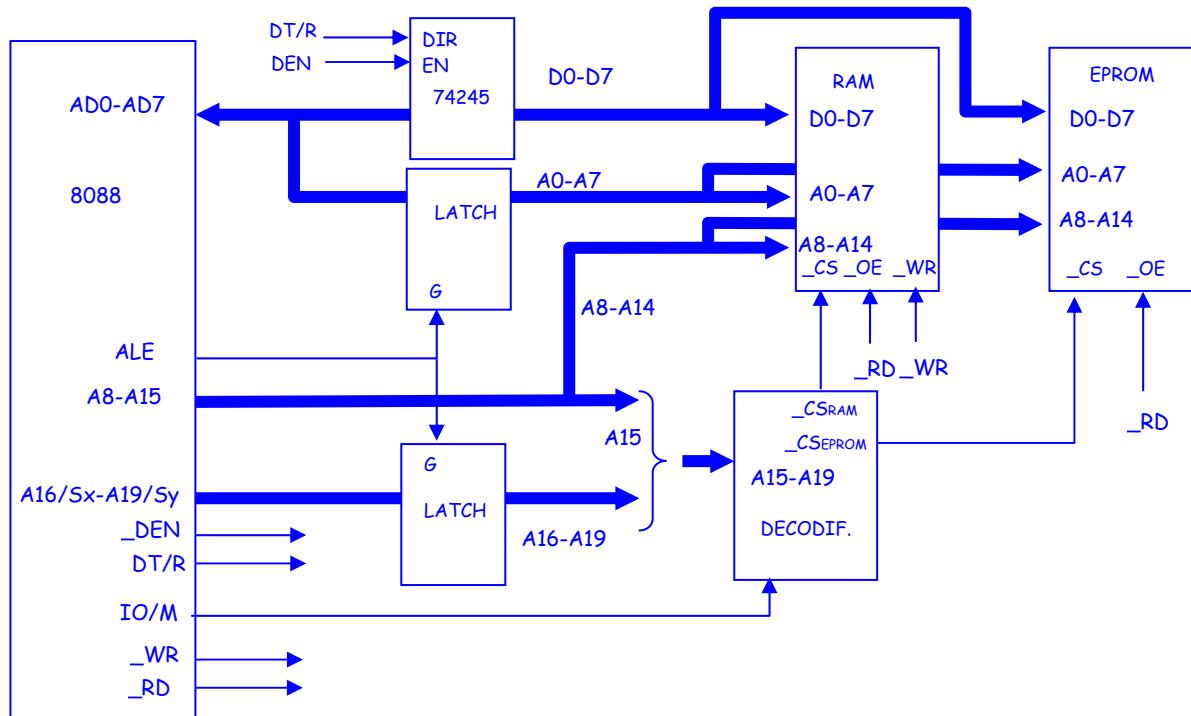
$$(_CS_{RAM})' = A'_{19} \cdot A'_{18} \cdot A'_{17} \cdot A'_{16} \cdot A'_{15} \cdot (IO/_M)'$$

$$_CS_{RAM} = [A'_{19} \cdot A'_{18} \cdot A'_{17} \cdot A'_{16} \cdot A'_{15} \cdot (IO/_M)]'$$

De la tabla anterior $_CS_{EPROM}$ debe ser cero (0) cuando $IO/_M=0$ y $A19=1$ y $A18=0$ y $A17=0$ y $A16=0$ y $A15=0$.
 En forma de ecuación esto es:

$$(_CS_{EPROM})' = A_{19} \cdot A'_{18} \cdot A'_{17} \cdot A'_{16} \cdot A'_{15} \cdot (IO/_M)'$$

$$_CS_{EPROM} = [A_{19} \cdot A'_{18} \cdot A'_{17} \cdot A'_{16} \cdot A'_{15} \cdot (IO/_M)]'$$

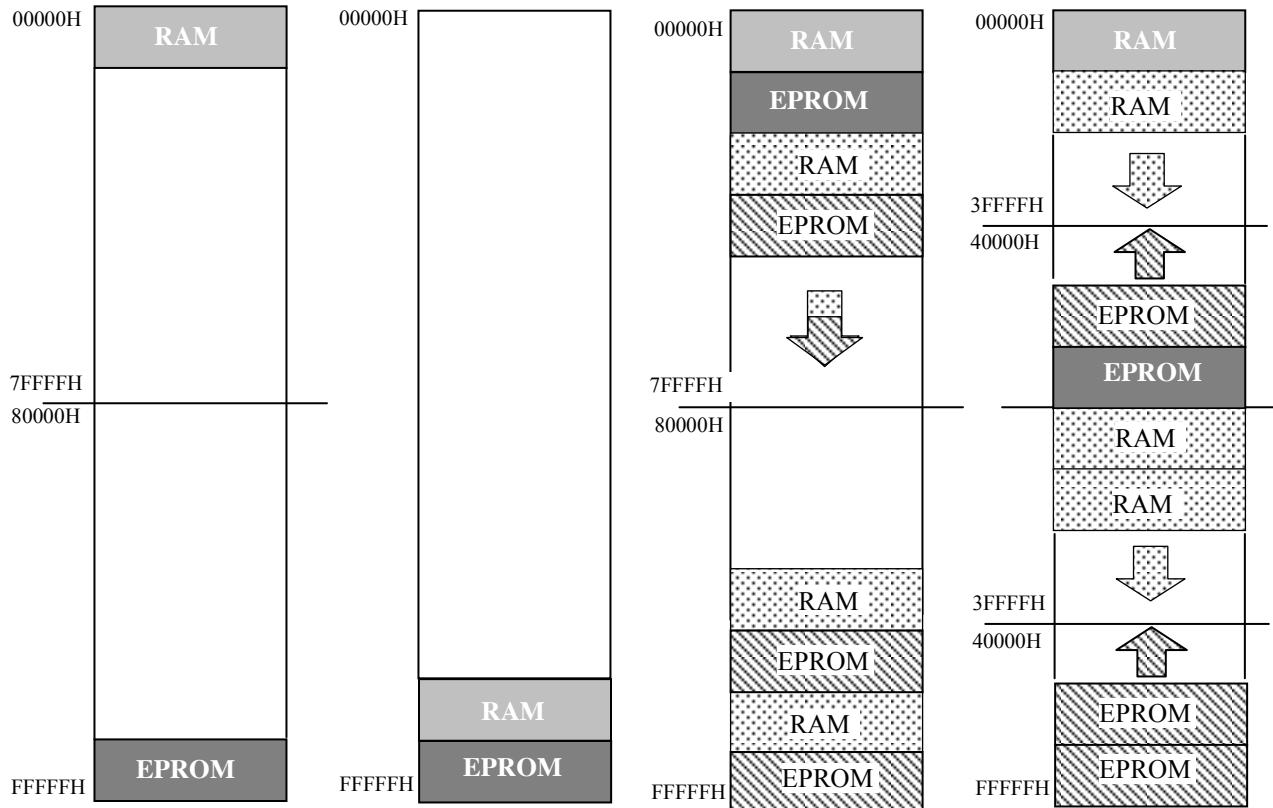


EJERCICIOS DEODIFICADORES

UNIVERSIDAD AUTONOMA DE BAJA CALIFORNIA
Facultad de Ciencias Químicas e Ingeniería
Ingeniero en Computación
Arquitectura de Computadoras II

Tarea

Diseñe los decodificadores de memoria para las siguientes configuraciones de memoria del 80C188. (EPROM: 8K, RAM: 8K)



Direcciones
Espejo de RAM

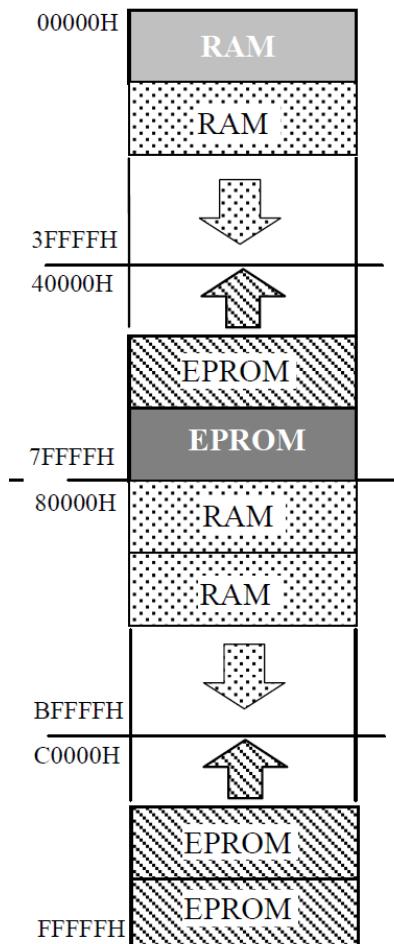


Direcciones
Espejo de EPROM



Tarea Decodificadores (inciso 4)

Diseñe los decodificadores de Memoria para las siguientes configuraciones de memoria del 80C88. (EPROM: 8K, RAM:8K)



Rango de RAM:

$$8K = 1K * 8 = 1024(8) = 8192 \text{ (8192 localidades)}$$

Dirección inicial = 0000H (todos sus bits son ceros)

Dirección final = (Tamaño de la memoria -1)

/* menos 1 porque la 0 cuenta como localidad */

Dirección final = 1FFFH (todos sus bits son unos)

Rango de EPROM:

$$8K = 1K * 8 = 1024(8) = 8192 \text{ (8192 localidades)}$$

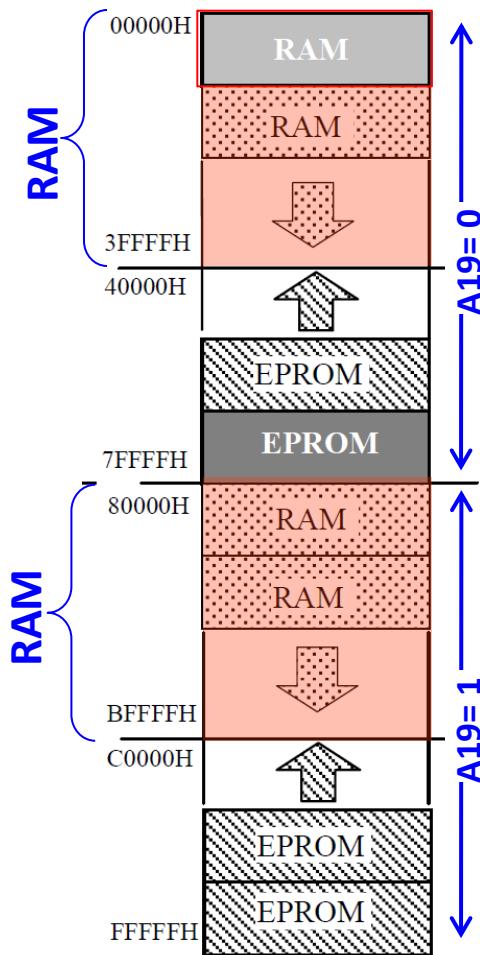
Dirección inicial = 0000H (todos sus bits son ceros)

Dirección final = (Tamaño de la memoria -1)

/* menos 1 porque la 0 cuenta como localidad */

Dirección final = 1FFFH (todos sus bits son unos)

Direcciones de la RAM:



Analizando la primera mitad del espacio de memoria (00000H-3FFFFH)

La RAM está en los primeros 8K, es decir de la dirección física 00000H a la 01FFFH. Pero también es vista (accesible) en los siguientes bloques de 8K hasta la dirección 3FFFFH. Es decir, existen direcciones espejo.

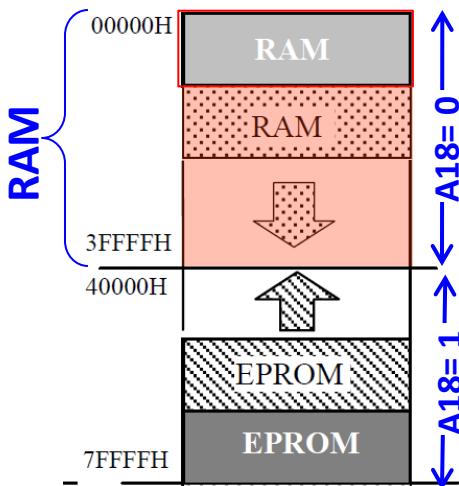
Analizando la segunda mitad del espacio de memoria (80000H-FFFFFH)

La RAM puede verse que inicia en los primeros 8K (de la segunda mitad) iniciando en la dirección física 80000H a la 81FFFH. También es vista en los siguientes bloques de 8K hasta la dirección BFFFFH.

Lo que podemos ver de esto es que esta sección de RAM de la segunda mitad es el espejo de la sección de RAM de la primera mitad. Si analizamos los rangos donde es vista la memoria podemos concluir que la única línea de dirección que difiere entre esa dos secciones es la A19. Esto es, con A19=0 se está en la primera mitad y con A19=1 en la segunda mitad.

Con esto podemos entonces enfocar el diseño a sólo la sección de la RAM de la primera mitad sin considerar A19 para que el decodificador obtenido haga que la memoria sea vista también al inicio de la segunda mitad del espacio de memoria,

Enfocándose a la primera mitad únicamente (Rango: 00000H-7FFFFH).



Aquí se puede observar que la memoria sólo es vista cuando $A18=0$, es la primera mitad de este espacio de memoria. Pues en el otro caso cuando $A18=1$ (segunda mitad) la RAM no es accesible.

Está detección puede determinarse haciendo una tabla con las líneas de dirección involucradas que sería desde A18 hasta A13. Esto pues A12 a A0 no deben considerarse pues son las líneas de dirección que tiene la memoria RAM y que no nos ayudan a determinar en qué bloque de 8K estamos.

Haciendo la tabla veríamos que de A17 a A13 cambiarían en todo el rango de la sección de RAM (00000H a 3FFFFH) y que A18 permanecería constante en 0 ($A18=0$). Por lo que sólo A18 ayuda a determinar donde estamos.

Por tanto el _CS (chip_select) de la RAM depende de A18 y de IO/M (Recordemos que es un decodificador de memoria). Así que tenemos:

Entradas		Salida	
IO/_M	A18	_CS	
0	0	0	-- Acceso a la RAM
0	1	1	-- No acceso a RAM
1	0	1	-- Acceso a E/S
1	1	1	-- Acceso a E/S

$$_CS = IO/_M + A18$$

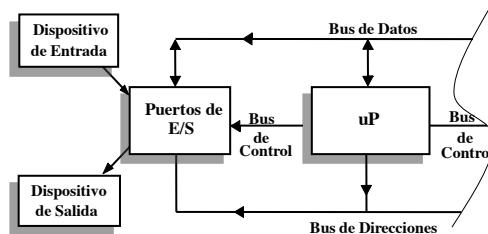
SECCION ENTRADA/SALIDA

Arquitectura de Computadoras

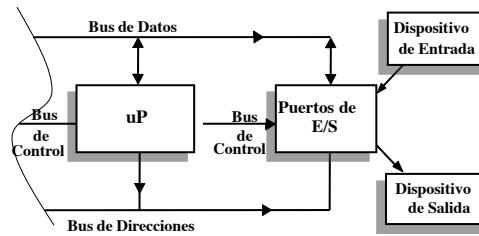
Un enfoque hacia sistemas empotrados
(Embedded Systems)

Sección de Entrada-Salida

Puertos

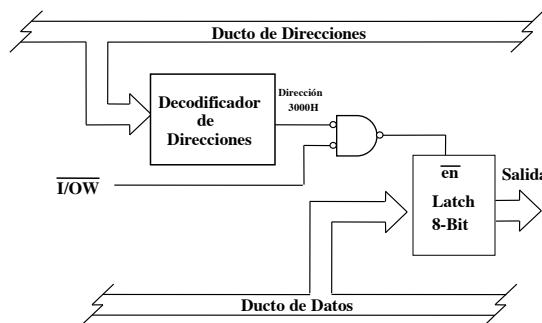


Puertos



3

Puertos de Salida



Lenguaje Ensamblador

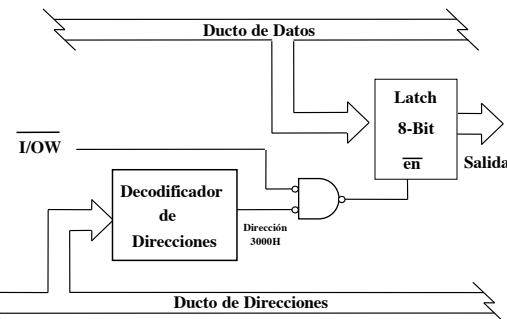
```
MOV AL, 00H  
MOV DX, 3000H  
OUT DX, AL
```

Lenguaje C

```
outportb(0x3000, 0);
```

4

Puertos de Salida



Lenguaje Ensamblador

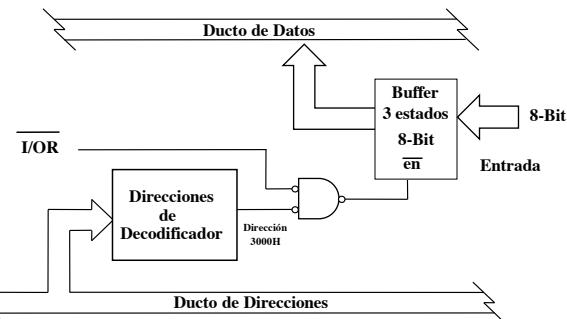
```
MOV AL, 00H  
MOV DX, 3000H  
OUT DX, AL
```

Lenguaje C

```
outportb(0x3000, 0);
```

5

Puertos de Salida



Lenguaje Ensamblador

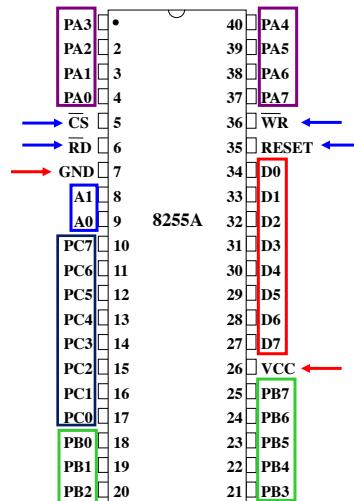
```
MOV DX, 3000H  
IN AL, DX
```

Lenguaje C

```
dato = inportb(0x3000);
```

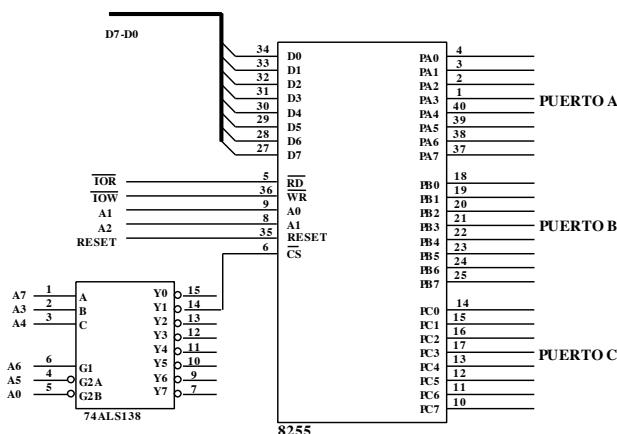
6

PPI-8255



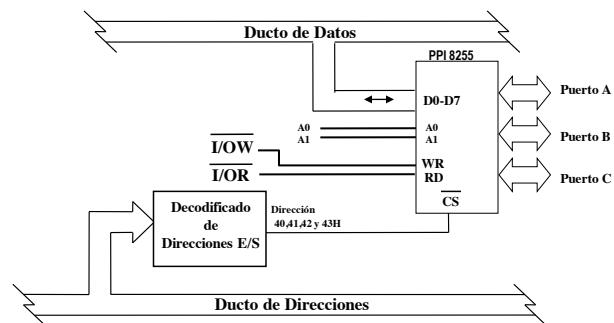
7

PPI-8255



8

Puertos de Salida



Lenguaje Ensamblador

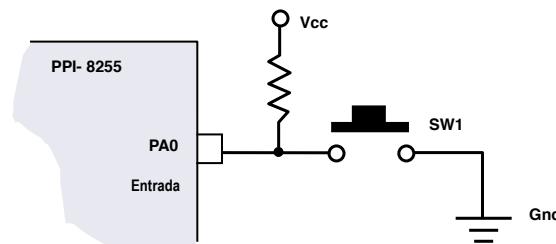
```
MOV AL, 00H  
MOV DX, 42H  
OUT DX, AL
```

Lenguaje C

```
outportb(0x42, 0);
```

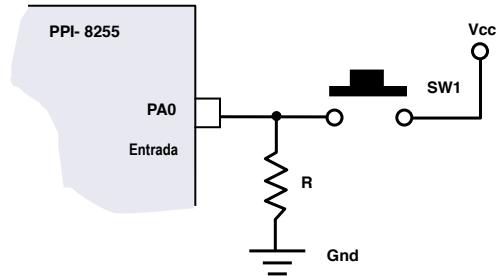
9

Lectura de Interruptor (SW)



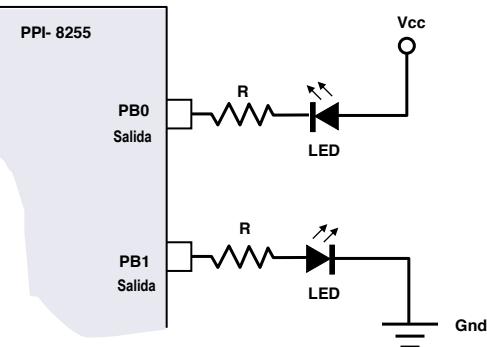
10

Lectura de Interruptor (SW)



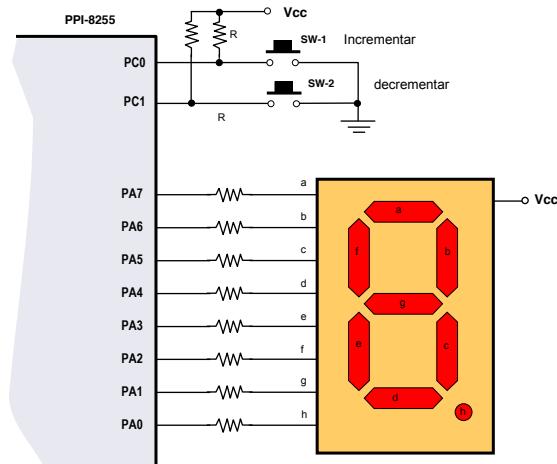
11

Activación de un LED (indicador)



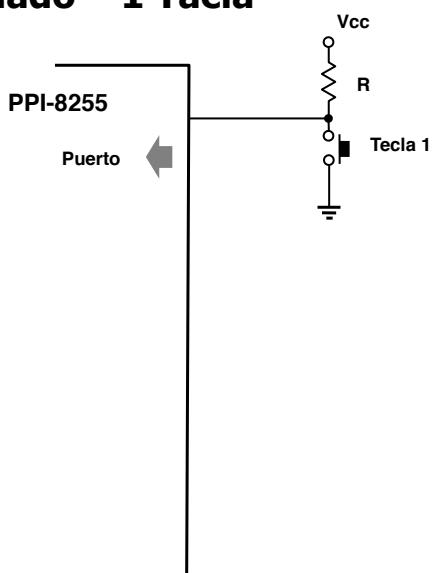
12

Indicador de 7 Segmentos e Interruptores



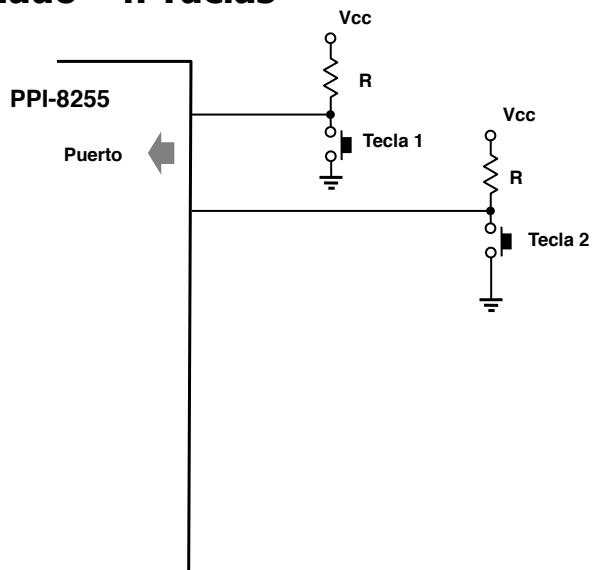
13

Teclado – 1 Tacla



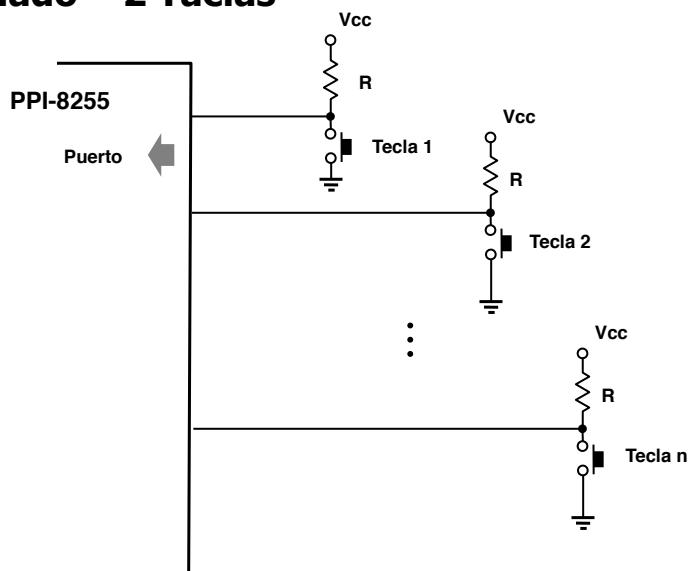
14

Teclado – n Taclas



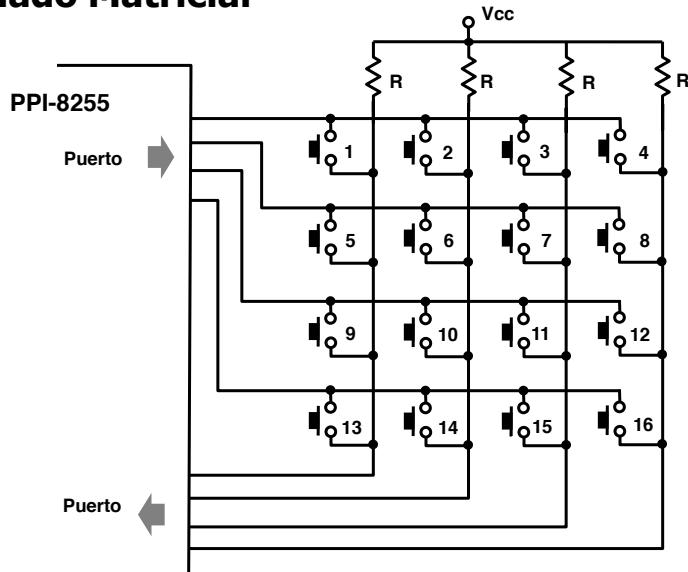
15

Teclado – 2 Taclas



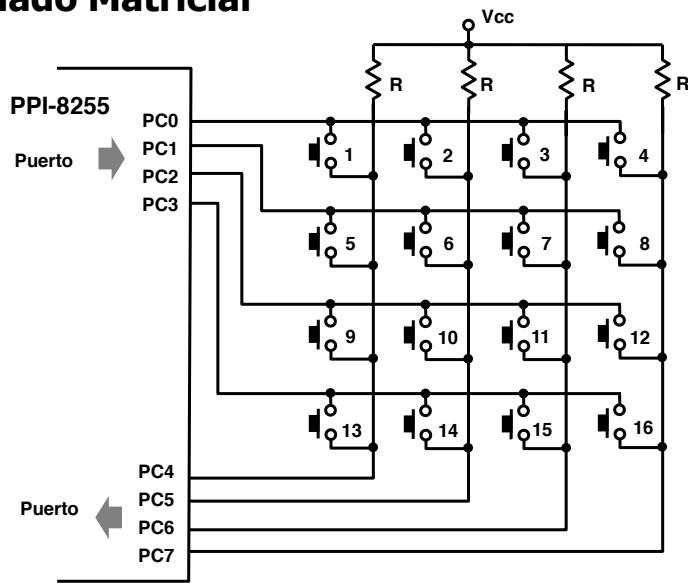
16

Teclado Matricial



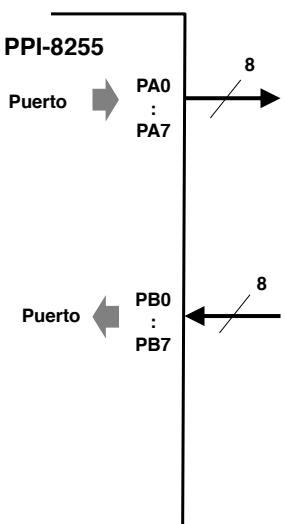
17

Teclado Matricial



18

Teclado Matricial



19

DATA SHEET (intersil)

CMOS Programmable Peripheral Interface

June 1998

Features

- Pin Compatible with NMOS 8255A
- 24 Programmable I/O Pins
- Fully TTL Compatible
- High Speed, No "Wait State" Operation with 5MHz and 8MHz 80C86 and 80C88
- Direct Bit Set/Reset Capability
- Enhanced Control Word Read Capability
- L7 Process
- 2.5mA Drive Capability on All I/O Ports
- Low Standby Power (ICCSB) 10µA

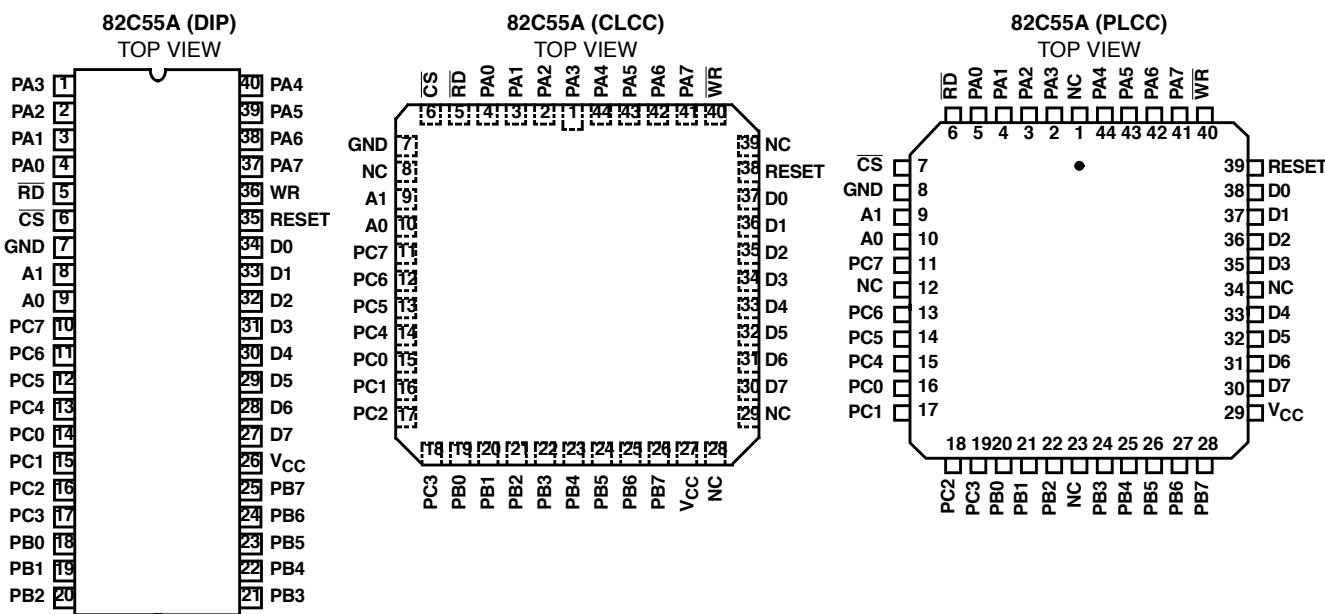
Ordering Information

PART NUMBERS		PACKAGE	TEMPERATURE RANGE	PKG. NO.
5MHz	8MHz			
CP82C55A-5	CP82C55A	40 Ld PDIP	0°C to 70°C	E40.6
IP82C55A-5	IP82C55A		-40°C to 85°C	E40.6
CS82C55A-5	CS82C55A	44 Ld PLCC	0°C to 70°C	N44.65
IS82C55A-5	IS82C55A		-40°C to 85°C	N44.65
CD82C55A-5	CD82C55A	40 Ld CERDIP	0°C to 70°C	F40.6
ID82C55A-5	ID82C55A		-40°C to 85°C	F40.6
MD82C55A-5/B	MD82C55A/B	40 Ld CERDIP	-55°C to 125°C	F40.6
8406601QA	8406602QA	SMD#		F40.6
MR82C55A-5/B	MR82C55A/B	44 Pad CLCC	-55°C to 125°C	J44.A
8406601XA	8406602XA	SMD#		J44.A

Description

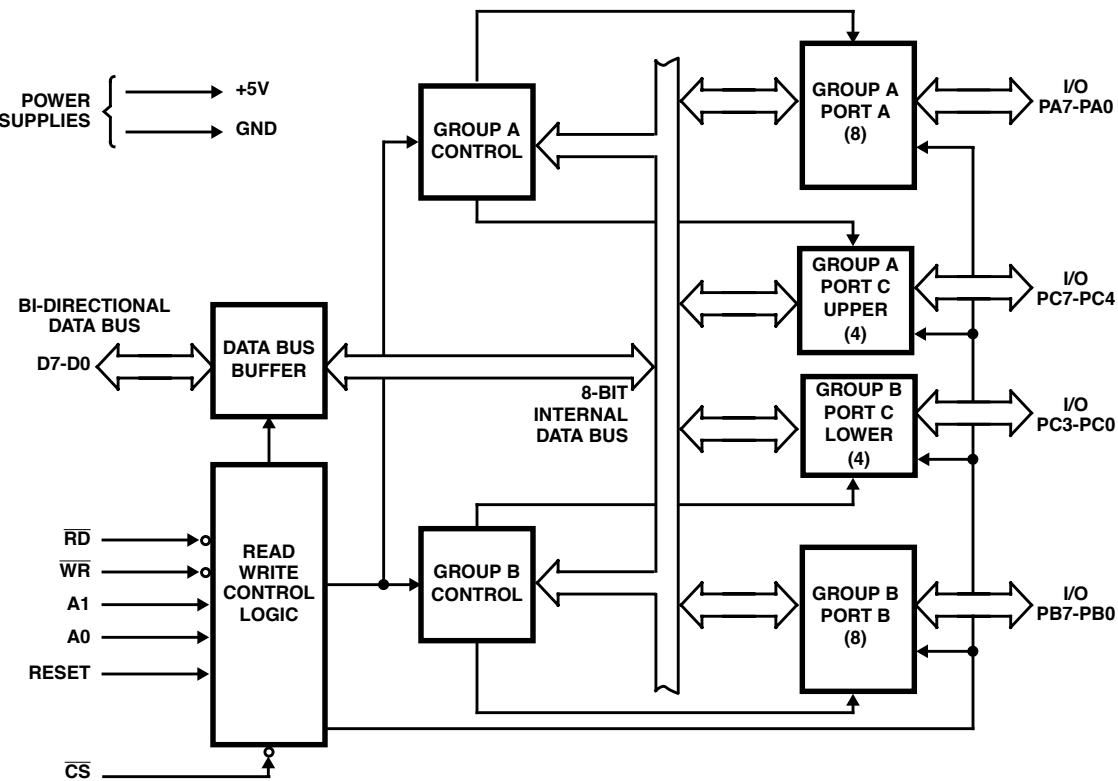
The Intersil 82C55A is a high performance CMOS version of the industry standard 8255A and is manufactured using a self-aligned silicon gate CMOS process (Scaled SAJ IV). It is a general purpose programmable I/O device which may be used with many different microprocessors. There are 24 I/O pins which may be individually programmed in 2 groups of 12 and used in 3 major modes of operation. The high performance and industry standard configuration of the 82C55A make it compatible with the 80C86, 80C88 and other microprocessors.

Static CMOS circuit design insures low operating power. TTL compatibility over the full military temperature range and bus hold circuitry eliminate the need for pull-up resistors. The Intersil advanced SAJ process results in performance equal to or greater than existing functionally equivalent products at a fraction of the power.

Pinouts

Pin Description

SYMBOL	PIN NUMBER	TYPE	DESCRIPTION
V _{CC}	26		V _{CC} : The +5V power supply pin. A 0.1 μ F capacitor between pins 26 and 7 is recommended for decoupling.
GND	7		GROUND
D0-D7	27-34	I/O	DATA BUS: The Data Bus lines are bidirectional three-state pins connected to the system data bus.
RESET	35	I	RESET: A high on this input clears the control register and all ports (A, B, C) are set to the input mode with the "Bus Hold" circuitry turned on.
CS	6	I	CHIP SELECT: Chip select is an active low input used to enable the 82C55A onto the Data Bus for CPU communications.
RD	5	I	READ: Read is an active low input control signal used by the CPU to read status information or data via the data bus.
WR	36	I	WRITE: Write is an active low input control signal used by the CPU to load control words and data into the 82C55A.
A0-A1	8, 9	I	ADDRESS: These input signals, in conjunction with the RD and WR inputs, control the selection of one of the three ports or the control word register. A0 and A1 are normally connected to the least significant bits of the Address Bus A0, A1.
PA0-PA7	1-4, 37-40	I/O	PORT A: 8-bit input and output port. Both bus hold high and bus hold low circuitry are present on this port.
PB0-PB7	18-25	I/O	PORT B: 8-bit input and output port. Bus hold high circuitry is present on this port.
PC0-PC7	10-17	I/O	PORT C: 8-bit input and output port. Bus hold circuitry is present on this port.

Functional Diagram

Functional Description

Data Bus Buffer

This three-state bi-directional 8-bit buffer is used to interface the 82C55A to the system data bus. Data is transmitted or received by the buffer upon execution of input or output instructions by the CPU. Control words and status information are also transferred through the data bus buffer.

Read/Write and Control Logic

The function of this block is to manage all of the internal and external transfers of both Data and Control or Status words. It accepts inputs from the CPU Address and Control busses and in turn, issues commands to both of the Control Groups.

(CS) Chip Select. A “low” on this input pin enables the communication between the 82C55A and the CPU.

(RD) Read. A “low” on this input pin enables 82C55A to send the data or status information to the CPU on the data bus. In essence, it allows the CPU to “read from” the 82C55A.

(WR) Write. A “low” on this input pin enables the CPU to write data or control words into the 82C55A.

(A0 and A1) Port Select 0 and Port Select 1. These input signals, in conjunction with the RD and WR inputs, control the selection of one of the three ports or the control word register. They are normally connected to the least significant bits of the address bus (A0 and A1).

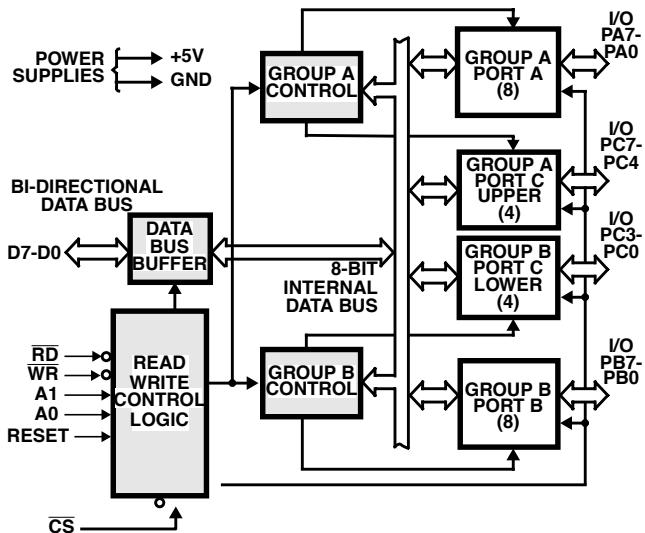


FIGURE 1. 82C55A BLOCK DIAGRAM. DATA BUS BUFFER, READ/WRITE, GROUP A & B CONTROL LOGIC FUNCTIONS

(RESET) Reset. A “high” on this input initializes the control register to 9Bh and all ports (A, B, C) are set to the input mode. “Bus hold” devices internal to the 82C55A will hold the I/O port inputs to a logic “1” state with a maximum hold current of 400 μ A.

Group A and Group B Controls

The functional configuration of each port is programmed by the systems software. In essence, the CPU “outputs” a control word to the 82C55A. The control word contains information such as “mode”, “bit set”, “bit reset”, etc., that initializes the functional configuration of the 82C55A.

Each of the Control blocks (Group A and Group B) accepts “commands” from the Read/Write Control logic, receives “control words” from the internal data bus and issues the proper commands to its associated ports.

Control Group A - Port A and Port C upper (C7 - C4)

Control Group B - Port B and Port C lower (C3 - C0)

The control word register can be both written and read as shown in the “Basic Operation” table. Figure 4 shows the control word format for both Read and Write operations. When the control word is read, bit D7 will always be a logic “1”, as this implies control word mode information.

82C55A BASIC OPERATION

INPUT OPERATION (READ)					
A1	A0	RD	WR	CS	
0	0	0	1	0	Port A → Data Bus
0	1	0	1	0	Port B → Data Bus
1	0	0	1	0	Port C → Data Bus
1	1	0	1	0	Control Word → Data Bus
OUTPUT OPERATION (WRITE)					
0	0	1	0	0	Data Bus → Port A
0	1	1	0	0	Data Bus → Port B
1	0	1	0	0	Data Bus → Port C
1	1	1	0	0	Data Bus → Control
DISABLE FUNCTION					
X	X	X	X	1	Data Bus → Three-State
X	X	1	1	0	Data Bus → Three-State

Ports A, B, and C

The 82C55A contains three 8-bit ports (A, B, and C). All can be configured to a wide variety of functional characteristics by the system software but each has its own special features or "personality" to further enhance the power and flexibility of the 82C55A.

Port A One 8-bit data output latch/buffer and one 8-bit data input latch. Both "pull-up" and "pull-down" bus-hold devices are present on Port A. See Figure 2A.

Port B One 8-bit data input/output latch/buffer and one 8-bit data input buffer. See Figure 2B.

Port C One 8-bit data output latch/buffer and one 8-bit data input buffer (no latch for input). This port can be divided into two 4-bit ports under the mode control. Each 4-bit port contains a 4-bit latch and it can be used for the control signal output and status signal inputs in conjunction with ports A and B. See Figure 2B.

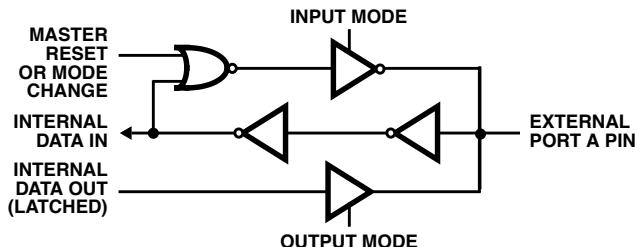


FIGURE 2A. PORT A BUS-HOLD CONFIGURATION

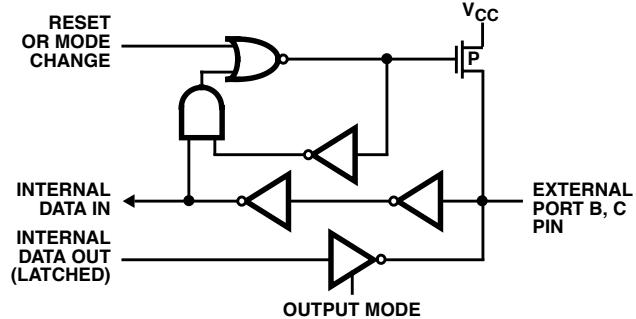


FIGURE 2B. PORT B AND C BUS-HOLD CONFIGURATION

FIGURE 2. BUS-HOLD CONFIGURATION

Operational Description

Mode Selection

There are three basic modes of operation than can be selected by the system software:

- Mode 0 - Basic Input/Output
- Mode 1 - Strobed Input/Output
- Mode 2 - Bi-directional Bus

When the reset input goes "high", all ports will be set to the input mode with all 24 port lines held at a logic "one" level by internal bus hold devices. After the reset is removed, the 82C55A can remain in the input mode with no additional initialization required. This eliminates the need to pullup or pulldown resistors in all-CMOS designs. The control word

register will contain 9Bh. During the execution of the system program, any of the other modes may be selected using a single output instruction. This allows a single 82C55A to service a variety of peripheral devices with a simple software maintenance routine. Any port programmed as an output port is initialized to all zeros when the control word is written.

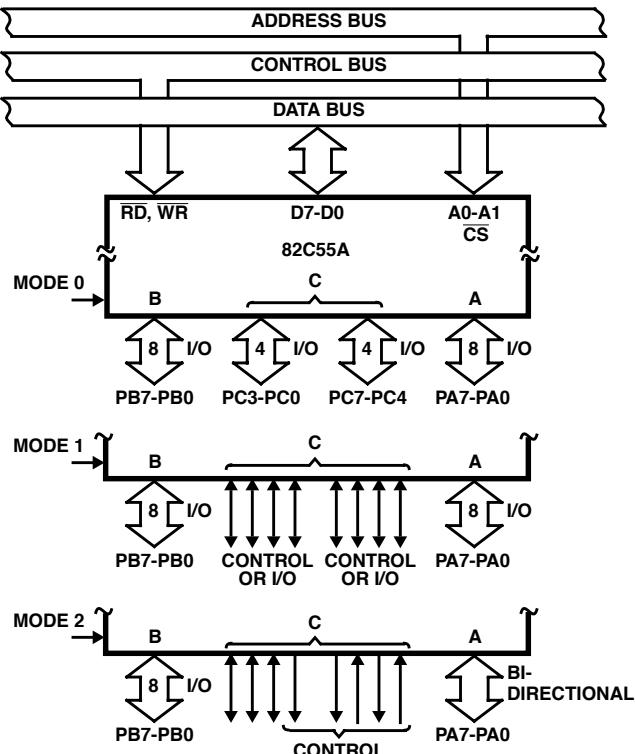


FIGURE 3. BASIC MODE DEFINITIONS AND BUS INTERFACE

CONTROL WORD

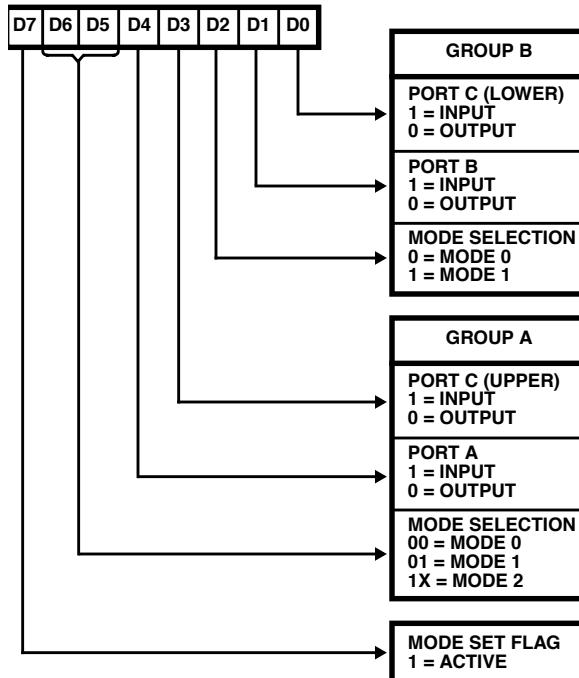


FIGURE 4. MODE DEFINITION FORMAT

The modes for Port A and Port B can be separately defined, while Port C is divided into two portions as required by the Port A and Port B definitions. All of the output registers, including the status flip-flops, will be reset whenever the mode is changed. Modes may be combined so that their functional definition can be "tailored" to almost any I/O structure. For instance: Group B can be programmed in Mode 0 to monitor simple switch closings or display computational results, Group A could be programmed in Mode 1 to monitor a keyboard or tape reader on an interrupt-driven basis.

The mode definitions and possible mode combinations may seem confusing at first, but after a cursory review of the complete device operation a simple, logical I/O approach will surface. The design of the 82C55A has taken into account things such as efficient PC board layout, control signal definition vs. PC layout and complete functional flexibility to support almost any peripheral device with no external logic. Such design represents the maximum use of the available pins.

Single Bit Set/Reset Feature (Figure 5)

Any of the eight bits of Port C can be Set or Reset using a single Output instruction. This feature reduces software requirements in control-based applications.

When Port C is being used as status/control for Port A or B, these bits can be set or reset by using the Bit Set/Reset operation just as if they were output ports.

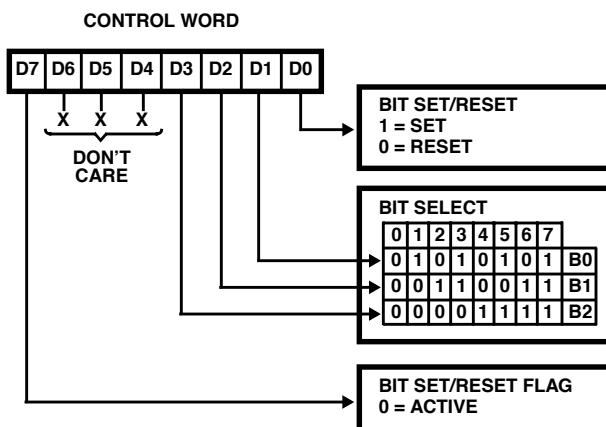


FIGURE 5. BIT SET/RESET FORMAT

Interrupt Control Functions

When the 82C55A is programmed to operate in mode 1 or mode 2, control signals are provided that can be used as interrupt request inputs to the CPU. The interrupt request signals, generated from port C, can be inhibited or enabled by setting or resetting the associated INTE flip-flop, using the bit set/reset function of port C.

This function allows the programmer to enable or disable a CPU interrupt by a specific I/O device without affecting any other device in the interrupt structure.

INTE Flip-Flop Definition

(BIT-SET)-INTE is SET - Interrupt Enable

(BIT-RESET)-INTE is Reset - Interrupt Disable

NOTE: All Mask flip-flops are automatically reset during mode selection and device Reset.

Operating Modes

Mode 0 (Basic Input/Output). This functional configuration provides simple input and output operations for each of the three ports. No handshaking is required, data is simply written to or read from a specific port.

Mode 0 Basic Functional Definitions:

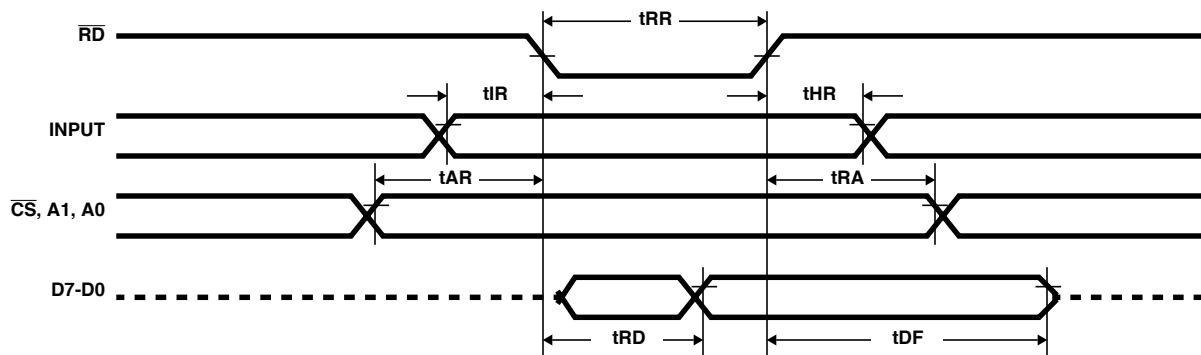
- Two 8-bit ports and two 4-bit ports
- Any Port can be input or output
- Outputs are latched
- Inputs are not latched
- 16 different Input/Output configurations possible

MODE 0 PORT DEFINITION

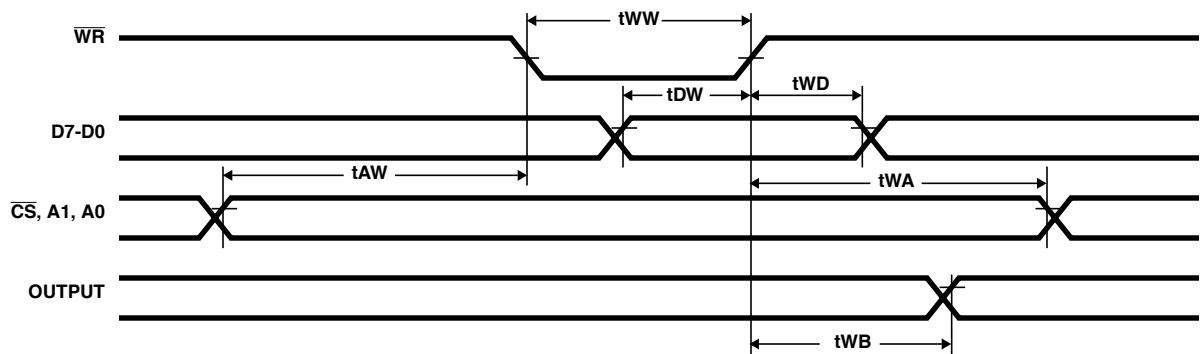
A			B		GROUP A		#	GROUP B	
D4	D3	D1	D0	PORTC (Upper)	PORTC (Lower)	PORT B	PORT C (Lower)		
0	0	0	0	Output	Output	0	Output	Output	
0	0	0	1	Output	Output	1	Output	Input	
0	0	1	0	Output	Output	2	Input	Output	
0	0	1	1	Output	Output	3	Input	Input	
0	1	0	0	Output	Input	4	Output	Output	
0	1	0	1	Output	Input	5	Output	Input	
0	1	1	0	Output	Input	6	Input	Output	
0	1	1	1	Output	Input	7	Input	Input	
1	0	0	0	Input	Output	8	Output	Output	
1	0	0	1	Input	Output	9	Output	Input	
1	0	1	0	Input	Output	10	Input	Output	
1	0	1	1	Input	Output	11	Input	Input	
1	1	0	0	Input	Input	12	Output	Output	
1	1	0	1	Input	Input	13	Output	Input	
1	1	1	0	Input	Input	14	Input	Output	
1	1	1	1	Input	Input	15	Input	Input	

82C55A

Mode 0 (Basic Input)

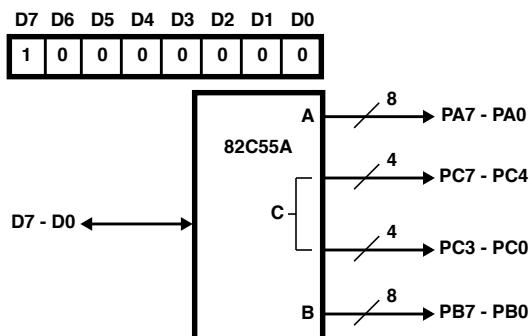


Mode 0 (Basic Output)

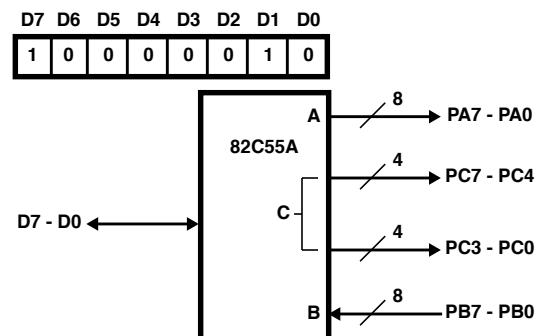


Mode 0 Configurations

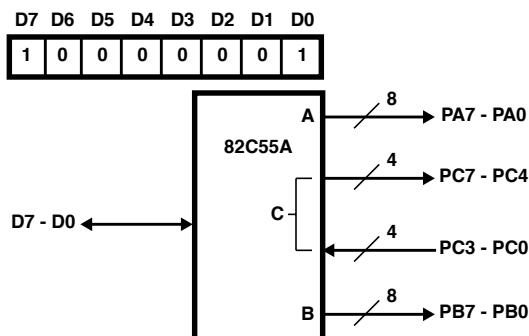
CONTROL WORD #0



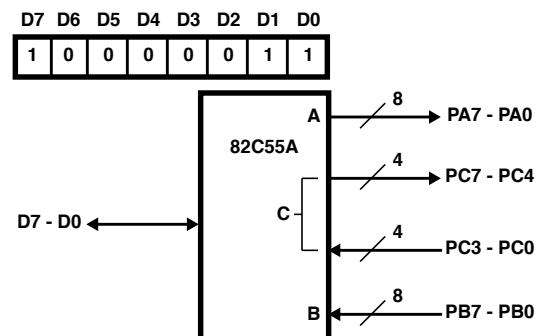
CONTROL WORD #2



CONTROL WORD #1



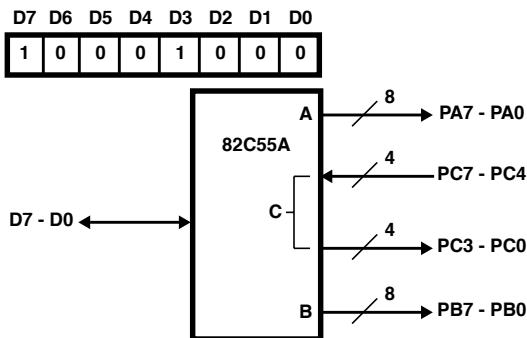
CONTROL WORD #3



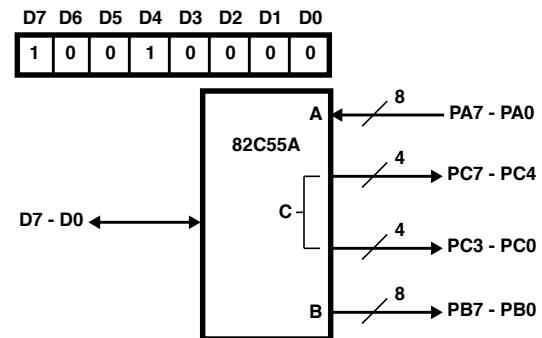
82C55A

Mode 0 Configurations (Continued)

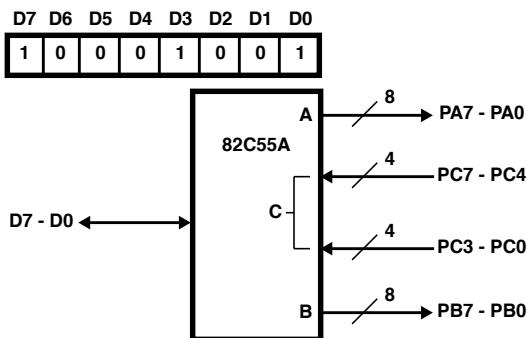
CONTROL WORD #4



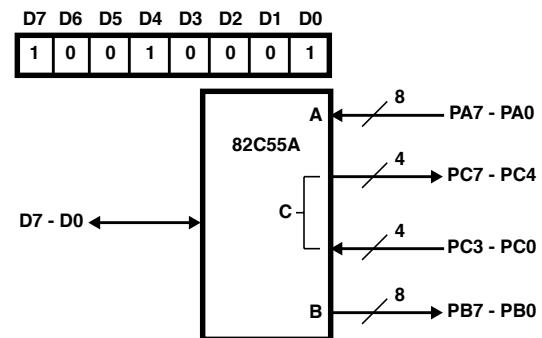
CONTROL WORD #8



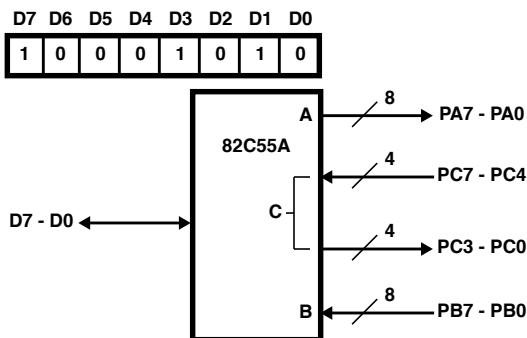
CONTROL WORD #5



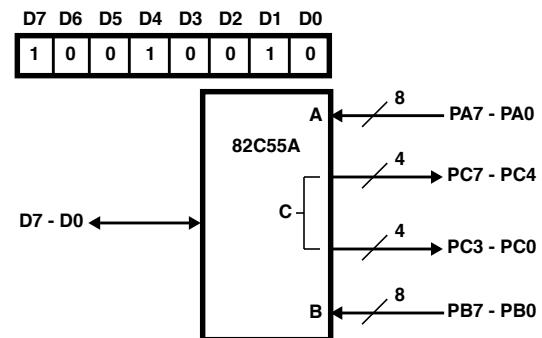
CONTROL WORD #9



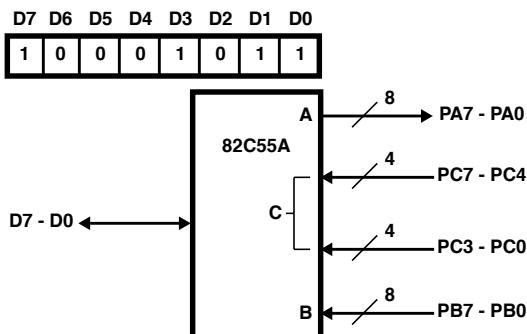
CONTROL WORD #6



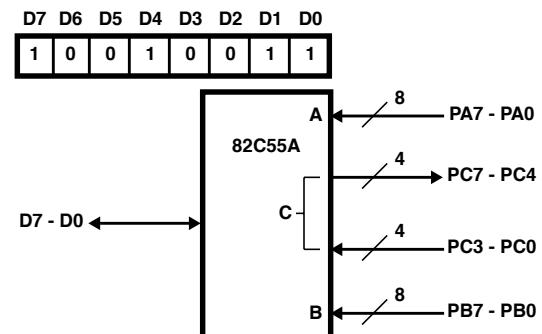
CONTROL WORD #10



CONTROL WORD #7

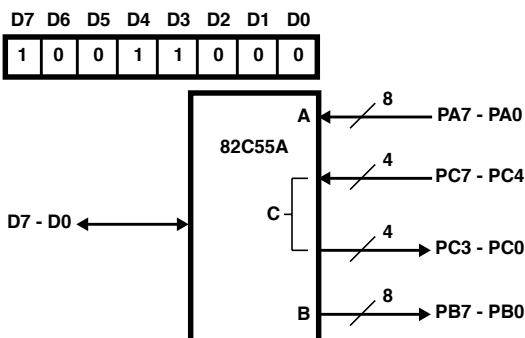


CONTROL WORD #11

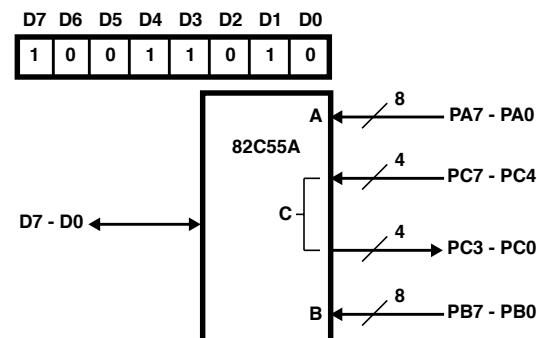


Mode 0 Configurations (Continued)

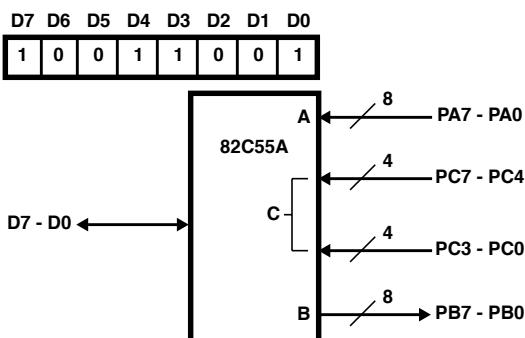
CONTROL WORD #12



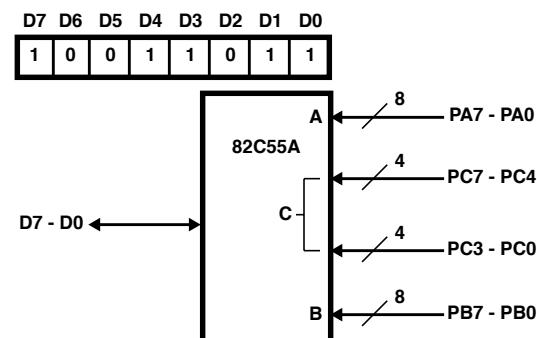
CONTROL WORD #14



CONTROL WORD #13



CONTROL WORD #15

**Operating Modes**

Mode 1 - (Strobed Input/Output). This functional configuration provides a means for transferring I/O data to or from a specified port in conjunction with strobes or "hand shaking" signals. In mode 1, port A and port B use the lines on port C to generate or accept these "hand shaking" signals.

Mode 1 Basic Function Definitions:

- Two Groups (Group A and Group B)
- Each group contains one 8-bit port and one 4-bit control/data port
- The 8-bit data port can be either input or output. Both inputs and outputs are latched.
- The 4-bit port is used for control and status of the 8-bit port.

Input Control Signal Definition

(Figures 6 and 7)

STB (Strobe Input)

A "low" on this input loads data into the input latch.

IBF (Input Buffer Full F/F)

A "high" on this output indicates that the data has been loaded into the input latch: in essence, an acknowledgement. IBF is set by STB input being low and is reset by the rising edge of the RD input.

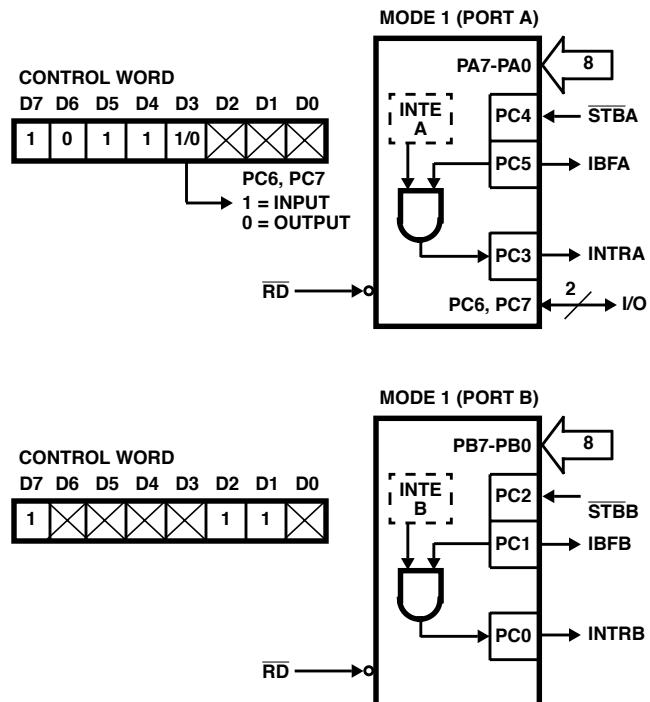


FIGURE 6. MODE 1 INPUT

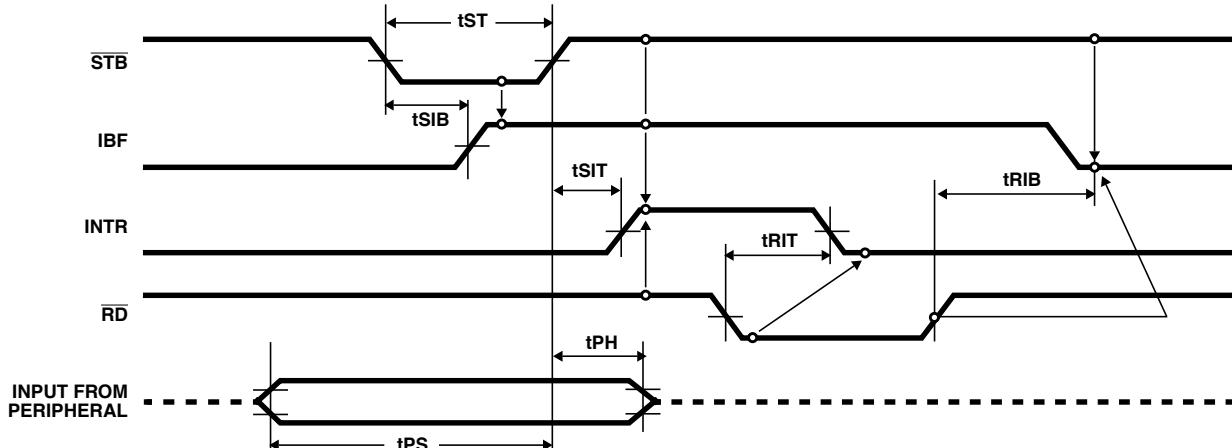


FIGURE 7. MODE 1 (STROBED INPUT)

INTR (Interrupt Request)

A “high” on this output can be used to interrupt the CPU when an input device is requesting service. INTR is set by the condition: STB is a “one”, IBF is a “one” and INTE is a “one”. It is reset by the falling edge of RD. This procedure allows an input device to request service from the CPU by simply strobing its data into the port.

INTE A

Controlled by bit set/reset of PC4.

INTE B

Controlled by bit set/reset of PC2.

Output Control Signal Definition

(Figure 8 and 9)

OBF - Output Buffer Full F/F). The \overline{OBF} output will go “low” to indicate that the CPU has written data out to be specified port. This does not mean valid data is sent out of the part at this time since \overline{OBF} can go true before data is available. Data is guaranteed valid at the rising edge of \overline{OBF} , (See Note 1). The \overline{OBF} F/F will be set by the rising edge of the WR input and reset by ACK input being low.

ACK - Acknowledge Input). A “low” on this input informs the 82C55A that the data from Port A or Port B is ready to be accepted. In essence, a response from the peripheral device indicating that it is ready to accept data, (See Note 1).

INTR - (Interrupt Request). A “high” on this output can be used to interrupt the CPU when an output device has accepted data transmitted by the CPU. INTR is set when ACK is a “one”, OBF is a “one” and INTE is a “one”. It is reset by the falling edge of WR.

INTE A

Controlled by Bit Set/Reset of PC6.

INTE B

Controlled by Bit Set/Reset of PC2.

NOTE:

1. To strobe data into the peripheral device, the user must operate the strobe line in a hand shaking mode. The user needs to send \overline{OBF} to the peripheral device, generates an ACK from the peripheral device and then latch data into the peripheral device on the rising edge of \overline{OBF} .

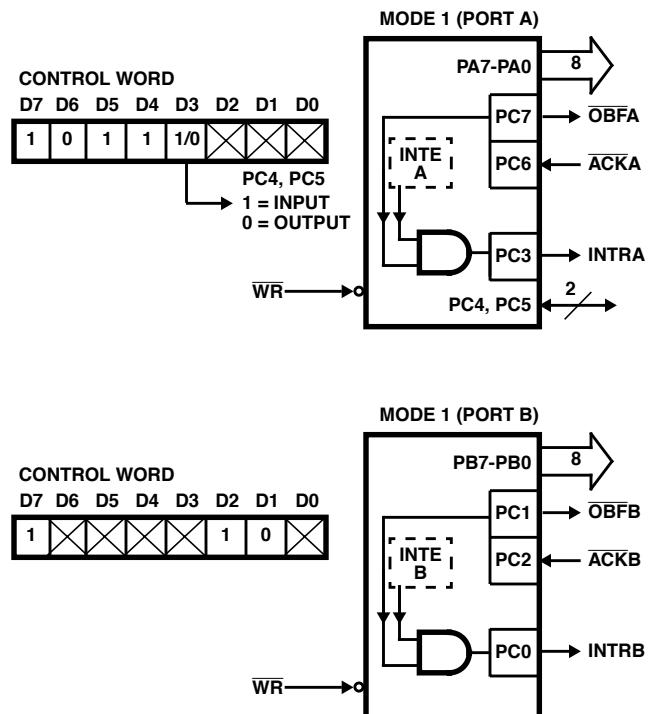


FIGURE 8. MODE 1 OUTPUT

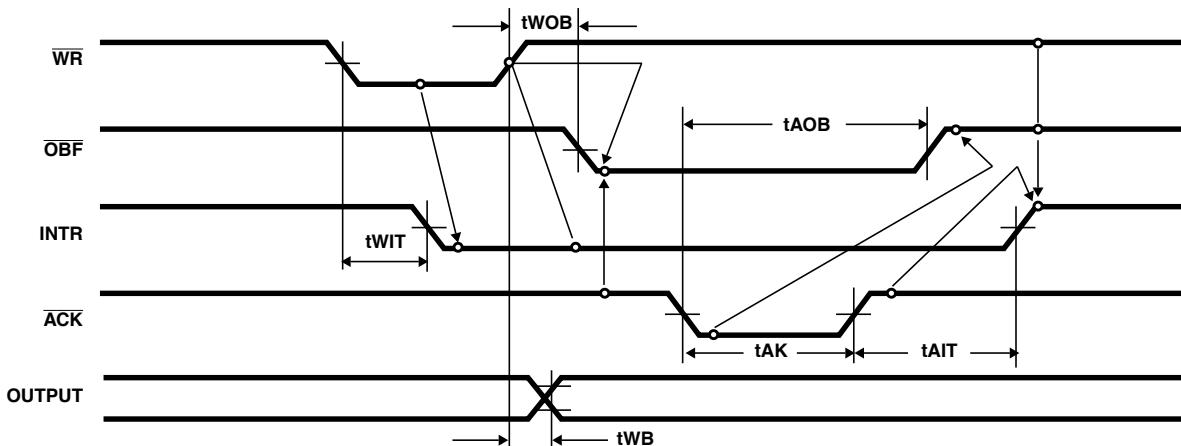
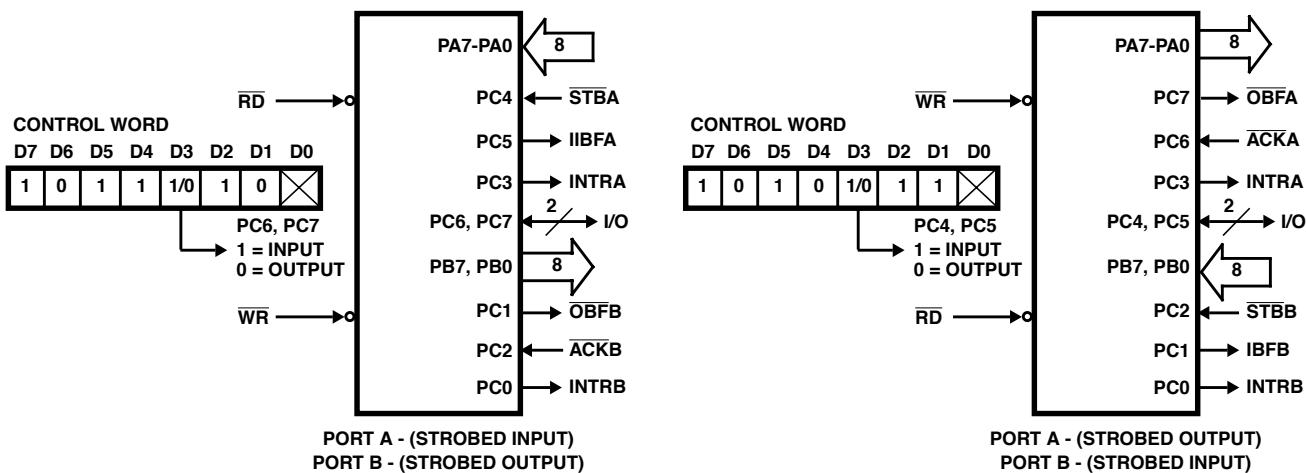


FIGURE 9. MODE 1 (STROBED OUTPUT)



Combinations of Mode 1: Port A and Port B can be individually defined as input or output in Mode 1 to support a wide variety of strobed I/O applications.

FIGURE 10. COMBINATIONS OF MODE 1

Operating Modes

Mode 2 (Strobed Bi-Directional Bus I/O)

The functional configuration provides a means for communicating with a peripheral device or structure on a single 8-bit bus for both transmitting and receiving data (bi-directional bus I/O). "Hand shaking" signals are provided to maintain proper bus flow discipline similar to Mode 1. Interrupt generation and enable/disable functions are also available.

Mode 2 Basic Functional Definitions:

- Used in Group A only
- One 8-bit, bi-directional bus Port (Port A) and a 5-bit control Port (Port C)
- Both inputs and outputs are latched
- The 5-bit control port (Port C) is used for control and status for the 8-bit, bi-directional bus port (Port A)

Bi-Directional Bus I/O Control Signal Definition

(Figures 11, 12, 13, 14)

INTR - (Interrupt Request). A high on this output can be used to interrupt the CPU for both input or output operations.

Output Operations

OBF - (Output Buffer Full). The **OBF** output will go "low" to indicate that the CPU has written data out to port A.

ACK - (Acknowledge). A "low" on this input enables the three-state output buffer of port A to send out the data. Otherwise, the output buffer will be in the high impedance state.

INTE 1 - (The INTE flip-flop associated with **OBF**). Controlled by bit set/reset of PC4.

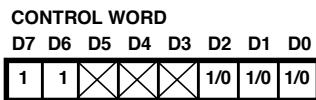
Input Operations

STB - (Strobe Input). A "low" on this input loads data into the input latch.

IBF - (Input Buffer Full F/F). A "high" on this output indicates that data has been loaded into the input latch.

INTE 2 - (The INTE flip-flop associated with **IBF**). Controlled by bit set/reset of PC4.

82C55A



PC2-PC0
 1 = INPUT
 0 = OUTPUT

PORT B
 1 = INPUT
 0 = OUTPUT

GROUP B MODE
 0 = MODE 0
 1 = MODE 1

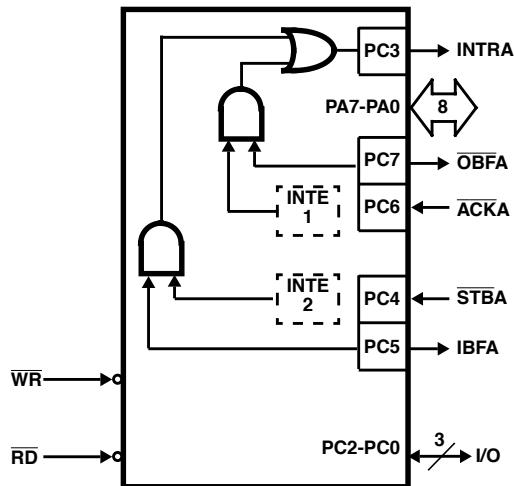
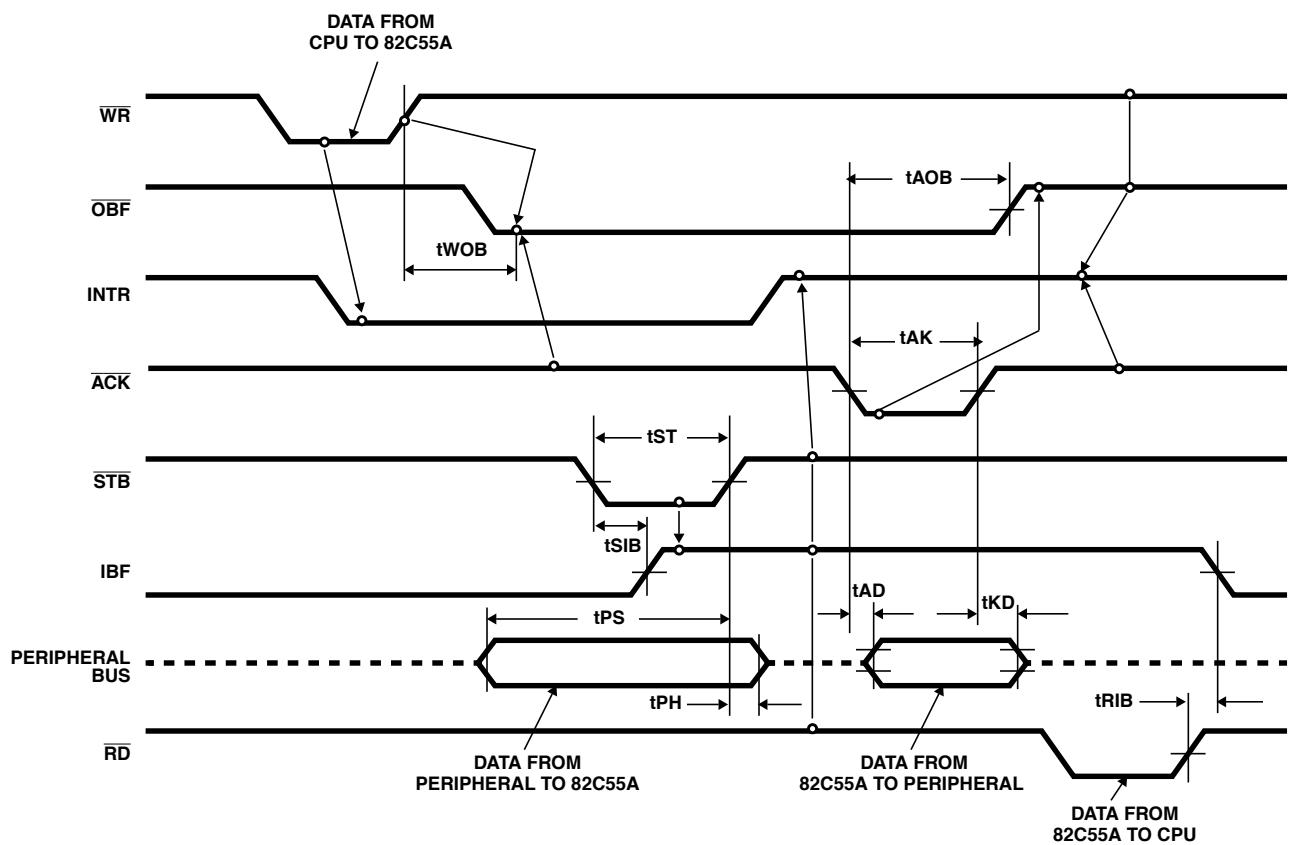


FIGURE 11. MODE CONTROL WORD

FIGURE 12. MODE 2



NOTE: Any sequence where \overline{WR} occurs before \overline{ACK} and \overline{STB} occurs before RD is permissible. ($INTR = IBF \cdot MASK \cdot \overline{STB} \cdot RD \div \overline{OBF} \cdot MASK \cdot \overline{ACK} \cdot \overline{WR}$)

FIGURE 13. MODE 2 (BI-DIRECTIONAL)

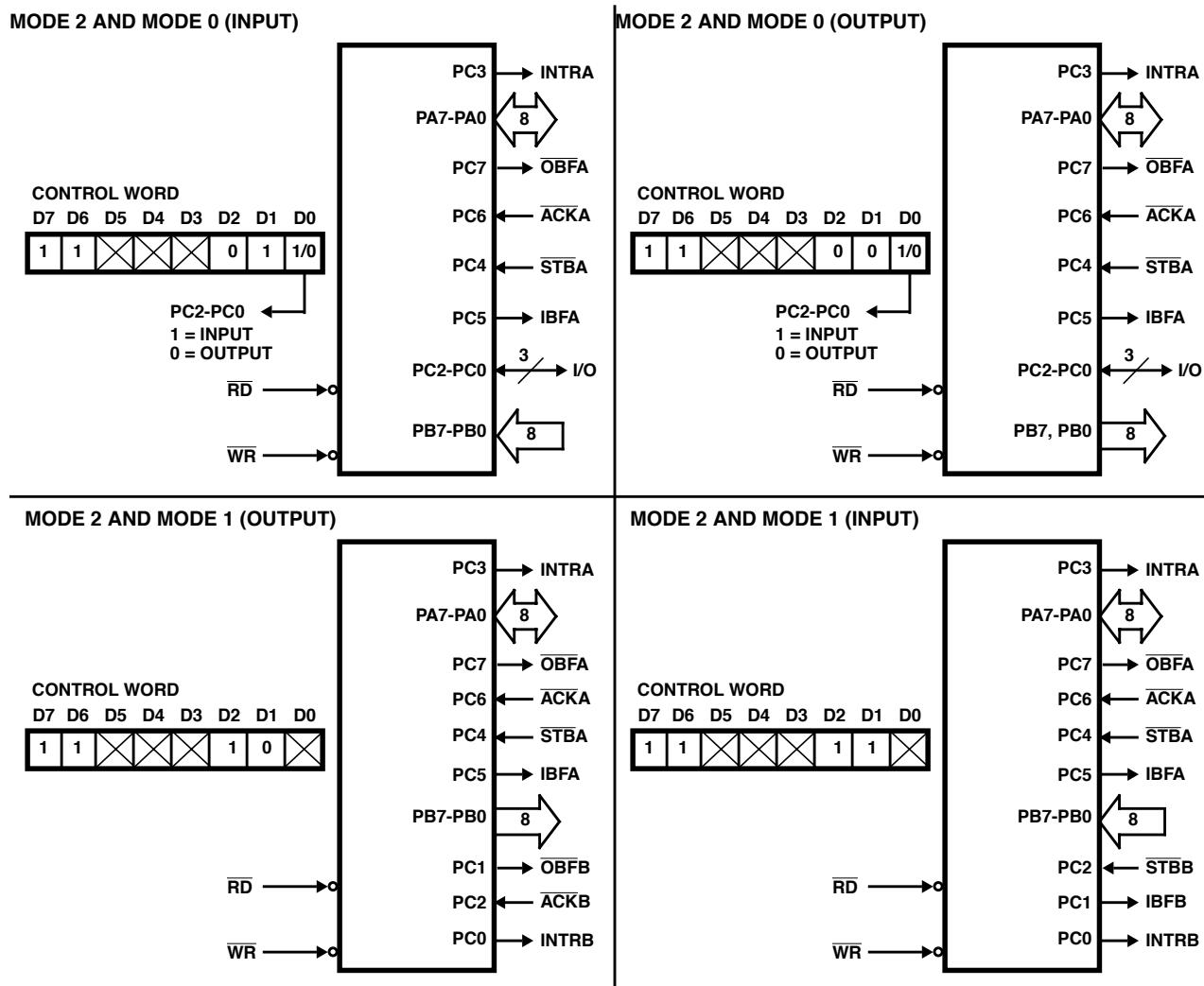


FIGURE 14. MODE 2 COMBINATIONS

	Mode Definition Summary				
	Mode 0		Mode 1		Mode 2
	In	Out	In	Out	Group A Only
PA0	In	Out	In	Out	
PA1	In	Out	In	Out	
PA2	In	Out	In	Out	
PA3	In	Out	In	Out	
PA4	In	Out	In	Out	
PA5	In	Out	In	Out	
PA6	In	Out	In	Out	
PA7	In	Out	In	Out	
PB0	In	Out	In	Out	
PB1	In	Out	In	Out	
PB2	In	Out	In	Out	
PB3	In	Out	In	Out	
PB4	In	Out	In	Out	
PB5	In	Out	In	Out	
PB6	In	Out	In	Out	
PB7	In	Out	In	Out	
PC0	In	Out	INTRB	INTRB	I/O
PC1	In	Out	IBFB	OBFB	I/O
PC2	In	Out	STBB	ACKB	I/O
PC3	In	Out	INTRA	INTRA	INTRA
PC4	In	Out	STBA	I/O	STBA
PC5	In	Out	IBFA	I/O	IBFA
PC6	In	Out	I/O	ACKA	ACKA
PC7	In	Out	I/O	OBFA	OBFA

Special Mode Combination Considerations

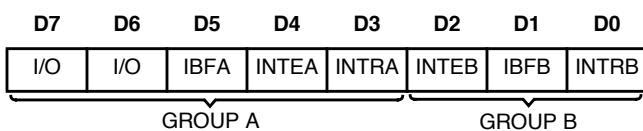
There are several combinations of modes possible. For any combination, some or all of Port C lines are used for control or status. The remaining bits are either inputs or outputs as defined by a "Set Mode" command.

During a read of Port C, the state of all the Port C lines, except the $\overline{\text{ACK}}$ and $\overline{\text{STB}}$ lines, will be placed on the data bus. In place of the $\overline{\text{ACK}}$ and $\overline{\text{STB}}$ line states, flag status will appear on the data bus in the PC2, PC4, and PC6 bit positions as illustrated by Figure 17.

Through a “Write Port C” command, only the Port C pins programmed as outputs in a Mode 0 group can be written. No other pins can be affected by a “Write Port C” command, nor can the interrupt enable flags be accessed. To write to any Port C output programmed as an output in Mode 1 group or to change an interrupt enable flag, the “Set/Reset Port C Bit” command must be used.

With a “Set/Reset Port Cea Bit” command, any Port C line programmed as an output (including IBF and OBF) can be written, or an interrupt enable flag can be either set or reset. Port C lines programmed as inputs, including ACK and STB lines, associated with Port C fare not affected by a “Set/Reset Port C Bit” command. Writing to the corresponding Port C bit positions of the ACK and STB lines with the “Set Reset Port C Bit” command will affect the Group A and Group B interrupt enable flags, as illustrated in Figure 17.

INPUT CONFIGURATION



OUTPUT CONFIGURATION

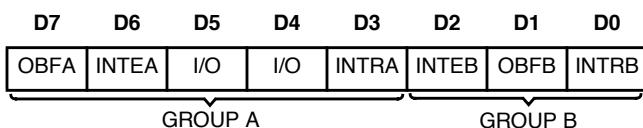


FIGURE 15. MODE 1 STATUS WORD FORMAT

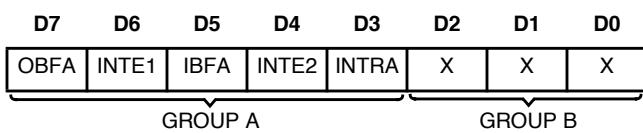


FIGURE 16. MORE 8 STATUS WORD FORMAT

Current Drive Capability

Any output on Port A, B or C can sink or source 2.5mA. This feature allows the 82C55A to directly drive Darlington type drivers and high-voltage displays that require such sink or source current.

Reading Port C Status (Figures 15 and 16)

In Mode 0, Port C transfers data to or from the peripheral device. When the 82C55A is programmed to function in Modes 1 or 2, Port C generates or accepts "hand shaking" signals with the peripheral device. Reading the contents of Port C allows the programmer to test or verify the "status" of each peripheral device and change the program flow accordingly.

There is not special instruction to read the status information from Port C. A normal read operation of Port C is executed to perform this function.

INTERRUPT ENABLE FLAG	POSITION	ALTERNATE PORT C PIN SIGNAL (MODE)
INTE B	PC2	ACKB (Output Mode 1) or STBB (Input Mode 1)
INTE A2	PC4	STBA (Input Mode 1 or Mode 2)
INTE A1	PC6	ACKA (Output Mode 1 or Mode 2)

FIGURE 17. INTERRUPT ENABLE FLAGS IN MODES 1 AND 2

Applications of the 82C55A

The 82C55A is a very powerful tool for interfacing peripheral equipment to the microcomputer system. It represents the optimum use of available pins and flexible enough to interface almost any I/O device without the need for additional external logic.

Each peripheral device in a microcomputer system usually has a "service routine" associated with it. The routine manages the software interface between the device and the CPU. The functional definition of the 82C55A is programmed by the I/O service routine and becomes an extension of the system software. By examining the I/O devices interface characteristics for both data transfer and timing, and matching this information to the examples and tables in the detailed operational description, a control word can easily be developed to initialize the 82C55A to exactly "fit" the application. Figures 18 through 24 present a few examples of typical applications of the 82C55A.

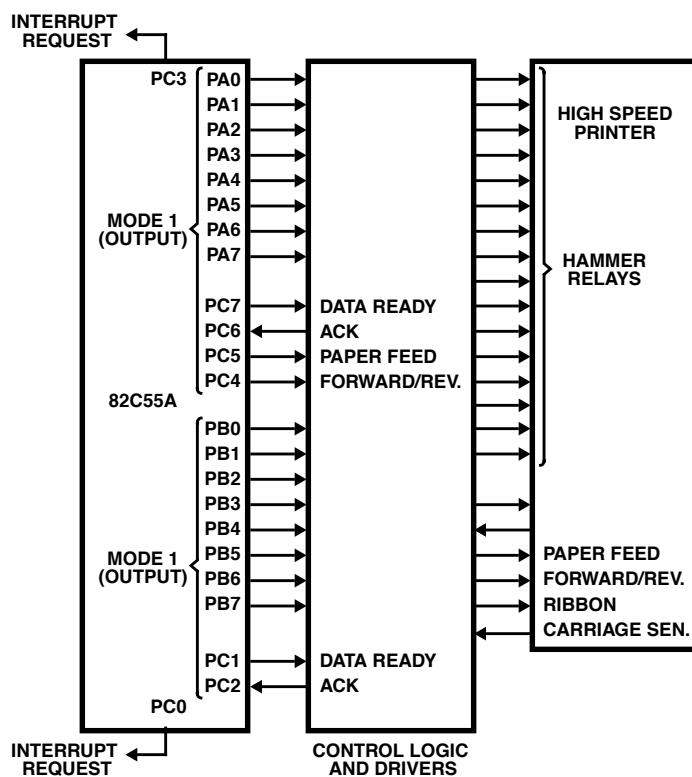
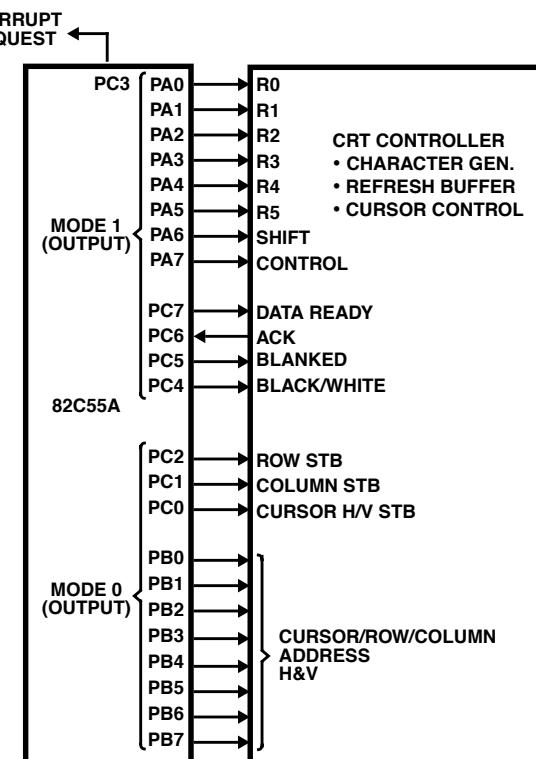
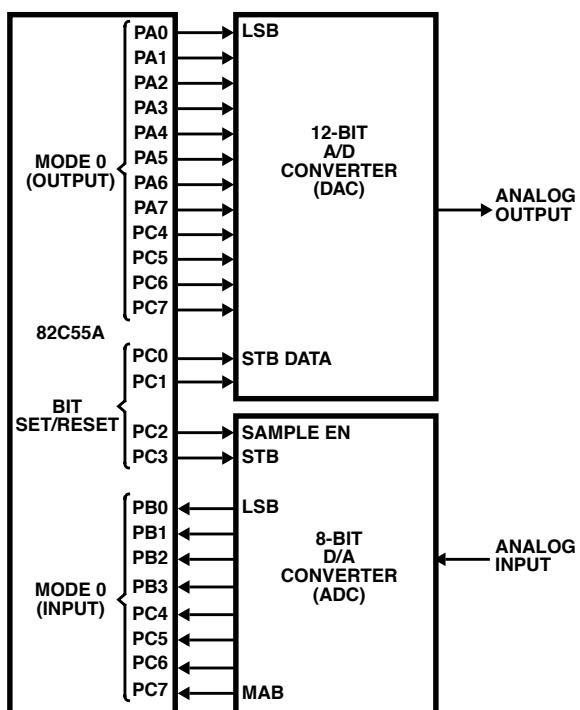
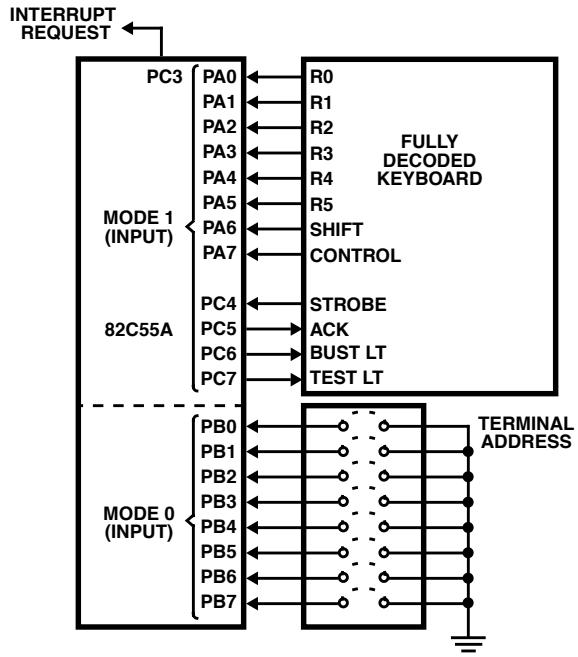
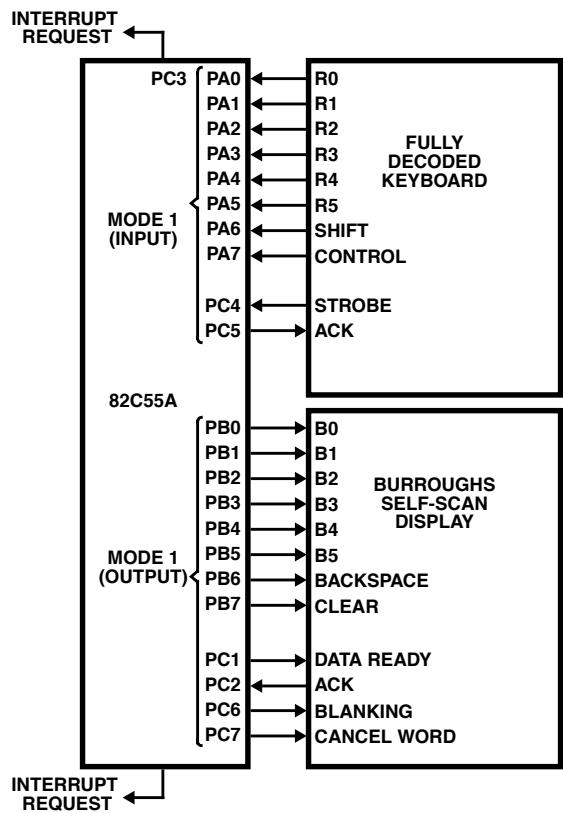


FIGURE 18. PRINTER INTERFACE

82C55A



82C55A

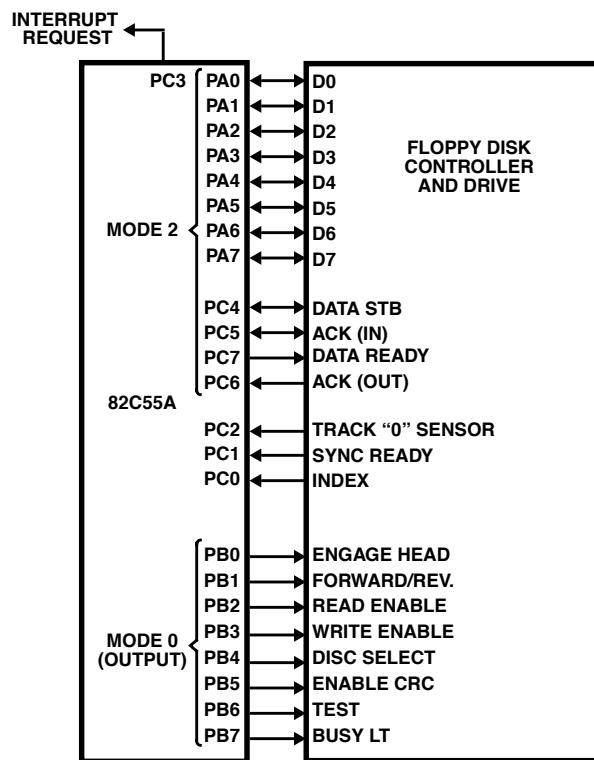


FIGURE 23. BASIC FLOPPY DISC INTERFACE

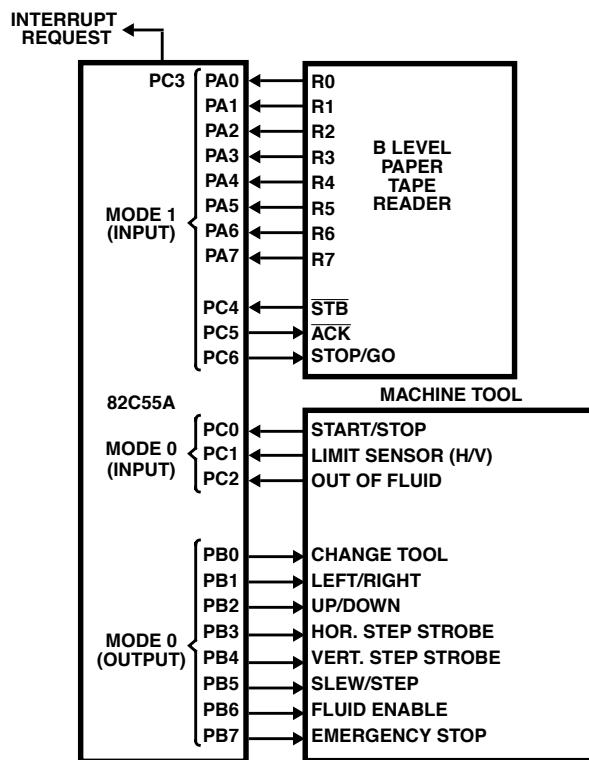


FIGURE 24. MACHINE TOOL CONTROLLER INTERFACE

Absolute Maximum Ratings $T_A = 25^\circ\text{C}$

Supply Voltage	+8.0V
Input, Output or I/O Voltage	GND-0.5V to $V_{CC}+0.5V$
ESD Classification	Class 1

Operating Conditions

Voltage Range	+4.5V to 5.5V
Operating Temperature Range	-40°C to 85°C
C82C55A	0°C to 70°C
I82C55A	-55°C to 125°C

Thermal Information

	θ_{JA}	θ_{JC}
CERDIP Package	50°C/W	10°C/W
CLCC Package	65°C/W	14°C/W
PDIP Package	50°C/W	N/A
PLCC Package	46°C/W	N/A
Maximum Storage Temperature Range	-65°C to 150°C	
Maximum Junction Temperature		
CDIP Package	175°C	
PDIP Package	150°C	
Maximum Lead Temperature (Soldering 10s)	300°C	(PLCC Lead Tips Only)

Die Characteristics

Gate Count	1000 Gates
------------------	------------

CAUTION: Stresses above those listed in "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress only rating and operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied.

NOTE:

1. θ_{JA} is measured with the component mounted on an evaluation PC board in free air.

Electrical Specifications $V_{CC} = 5.0V \pm 10\%$; $T_A = 0^\circ\text{C}$ to $+70^\circ\text{C}$ (C82C55A);
 $T_A = -40^\circ\text{C}$ to $+85^\circ\text{C}$ (I82C55A);
 $T_A = -55^\circ\text{C}$ to $+125^\circ\text{C}$ (M82C55A)

SYMBOL	PARAMETER	LIMITS		UNITS	TEST CONDITIONS
		MIN	MAX		
V_{IH}	Logical One Input Voltage	2.0 2.2	-	V	I82C55A, C82C55A, M82C55A
V_{IL}	Logical Zero Input Voltage	-	0.8	V	
V_{OH}	Logical One Output Voltage	3.0 $V_{CC}-0.4$	-	V	$I_{OH} = -2.5\text{mA}$, $I_{OH} = -100\mu\text{A}$
V_{OL}	Logical Zero Output Voltage	-	0.4	V	$I_{OL} + 2.5\text{mA}$
I_I	Input Leakage Current	-1.0	+1.0	μA	$V_{IN} = V_{CC}$ or GND, DIP Pins: 5, 6, 8, 9, 35, 36
I_O	I/O Pin Leakage Current	-10	+10	μA	$VO = V_{CC}$ or GND DIP Pins: 27 - 34
I_{BHH}	Bus Hold High Current	-50	-400	μA	$VO = 3.0\text{V}$. Ports A, B, C
I_{BHL}	Bus Hold Low Current	50	400	μA	$VO = 1.0\text{V}$. Port A ONLY
I_{DAR}	Darlington Drive Current	-2.5	Note 2, 4	mA	Ports A, B, C. Test Condition 3
I_{CCSB}	Standby Power Supply Current	-	10	μA	$V_{CC} = 5.5\text{V}$, $V_{IN} = V_{CC}$ or GND. Output Open
I_{CCOP}	Operating Power Supply Current	-	1	mA/MHz	$T_A = +25^\circ\text{C}$, $V_{CC} = 5.0\text{V}$, Typical (See Note 3)

NOTES:

2. No internal current limiting exists on Port Outputs. A resistor must be added externally to limit the current.
3. ICCOP = 1mA/MHz of Peripheral Read/Write cycle time. (Example: 1.0 μs I/O Read/Write cycle time = 1mA).
4. Tested as V_{OH} at -2.5mA.

Capacitance $T_A = 25^\circ\text{C}$

SYMBOL	PARAMETER	TYPICAL	UNITS	TEST CONDITIONS
C_{IN}	Input Capacitance	10	pF	FREQ = 1MHz, All Measurements are referenced to device GND
$C_{I/O}$	I/O Capacitance	20	pF	

82C55A

AC Electrical Specifications $V_{CC} = +5V \pm 10\%$, $GND = 0V$; $T_A = -55^{\circ}C$ to $+125^{\circ}C$ (M82C55A) (M82C55A-5);
 $T_A = -40^{\circ}C$ to $+85^{\circ}C$ (I82C55A) (I82C55A-5);
 $T_A = 0^{\circ}C$ to $+70^{\circ}C$ (C82C55A) (C82C55A-5)

SYMBOL	PARAMETER	82C55A-5		82C55A		UNITS	TEST CONDITIONS
		MIN	MAX	MIN	MAX		
READ TIMING							
(1) tAR	Address Stable Before \overline{RD}	0	-	0	-	ns	
(2) tRA	Address Stable After \overline{RD}	0	-	0	-	ns	
(3) tRR	\overline{RD} Pulse Width	250	-	150	-	ns	
(4) tRD	Data Valid From \overline{RD}	-	200	-	120	ns	1
(5) tDF	Data Float After \overline{RD}	10	75	10	75	ns	2
(6) tRV	Time Between \overline{RD} s and/or \overline{WR} s	300	-	300	-	ns	
WRITE TIMING							
(7) tAW	Address Stable Before \overline{WR}	0	-	0	-	ns	
(8) tWA	Address Stable After \overline{WR}	20	-	20	-	ns	
(9) tWW	\overline{WR} Pulse Width	100	-	100	-	ns	
(10) tDW	Data Valid to \overline{WR} High	100	-	100	-	ns	
(11) tWD	Data Valid After \overline{WR} High	30	-	30	-	ns	
OTHER TIMING							
(12) tWB	$\overline{WR} = 1$ to Output	-	350	-	350	ns	1
(13) tIR	Peripheral Data Before \overline{RD}	0	-	0	-	ns	
(14) tHR	Peripheral Data After \overline{RD}	0	-	0	-	ns	
(15) tAK	ACK Pulse Width	200	-	200	-	ns	
(16) tST	STB Pulse Width	100	-	100	-	ns	
(17) tPS	Peripheral Data Before STB High	20	-	20	-	ns	
(18) tPH	Peripheral Data After STB High	50	-	50	-	ns	
(19) tAD	ACK = 0 to Output	-	175	-	175	ns	1
(20) tKD	ACK = 1 to Output Float	20	250	20	250	ns	2
(21) tWOB	$\overline{WR} = 1$ to OBF = 0	-	150	-	150	ns	1
(22) tAOB	ACK = 0 to OBF = 1	-	150	-	150	ns	1
(23) tSIB	STB = 0 to IBF = 1	-	150	-	150	ns	1
(24) tRIB	$\overline{RD} = 1$ to IBF = 0	-	150	-	150	ns	1
(25) tTRIT	$\overline{RD} = 0$ to INTR = 0	-	200	-	200	ns	1
(26) tSIT	STB = 1 to INTR = 1	-	150	-	150	ns	1
(27) tAIT	ACK = 1 to INTR = 1	-	150	-	150	ns	1
(28) tWIT	$\overline{WR} = 0$ to INTR = 0	-	200	-	200	ns	1
(29) tRES	Reset Pulse Width	500	-	500	-	ns	1, (Note)

NOTE: Period of initial Reset pulse after power-on must be at least 50 μ sec. Subsequent Reset pulses may be 500ns minimum.

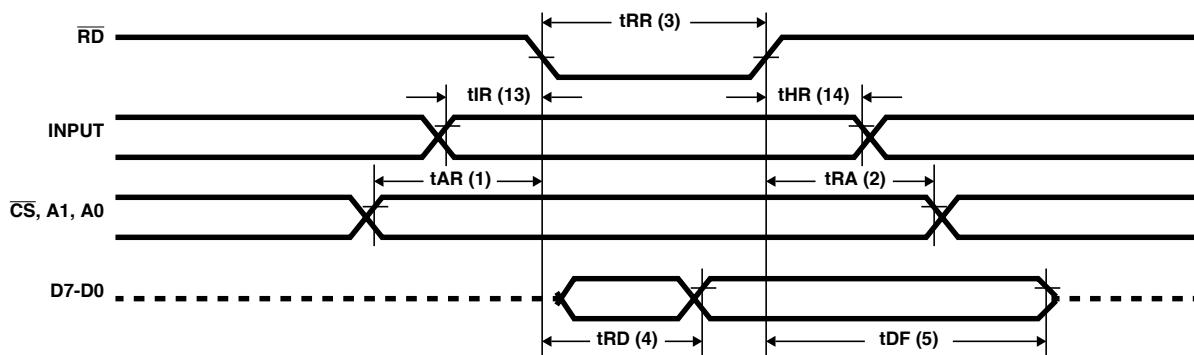
Timing Waveforms

FIGURE 25. MODE 0 (BASIC INPUT)

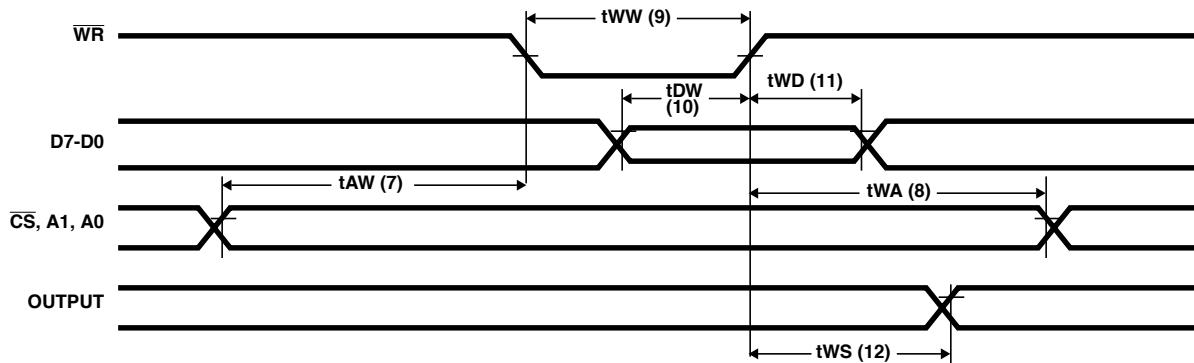


FIGURE 26. MODE 0 (BASIC OUTPUT)

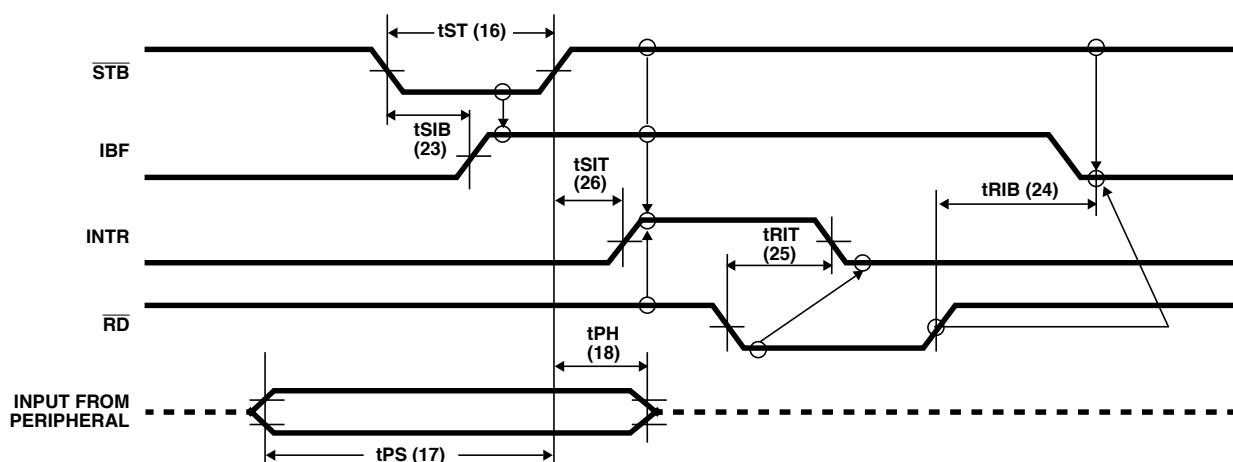


FIGURE 27. MODE 1 (STROBED INPUT)

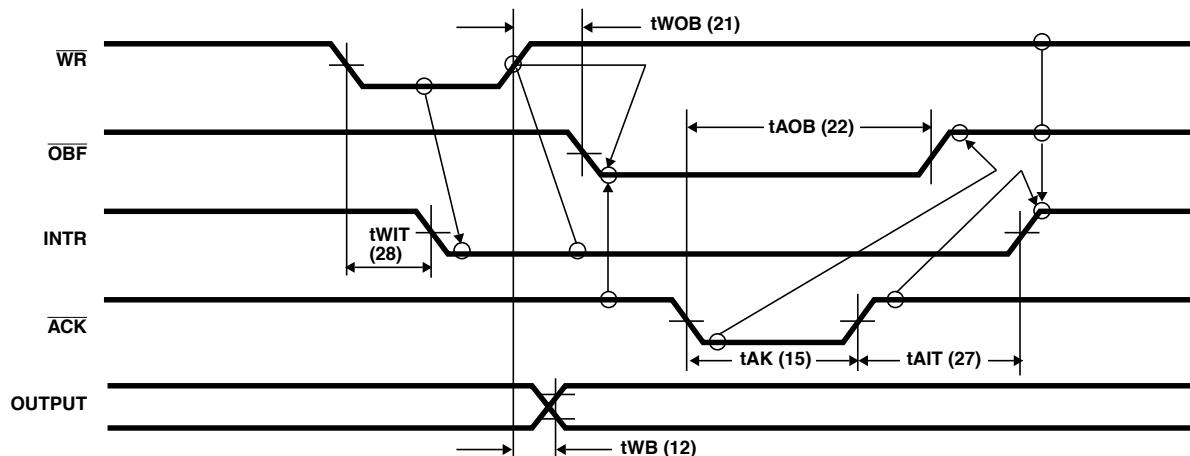
Timing Waveforms (Continued)

FIGURE 28. MODE 1 (STROBED OUTPUT)

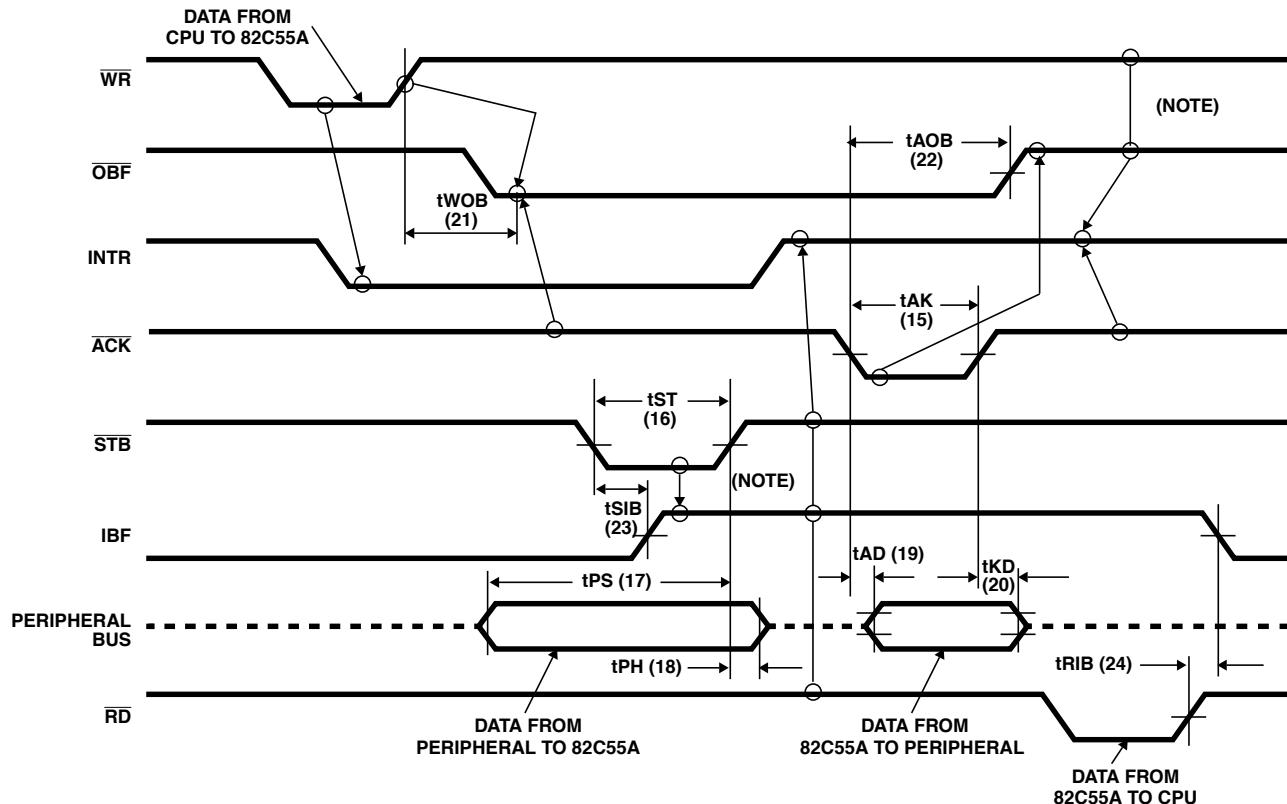


FIGURE 29. MODE 2 (BI-DIRECTIONAL)

NOTE: Any sequence where **WR** occurs before **ACK** and **STB** occurs before **RD** is permissible. (**INTR** = **IBF** • **MASK** • **STB** • **RD** • **OBF** • **MASK** • **ACK** • **WR**)

Timing Waveforms (Continued)

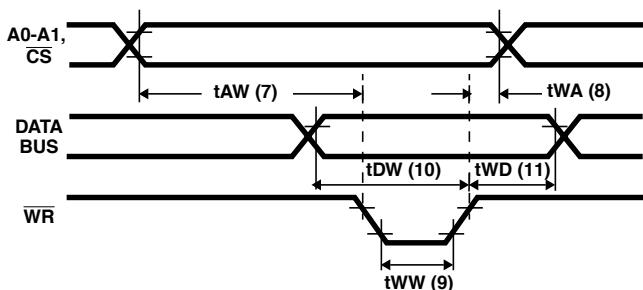


FIGURE 30. WRITE TIMING

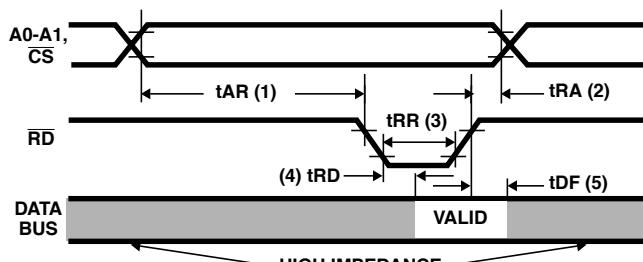
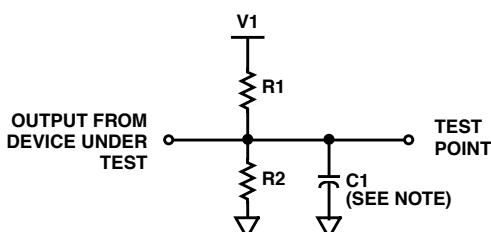


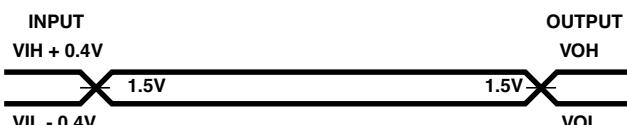
FIGURE 31. READ TIMING

AC Test Circuit



NOTE: Includes STRAY and JIG Capacitance

AC Testing Input, Output Waveforms



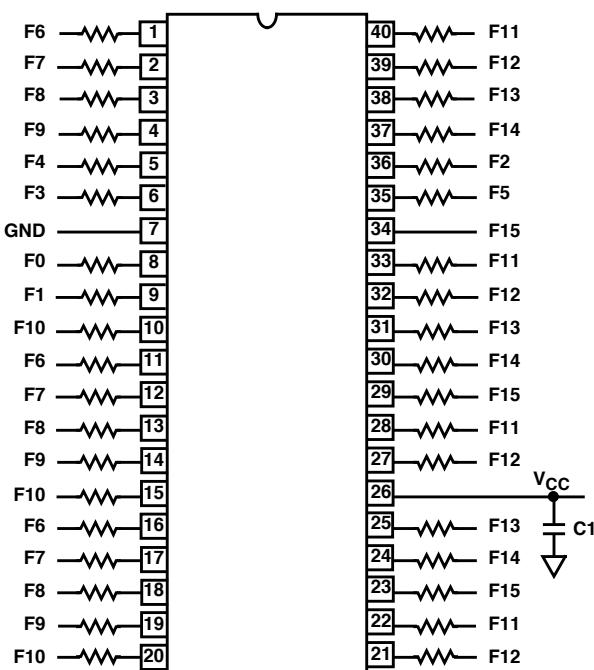
AC Testing: All AC Parameters tested as per test circuits. Input RISE and FALL times are driven at 1ns/V

TEST CONDITION DEFINITION TABLE

TEST CONDITION	V1	R1	R2	C1
1	1.7V	523Ω	Open	150pF
2	V_{CC}	2kΩ	1.7kΩ	50pF
3	1.5V	750Ω	Open	50pF

Burn-In Circuits

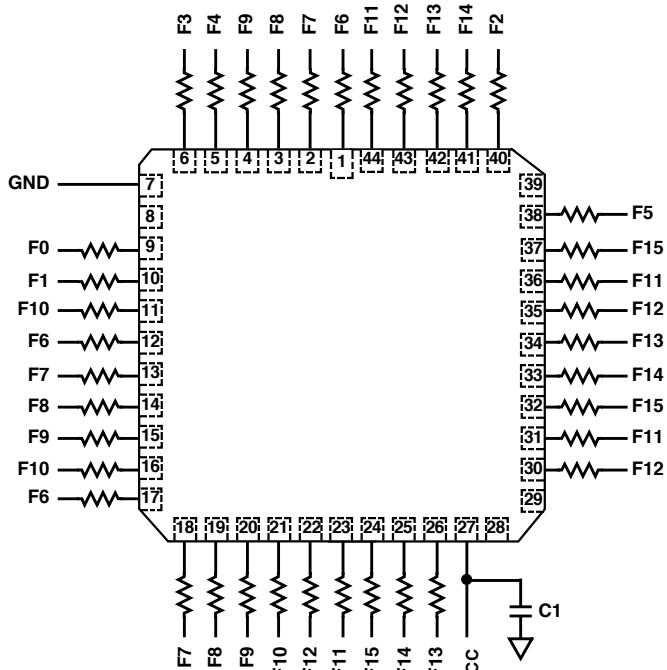
MD82C55A CERDIP



NOTES:

1. $V_{CC} = 5.5V \pm 0.5V$
 2. $VIH = 4.5V \pm 10\%$
 3. $VIL = -0.2V$ to $0.4V$
 4. $GND = 0V$

MR82C55A CLCC



NOTES:

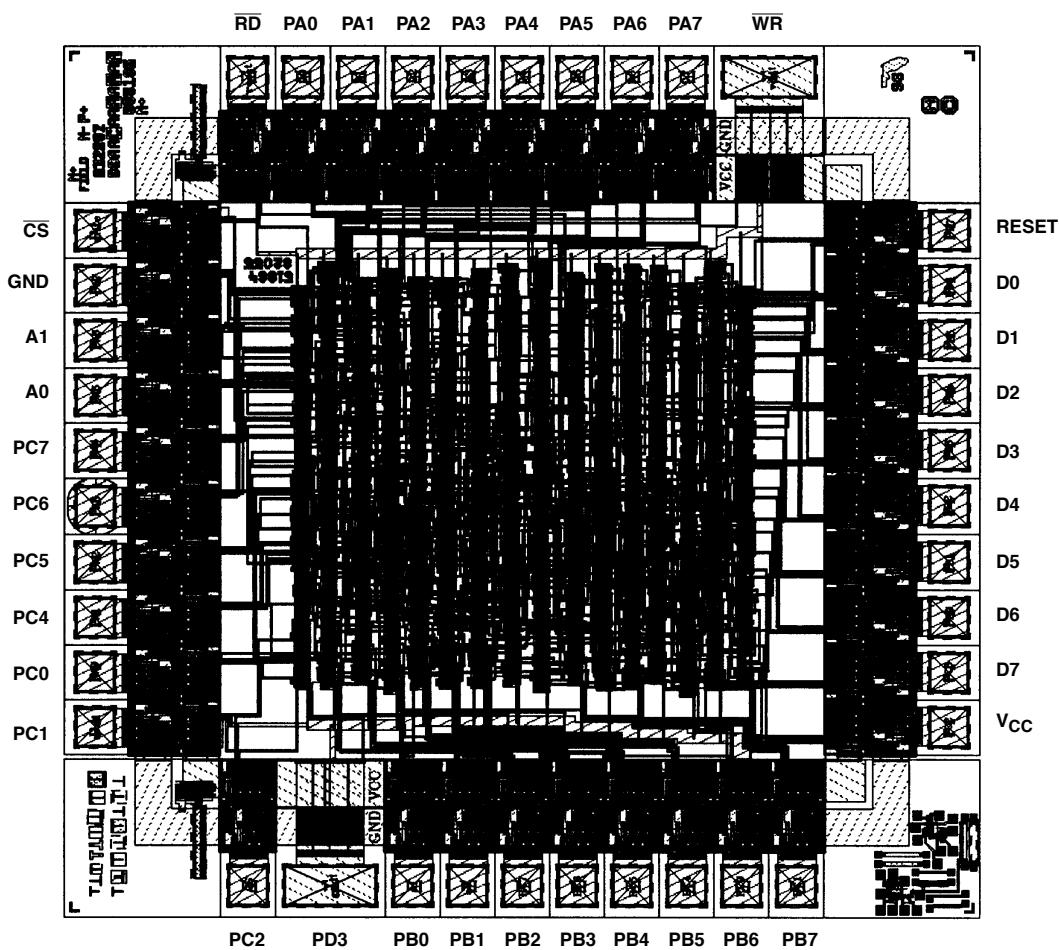
1. $C_1 = 0.01\mu F$ minimum
 2. All resistors are $47k\Omega \pm 5\%$
 3. $f_0 = 100\text{kHz} \pm 10\%$
 4. $f_1 = f_0 \div 2; f_2 = f_1 \div 2; \dots; f_{15} = f_{14} \div 2$

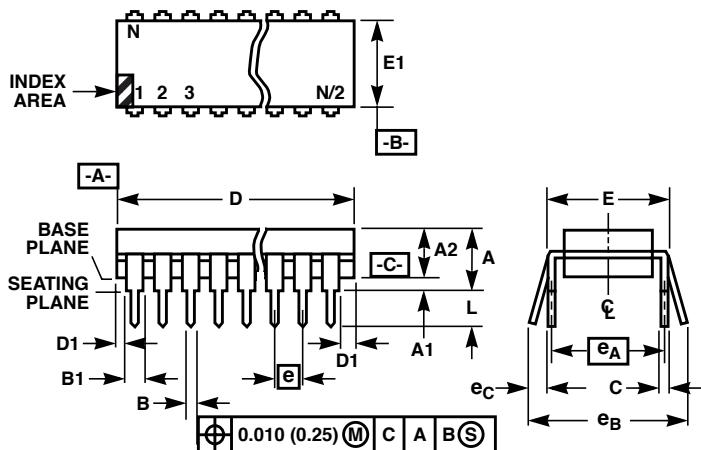
Die Characteristics**DIE DIMENSIONS:**95 x 100 x 19 \pm 1mils**METALLIZATION:**

Type: Silicon - Aluminum

Thickness: 11kÅ \pm 1kÅ**GLASSIVATION:**Type: SiO₂Thickness: 8kÅ \pm 1kÅ**WORST CASE CURRENT DENSITY:** 0.78×10^5 A/cm²***Metallization Mask Layout***

82C55A



Dual-In-Line Plastic Packages (PDIP)

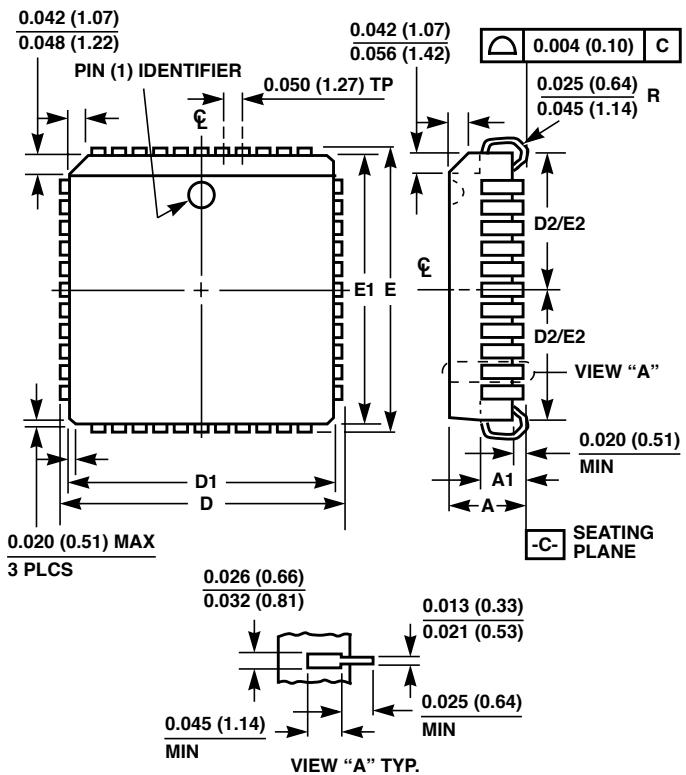
NOTES:

1. Controlling Dimensions: INCH. In case of conflict between English and Metric dimensions, the inch dimensions control.
2. Dimensioning and tolerancing per ANSI Y14.5M-1982.
3. Symbols are defined in the "MO Series Symbol List" in Section 2.2 of Publication No. 95.
4. Dimensions A, A1 and L are measured with the package seated in JEDEC seating plane gauge GS-3.
5. D, D1, and E1 dimensions do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.010 inch (0.25mm).
6. E and e_A are measured with the leads constrained to be perpendicular to datum $-C-$.
7. e_B and e_C are measured at the lead tips with the leads unconstrained. e_C must be zero or greater.
8. B1 maximum dimensions do not include dambar protrusions. Dambar protrusions shall not exceed 0.010 inch (0.25mm).
9. N is the maximum number of terminal positions.
10. Corner leads (1, N, N/2 and N/2 + 1) for E8.3, E16.3, E18.3, E28.3, E42.6 will have a B1 dimension of 0.030 - 0.045 inch (0.76 - 1.14mm).

**E40.6 (JEDEC MS-011-AC ISSUE B)
40 LEAD DUAL-IN-LINE PLASTIC PACKAGE**

SYMBOL	INCHES		MILLIMETERS		NOTES
	MIN	MAX	MIN	MAX	
A	-	0.250	-	6.35	4
A1	0.015	-	0.39	-	4
A2	0.125	0.195	3.18	4.95	-
B	0.014	0.022	0.356	0.558	-
B1	0.030	0.070	0.77	1.77	8
C	0.008	0.015	0.204	0.381	-
D	1.980	2.095	50.3	53.2	5
D1	0.005	-	0.13	-	5
E	0.600	0.625	15.24	15.87	6
E1	0.485	0.580	12.32	14.73	5
e	0.100 BSC		2.54 BSC		-
e_A	0.600 BSC		15.24 BSC		6
e_B	-	0.700	-	17.78	7
L	0.115	0.200	2.93	5.08	4
N	40		40		9

Rev. 0 12/93

Plastic Leaded Chip Carrier Packages (PLCC)

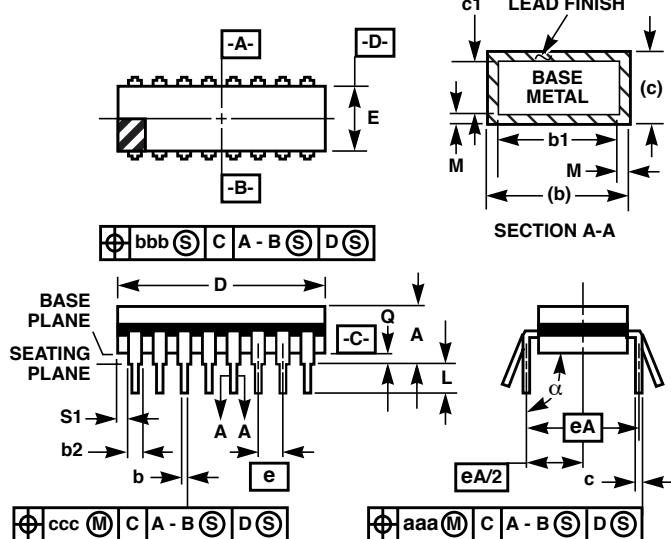
**N44.65 (JEDEC MS-018AC ISSUE A)
44 LEAD PLASTIC LEADED CHIP CARRIER PACKAGE**

SYM-BOL	INCHES		MILLIMETERS		NOTES
	MIN	MAX	MIN	MAX	
A	0.165	0.180	4.20	4.57	-
A1	0.090	0.120	2.29	3.04	-
D	0.685	0.695	17.40	17.65	-
D1	0.650	0.656	16.51	16.66	3
D2	0.291	0.319	7.40	8.10	4, 5
E	0.685	0.695	17.40	17.65	-
E1	0.650	0.656	16.51	16.66	3
E2	0.291	0.319	7.40	8.10	4, 5
N	44		44		6

Rev. 2 11/97

NOTES:

1. Controlling dimension: INCH. Converted millimeter dimensions are not necessarily exact.
2. Dimensions and tolerancing per ANSI Y14.5M-1982.
3. Dimensions D1 and E1 do not include mold protrusions. Allowable mold protrusion is 0.010 inch (0.25mm) per side. Dimensions D1 and E1 include mold mismatch and are measured at the extreme material condition at the body parting line.
4. To be measured at seating plane [-C-] contact point.
5. Centerline to be determined where center leads exit plastic body.
6. "N" is the number of terminal positions.

Ceramic Dual-In-Line Frit Seal Packages (CERDIP)

NOTES:

- Index area: A notch or a pin one identification mark shall be located adjacent to pin one and shall be located within the shaded area shown. The manufacturer's identification shall not be used as a pin one identification mark.
- The maximum limits of lead dimensions b and c or M shall be measured at the centroid of the finished lead surfaces, when solder dip or tin plate lead finish is applied.
- Dimensions b1 and c1 apply to lead base metal only. Dimension M applies to lead plating and finish thickness.
- Corner leads (1, N, N/2, and N/2+1) may be configured with a partial lead paddle. For this configuration dimension b3 replaces dimension b2.
- This dimension allows for off-center lid, meniscus, and glass overrun.
- Dimension Q shall be measured from the seating plane to the base plane.
- Measure dimension S1 at all four corners.
- N is the maximum number of terminal positions.
- Dimensioning and tolerancing per ANSI Y14.5M - 1982.
- Controlling dimension: INCH.

**F40.6 MIL-STD-1835 GDIP1-T40 (D-5, CONFIGURATION A)
40 LEAD CERAMIC DUAL-IN-LINE FRIT SEAL PACKAGE**

SYMBOL	INCHES		MILLIMETERS		NOTES
	MIN	MAX	MIN	MAX	
A	-	0.225	-	5.72	-
b	0.014	0.026	0.36	0.66	2
b1	0.014	0.023	0.36	0.58	3
b2	0.045	0.065	1.14	1.65	-
b3	0.023	0.045	0.58	1.14	4
c	0.008	0.018	0.20	0.46	2
c1	0.008	0.015	0.20	0.38	3
D	-	2.096	-	53.24	5
E	0.510	0.620	12.95	15.75	5
e	0.100 BSC		2.54 BSC		-
eA	0.600 BSC		15.24 BSC		-
eA/2	0.300 BSC		7.62 BSC		-
L	0.125	0.200	3.18	5.08	-
Q	0.015	0.070	0.38	1.78	6
S1	0.005	-	0.13	-	7
α	90°	105°	90°	105°	-
aaa	-	0.015	-	0.38	-
bbb	-	0.030	-	0.76	-
ccc	-	0.010	-	0.25	-
M	-	0.0015	-	0.038	2,3
N	40		40		8

Rev. 0 4/94

All Intersil semiconductor products are manufactured, assembled and tested under **ISO9000** quality systems certification.

Intersil products are sold by description only. Intersil Corporation reserves the right to make changes in circuit design and/or specifications at any time without notice. Accordingly, the reader is cautioned to verify that data sheets are current before placing orders. Information furnished by Intersil is believed to be accurate and reliable. However, no responsibility is assumed by Intersil or its subsidiaries for its use; nor for any infringements of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of Intersil or its subsidiaries.

For information regarding Intersil Corporation and its products, see web site <http://www.intersil.com>

Sales Office Headquarters**NORTH AMERICA**

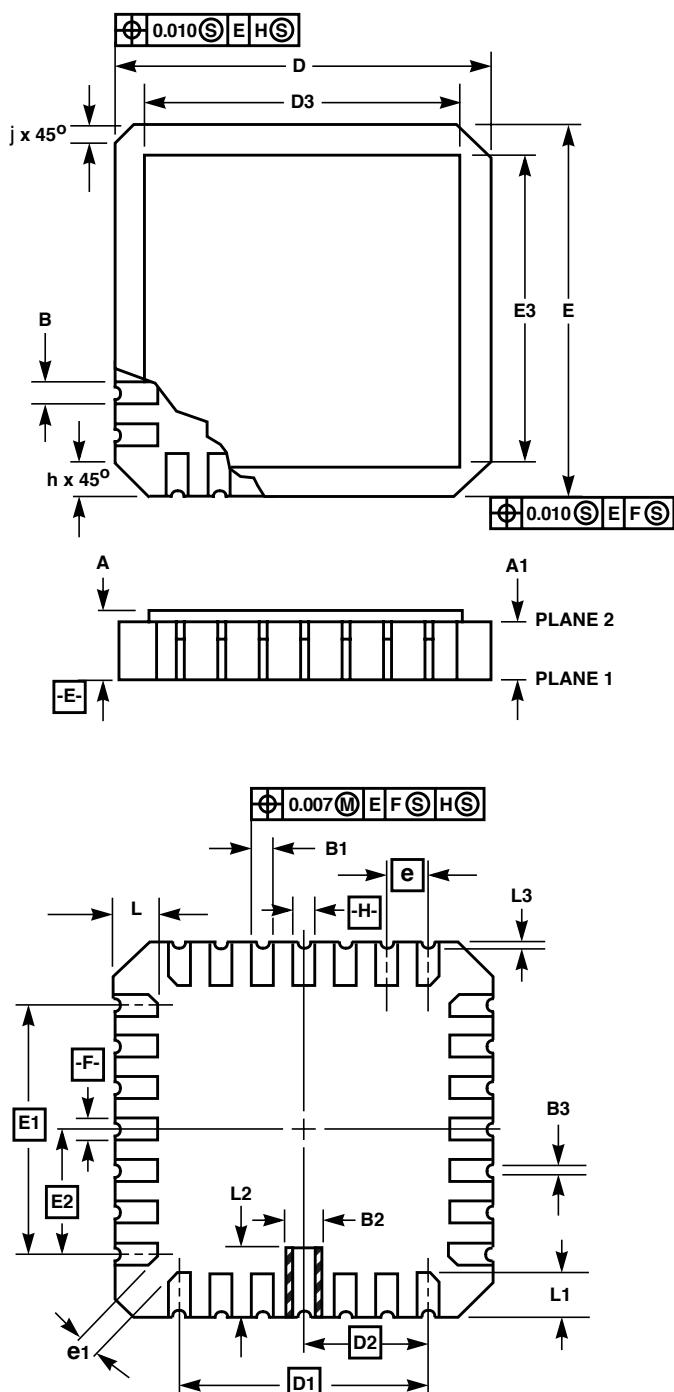
Intersil Corporation
P. O. Box 883, Mail Stop 53-204
Melbourne, FL 32902
TEL: (407) 724-7000
FAX: (407) 724-7240

EUROPE

Intersil SA
Mercure Center
100, Rue de la Fusée
1130 Brussels, Belgium
TEL: (32) 2.724.2111
FAX: (32) 2.724.22.05

ASIA

Intersil (Taiwan) Ltd.
Taiwan Limited
7F-6, No. 101 Fu Hsing North Road
Taipei, Taiwan
Republic of China
TEL: (886) 2 2716 9310
FAX: (886) 2 2715 3029

Ceramic Leadless Chip Carrier Packages (CLCC)

**J44.A MIL-STD-1835 CQCC1-N44 (C-5)
44 PAD CERAMIC LEADLESS CHIP CARRIER PACKAGE**

SYMBOL	INCHES		MILLIMETERS		NOTES
	MIN	MAX	MIN	MAX	
A	0.064	0.120	1.63	3.05	6, 7
A1	0.054	0.088	1.37	2.24	-
B	0.033	0.039	0.84	0.99	4
B1	0.022	0.028	0.56	0.71	2, 4
B2	0.072 REF		1.83 REF		-
B3	0.006	0.022	0.15	0.56	-
D	0.640	0.662	16.26	16.81	-
D1	0.500 BSC		12.70 BSC		-
D2	0.250 BSC		6.35 BSC		-
D3	-	0.662	-	16.81	2
E	0.640	0.662	16.26	16.81	-
E1	0.500 BSC		12.70 BSC		-
E2	0.250 BSC		6.35 BSC		-
E3	-	0.662	-	16.81	2
e	0.050 BSC		1.27 BSC		-
e1	0.015	-	0.38	-	2
h	0.040 REF		1.02 REF		5
j	0.020 REF		0.51 REF		5
L	0.045	0.055	1.14	1.40	-
L1	0.045	0.055	1.14	1.40	-
L2	0.075	0.095	1.90	2.41	-
L3	0.003	0.015	0.08	0.38	-
ND	11		11		3
NE	11		11		3
N	44		44		3

Rev. 0 5/18/94

NOTES:

1. Metallized castellations shall be connected to plane 1 terminals and extend toward plane 2 across at least two layers of ceramic or completely across all of the ceramic layers to make electrical connection with the optional plane 2 terminals.
2. Unless otherwise specified, a minimum clearance of 0.015 inch (0.38mm) shall be maintained between all metallized features (e.g., lid, castellations, terminals, thermal pads, etc.)
3. Symbol "N" is the maximum number of terminals. Symbols "ND" and "NE" are the number of terminals along the sides of length "D" and "E", respectively.
4. The required plane 1 terminals and optional plane 2 terminals (if used) shall be electrically connected.
5. The corner shape (square, notch, radius, etc.) may vary at the manufacturer's option, from that shown on the drawing.
6. Chip carriers shall be constructed of a minimum of two ceramic layers.
7. Dimension "A" controls the overall package thickness. The maximum "A" dimension is package height before being solder dipped.
8. Dimensioning and tolerancing per ANSI Y14.5M-1982.
9. Controlling dimension: INCH.

Interfaz paralela programable 8255

INTERFAZ DE PERIFERICOS PROGRAMABLE 8255A-5.

La *interfaz de periféricos programable 8255A-5 (PPI)* es un componente de bajo costo muy popular encontrado en muchas aplicaciones. El PPI tiene 24 terminales para E/S, programables en grupos de 12, que se emplea en tres modos separados de operación. El 8255A-5 puede interfazar cualquier dispositivo de E/S compatible TTL, al microprocesador 8088. El 8255-5 requiere de la inserción de dos estados de espera (para un 8088 de 8MHz) cada vez que se activa por un comando IN u OUT. Debido a que los dispositivos de E/S son inherentemente mas lentos de cualquier forma, los estados de espera utilizados durante las transferencias de E/S no afectan la velocidad del sistema. El 8255-5 todavía encuentra aplicaciones (compatibles para programación, sin embargo, puede no aparecer en el sistema) aun en los últimos sistemas de computadoras basados en μ P 80486. El 8255-5 se emplea para interfaz del teclado y el puerto paralelo de la impresora en las computadoras personales.

Descripción Básica del 8255A-5.

La Figura 1 (a) ilustra el diagrama de los terminales de salida del 8255A-5 y la Figura 1 (b) muestra el diagrama interno. Sus tres puertos de E/S (etiquetados A, B y C) se programan en grupos de 12 terminales. Las conexiones del grupo A consisten del Puerto A (PA₇-PA₀) y la mitad superior del puerto C (PC₇-PC₄), y el grupo B consiste del Puerto B (PB₇-PB₀) y la mitad inferior del Puerto C (PC₃-PC₀).

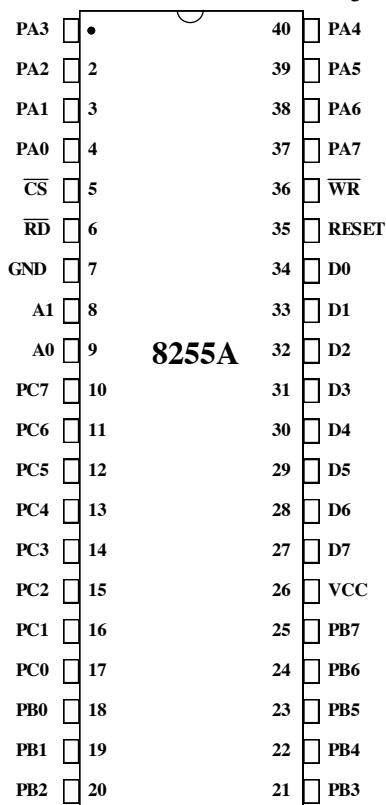


Figura 1 (a) Terminales del 8255A Interfaz Periférico Programable (PPI).

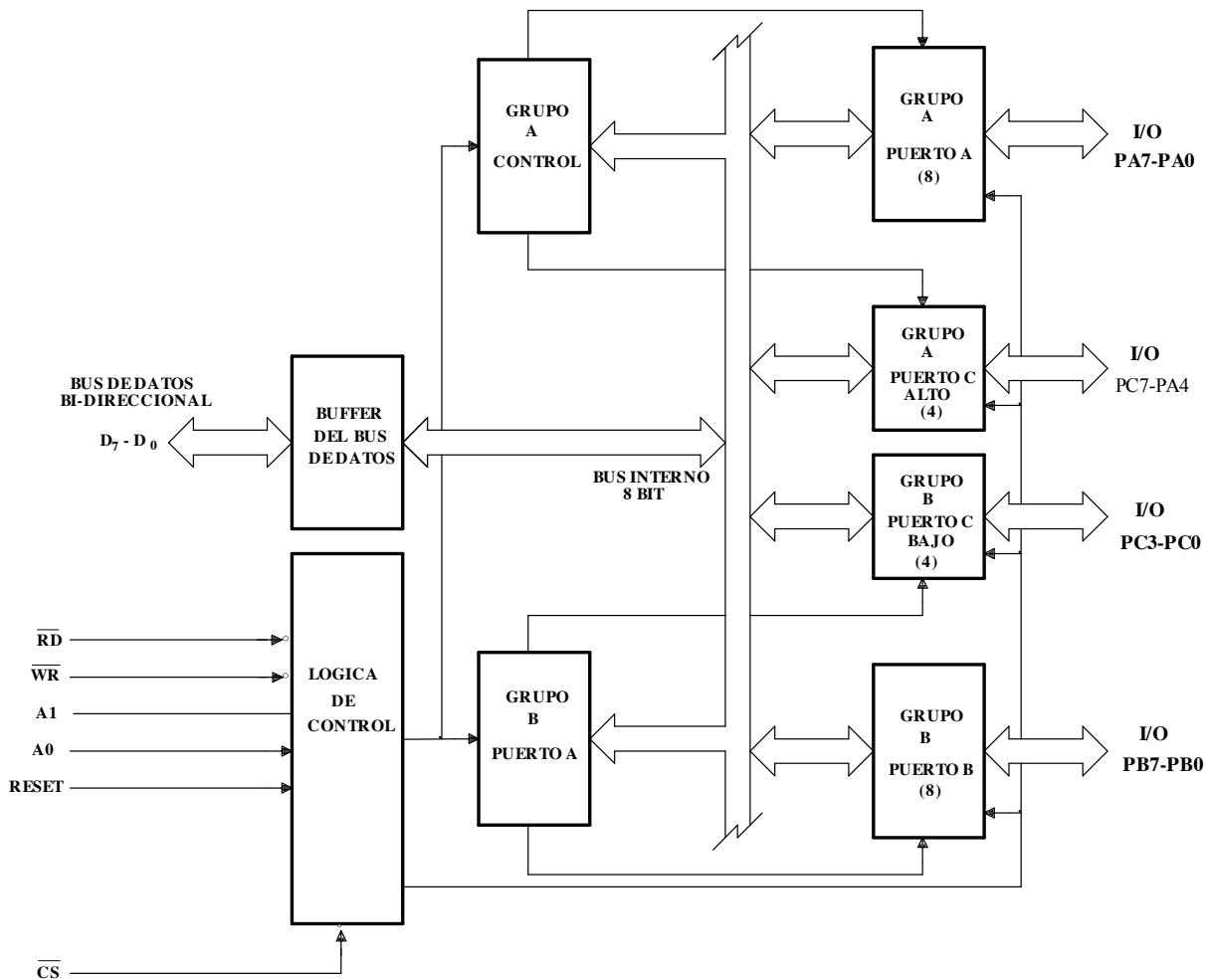


Figura 1 (b) Diagrama de bloques.

El 8255A-5 se selecciona por su terminal **CS** para programación y lectura o escritura a un puerto. La selección del registro es realizada a través de la terminales **A₁** y **A₀** que seleccionan un registro interno para operaciones de programación.

Tabla 1 Asignaciones de puerto de E/S para el 8255A-5

A ₁	A ₀	Función
0	0	Puerto A
0	1	Puerto B
1	0	Puerto C
1	1	Registro de Comando

La Tabla 1 muestra las asignaciones de los puertos de E/S usados para programación y acceso a los puertos de E/S. En la computadora personal, un 8255A o su equivalente es decodificado en los puertos de E/S 60H-63H.

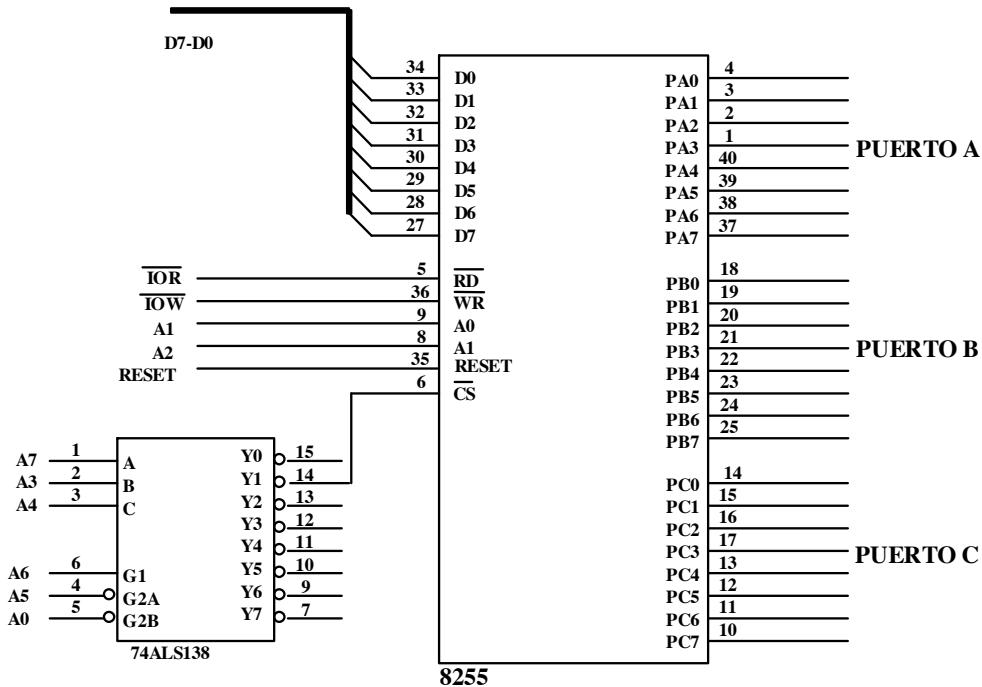


Figura 2 El 8255A interfazado al banco bajo del microprocesador 8088.

El 8255A-5 es un dispositivo bastante simple de interfazar al 8088 y programarlo. Para que el 8255A-5 pueda ser leído o grabado, la señal \overline{CS} debe ser un 0 lógico y la dirección de E/S correcta debe ser aplicada a los terminales A₁-A₀. Los terminales de direcciones del puerto restantes no importan y son externamente decodificados para seleccionar el 8255A-5.

La Figura 2 muestra un 8255A-5 conectado al 8088, de tal forma que trabaja con puertos de E/S de direcciones de 8 bit C0H (puerto A), C2H (puerto B), C4H (puerto C), y C6H (registro de comandos). Notar en esta interfaz que todos los terminales del 8255A-5 son conexiones directas al 8088 excepto por la terminal \overline{CS} . La terminal se \overline{CS} decodifica y seleccionada por un decodificador 74ALS138.

La entrada de RESET del 8255A-5, inicializa el dispositivo cada vez que el microprocesador es reinicializado. Una entrada de RESET en el 8255A-5, ocasiona que todos los puertos sean puestos como puertos simples usando el modo 0 de operación. Debido a que los terminales del puerto se programan internamente como terminales de entrada en un restablecimiento, esto evita daños cuando la alimentación es aplicada al sistema. Después de una señal de RESET, ningún otro comando es necesario para programar el 8255A-5 mientras se deba programar como un dispositivo de entrada en los tres puertos. Notar que un 8255A-5 se interfiere a una computadora personal en las direcciones puerto 60H-63H para control del teclado y también para controlar la bocina, el temporizador y otros dispositivos internos como la expansión de memoria. El 8255A-5 es fácil de programar debido a que solo contiene dos registros de comandos internos como se muestra en la Figura 3. Note que el bit de la posición 7 selecciona el byte de control A o el byte de control B.

Programación de 8255A-5

El byte de control A programa la función del grupo A y B, mientras que el byte de control B pone (1) o reinicializa (0) los bit del puerto C solo si el 8255A-5 se programa en el modo 1 o 2. Los terminales del grupo B (puerto B y la parte baja del puerto C) se programan como terminales de entrada o salida. El grupo B puede operar en el modo 0 o en el modo 1. El modo 0 es el modo básico de E/S que permite que los terminales del grupo B sean programados como simples entradas y conexiones de salida amarradas. La operación del modo 1 es la operación de habilitación periódica para las conexiones del grupo B donde los datos se transfieren a través del puerto B, y el puerto C proporciona las señales de reconocimiento.

Los terminales del grupo A (puerto A y la parte superior del puerto C) son también programadas como terminales de entrada o de salida. La diferencia es que este grupo A puede operar en los modos 0, 1 y 2. La función del modo 2 es operar en modo bidireccional para el puerto A.

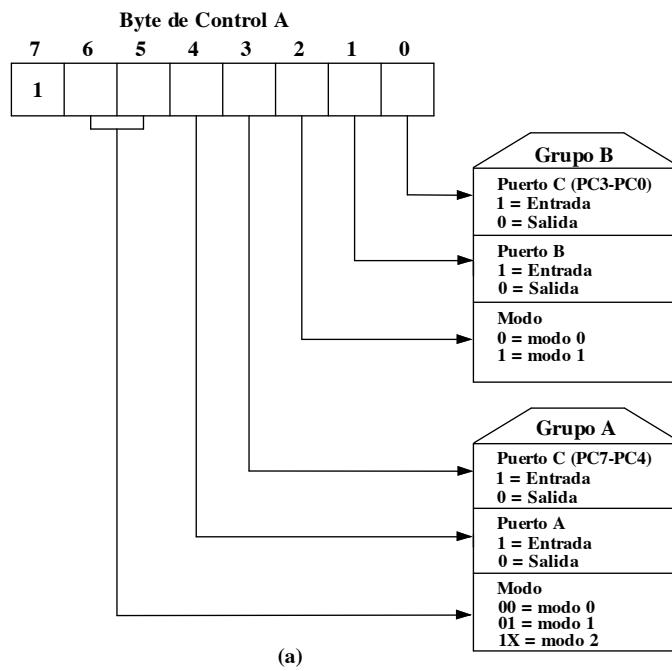


Figura 3 (a) El byte de control para el registro de control del 8255A.

Si se pone un 0 en la posición del bit 7 del byte de control, se selecciona el byte de control B. Este control permite que cualquier bit del puerto C sea puesto (1) o reinicializado (0) si el 8255A-5 opera en el modo 1 o en el modo 2; de otra forma, este byte de control no se emplea para programación. Nosotros frecuentemente usamos la función de activar/borrar un bit en sistemas de control para poner o limpiar un bit de control en el puerto C.

Operación del MODO 0.

La operación del Modo 0 ocasiona que el 8255A-5 funcione como una entrada reforzada o como una salida amarrada. La figura 4 muestra el 8255A-5 conectado a un arreglo de

ocho indicadores de 7 segmentos. En este circuito, los puertos A y B se programan (modo 0) como simples puertos de salida con amarradores.

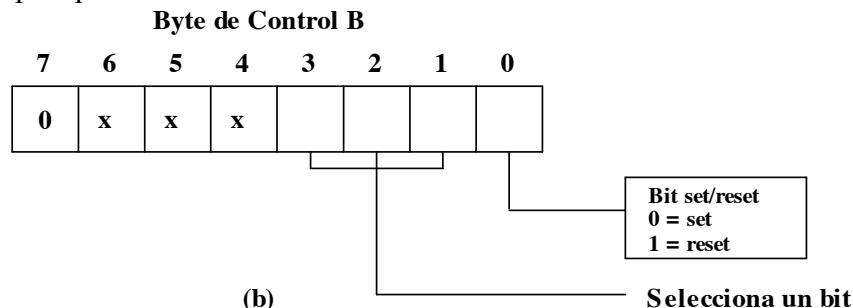


Figura 3 (b) Pone a 1 o borra el bit indicado en el campo del bit seleccionado.

El puerto A provee las entradas de datos a los segmentos del indicador y el puerto B selecciona una posición a la vez por medio de la multicanalización de los indicadores. El 8255A-5 es interfazado al 8088 a través de un decodificador de direcciones que opera en las direcciones de puertos de E/S.

Los valores de las resistencias son elegidos en la Figura 4, en base a la corriente de segmento que es 80 mA. Esta corriente es requerida para producir una corriente promedio de 10 mA por segmento cuando los indicadores son multicanalizados. En este tipo de sistemas de indicador, solo una de las ocho posiciones del indicador esta encendida en cualquier instante dado. la corriente pico de ánodo es de 560 mA, pero la corriente promedio de ánodo es de 70 mA. Cada vez que los indicadores son multicanalizados, incrementamos la corriente de segmento desde 10 mA hasta un valor igual a 'numero de posiciones de indicador' veces 10 mA. Esto significa que un indicador de 4 dígitos utiliza 40 mA por segmento, un indicador de 5 dígitos utiliza 50 mA, y así sucesivamente.

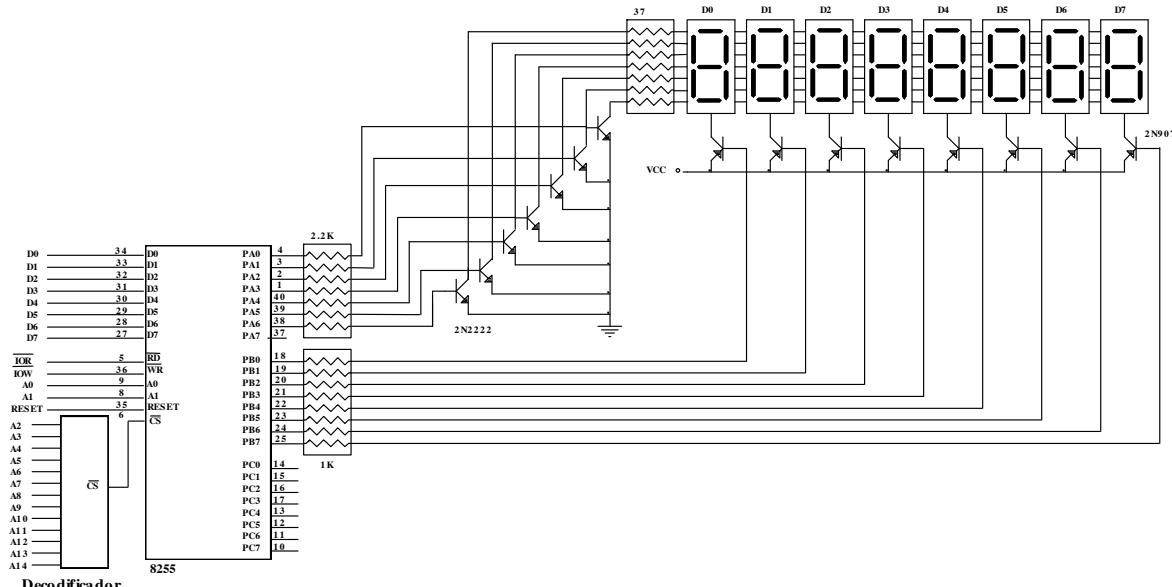


Figura 4 Conjunto de 8 despegados de 7 segmentos interfazado al 8088 mediante un 8255A.

Antes de que el software para operar el indicador sea examinado, debemos primero programar el 8255A-5. Esto se realiza una secuencia de instrucciones listadas en el Ejemplo 1. Aquí, los puertos A y B se programan como salidas.

Ejemplo 1

; software para programar el 8255A

MOV AL,1000000B	;byte de control
MOV DX,COMMAND	;direccionar registro de control
OUT DX,AL	;manda byte de control al 8255A

El procedimiento para manejar estos indicadores es listado en el Ejemplo 2. Para que este sistema de indicador funcione correctamente, debemos llamar a este procedimiento frecuentemente. Notar que el procedimiento llama a otro procedimiento (DELAY) que causa un retraso de 1ms.

Ejemplo 2

;procedimiento para el rastreo de 8 dígitos del indicadores multicanalizados.
;Este procedimiento debe ser llamado continuamente desde un programa
;para desplegar la información codificada en 7 segmentos en el área
MEMORY.

```

;
DISP      PROC NEAR

    PUSHF           ;salvar registros
    PUSH AX
    PUSH BX
    PUSH DX
    PUSH SI

    ;coloca registros para el indicador

    MOV BX,8          ;cargar cuenta
    MOV AH,7FH        ;cargar patrón elegido
    MOV SI,OFFSET MEMORY-1 ;direccionar indicador RAM
    MOV DX,PORTB      ;direccionar puerto B

    ;despliega ocho dígitos

DISP1:   MOV AL,AH          ;seleccionar dígito
          OUT DX,AL
          DEC DX            ;direccionar puerto A
          MOV AL,[BX+SI]     ;obtener dato
          OUT DX,AL         ;desplegar dato

          CALL DELAY        ;esperar 1 ms

          ROR AH,1          ;ajustar código de selección

```

INC DX	;direccionar puerto B
DEC BX	;ajustar cuenta
JNZ DISP1	;repetir 8 veces
POP SI	;recuperar registros
POP DX	
POP BX	
POP AX	
POPF	
RET	
DISP	ENDP

Este tiempo de retraso no se ilustra en este ejemplo, pero se emplea para permitir el tiempo necesario para que cada posición se encienda. Es recomendado por los fabricantes de indicadores LED que la iluminación del indicador sea entre 100 Hz y 1,500 Hz. Usando un retraso de tiempo de 1 ms, iluminamos cada dígito durante 1 ms para una velocidad de iluminación de 1,000 Hz/8 o 125.

El procedimiento del indicador direcciona una área de memoria donde los datos, en código de 7 segmento, son almacenados por los ocho dígitos del indicador. El registro AH es cargado con un código (7FH) que inicialmente direcciona la posición mas significante del indicador. Una vez que se selecciona la posición del indicador, se direcciona el contenido de la localidad de memoria MEMORY+7 y se envía al dígito mas significativo. El código de selección se ajusta para seleccionar el siguiente dígito del indicador, como en la memoria.

Este proceso se repite ocho veces para desplegar el contenido de la locación MEMORY hasta MEMORY+7 en los 8 dígitos del indicador.

MODO 1 Entrada con habilitación periódica

Este modo de operación ocasiona que el puerto A y/o el puerto B funcionen como dispositivos de entrada con amarradores. Esto permite que los datos externos se almacenen en el puerto hasta que el microprocesador este listo para almacenarlos. El puerto C se emplea en la operación de modo 1, no para datos, sino para señales de control o reconocimiento que ayudan a operar a uno o ambos puertos A y B como entrada con puertos de habilitación periódica. La Figura 5 muestra ambos diagramas de tiempo y como ambos puertos son estructurados para el modo 1 de operación entrada con habilitación periódica.

El puerto con entrada de habilitación periódica captura datos de la terminal del puerto cuando se activa la señal de habilitación periódica (STB). Notar que la señal de habilitación periódica captura los datos del puerto en la transición de 0 a 1. La señal (STB) ocasiona que se capturen los datos en el puerto y también activa las señales IBF (input buffer full) y INTR (interrup request). Una vez que el microprocesador advierte

que los datos están presentes en el puerto, a través del software (IBF) o del hardware (INTR), ejecuta una instrucción IN para leer el puerto (\overline{RD}). La acción de lectura del puerto restaura a IBF y a INTR a sus estados inactivos hasta que el próximo dato es recibido en el puerto.

Definiciones de Señales para Modo 1 Entrada con habilitación periódica.

1. STB - Señal de habilitación periódica: Una entrada usada para cargar datos dentro del amarrador del puerto, el cual retiene la información hasta que se lee por el microprocesador a través de una instrucción IN.
2. IBF - Reforzador de Entrada Lleno: Una salida que indica que el amarrador de entrada contiene información.
3. INTR - Requerimiento de Interrupción: Una salida que requiere una interrupción. La terminal INTR cambia a 1 lógico cuando la señal STB regresa a 1 lógico y es limpiada cuando el dato es metido desde el puerto por el microprocesador.
4. INT - Habilitador de Interrupción: Un bit interno programado a través del puerto PC₄ (puerto A) o una posición de bit en PC₂ (puerto B).
5. PC₇, PC₆ - Terminales 7 y 6 del Puerto: Terminales de E/S de propósito general que son disponibles para cualquier propósito.

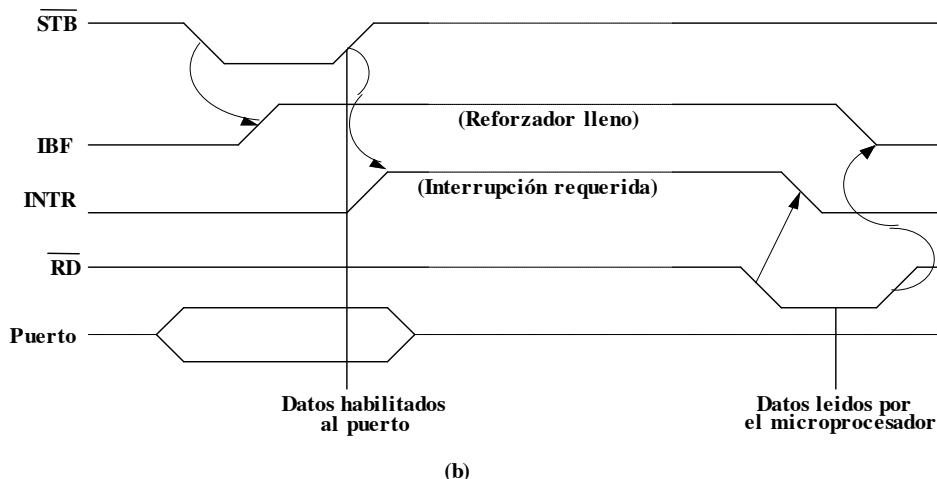
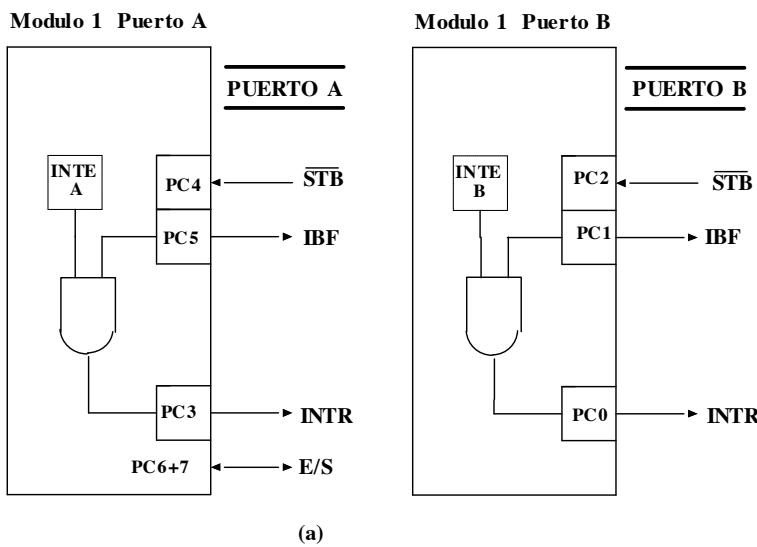


Figura 5 Operación de habilitación periódica de entrada (modo 1) del 8255A. (a) Estructura interna. (b) Diagrama de tiempos.

Ejemplo de Señal de Habilitación Periódica de Entrada. Un teclado es un excelente ejemplo de dispositivo de entrada de habilitación periódica. El codificador del teclado elimina las interferencias de los interruptores de las teclas y proporciona una señal de habilitación periódica cuando se presiona cada tecla, y el dato de salida contiene la codificación ASCII de la tecla.

La Figura 6 ilustra un teclado conectado para enviar una señal de habilitación periódica al puerto A. Aquí, \overline{DAV} (data available) se activa por 1 μs cada vez que una tecla se presiona en el teclado. Esto genera datos a ser enviados en forma de habilitación periódica al puerto A debido a que \overline{DAV} se conecta a la entrada \overline{STB} del puerto A. Entonces, cada vez que una tecla se presiona, se almacena en el puerto A del 8255A-5. La entrada \overline{STB} también activa la señal IBF, indicando que los datos están en el puerto A.

El ejemplo 3 muestra un procedimiento que lee datos desde el teclado cada vez que una tecla se presiona. Este procedimiento lee la tecla desde el puerto A y regresa el código ASCII en AL. Para detectar una tecla, se lee el puerto C y el bit IBF (posición del bit PC_5) se prueba para ver si el reforzador esta lleno. Si el reforzador esta vacío ($IBF=0$), entonces el procedimiento se mantiene probando este bit, esperando que un carácter sea tecleado.

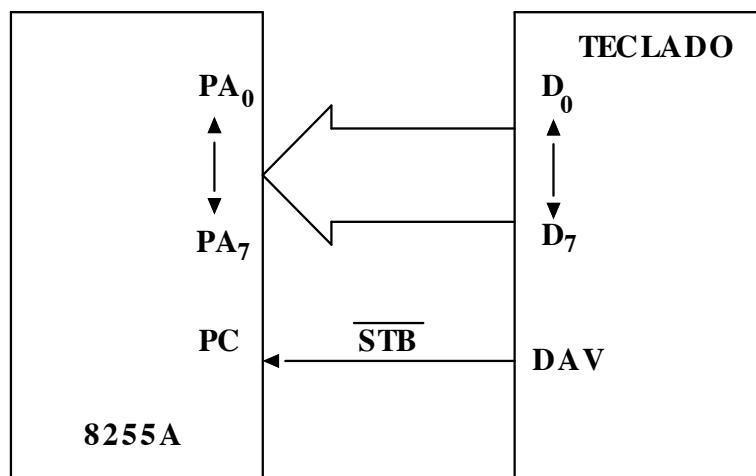


Figura 6 Utilización del 8255A para operación de habilitación periódica de entrada de un teclado.

Ejemplo 3

;procedimiento que lee el codificador del teclado y regresa con el carácter ASCII en AL

```
BIT5 EQU 20H

READ    PROC NEAR
        IN     AL,PORTC      ;leer puerto C
        TEST   AL,BIT5       ; verificar IBF
        JZ     READ          ;si IBF = 0
        IN     AL,PORTA      ;leer código ASCII
        RET
READ    ENDP
```

Salida de Habilitación Periódica Modo 1

La Figura 7 ilustra la configuración interna y los diagramas de tiempo del 8255A-5 cuando se opera como un dispositivo de salida de habilitación periódica bajo el modo 1. La operación de salida de habilitación periódica es similar a la salida de modo 0 excepto que señales de control se incluyen para proporcionar el reconocimiento.

Cuando un dato se escribe a un puerto programado como puerto de salida de habilitación periódica, la señal OBF (output buffer full) cambia a 0 lógico para indicar que un dato esta presente en el amarrador del puerto. Esta señal indica que los datos están disponibles para un dispositivo externo de E/S que removerá el dato por medio una señal de habilitación periódica en la terminal ACK (acknowledge) del puerto. La señal ACK regresa la señal OBF a un 1 lógico, indicando que el reforzador no esta lleno.

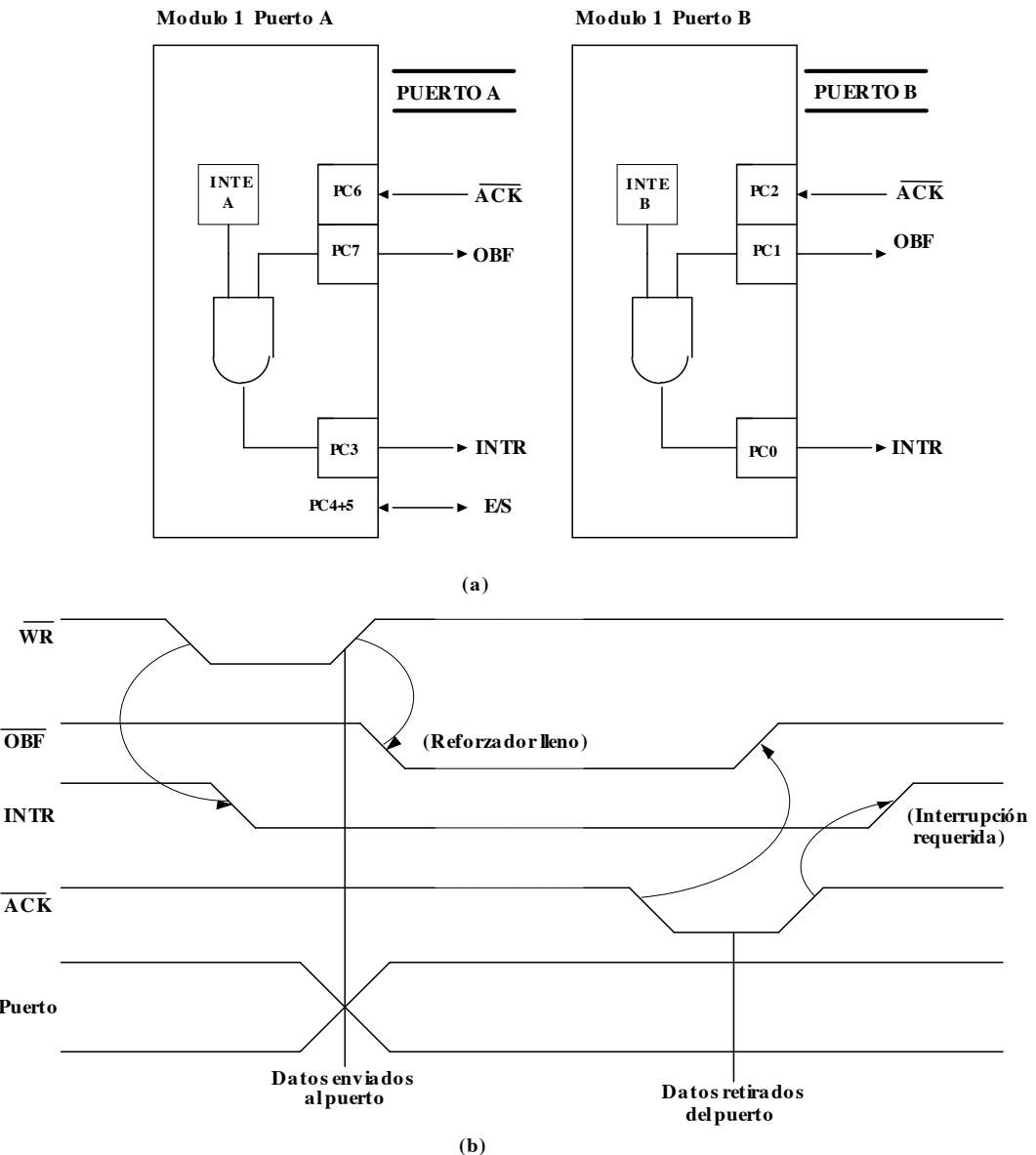


Figura 7 Operación de habilitación periódica de salida (modo 1) del 8266A. (a) Estructura interna. (b) Diagrama de tiempos.

Definiciones de Señales para el Modo 1 de Salida con habilitación periódica.

1. **OBF** - Reforzador de Salida lleno: Una salida que cambia a bajo cuando los datos son sacados (OUT) del amarrador del puerto A o del puerto B. Esta señal se pone a 1 lógico cuando el pulso ACK se regresa desde el dispositivo externo.
2. **ACK** - Reconocimiento: Una señal que ocasiona que la terminal **OBF** regrese al nivel 1 lógico. La **ACK** es una respuesta desde un dispositivo externo que indica que ha recibido el dato del puerto del 8255A-5.
3. **INTR** - Requerimiento de interrupción: Una señal que interrumpe al microprocesador cuando el dispositivo externo recibe el dato a través de la señal **ACK**. Esta terminal es calificada por el bit interno **INTE** (interrupt enable).

4. INTE - Habilitador de Interrupciones: No es una entrada ni una salida, es un bit interno programado para habilitar o deshabilitar la terminal INTR. El bit INTE A se programa como PC₆ y INTE B es PC₂.
5. PC₅, PC₄ - Bit 5 y 4 del puerto C que son terminales de E/S de propósito general: El bit de control poner o reinicializar puede ser usado para colocar o reiniciar estas dos terminales.

Ejemplo de Salida con habilitación periódica. La interfaz de la impresora discutida anteriormente se emplea aquí para demostrar como realizar la sincronización de salida de habilitación periódica entre la impresora y el 8255A-5.

La Figura 8 ilustra el puerto B conectado a una impresora en paralelo con ocho entradas de datos para recibir datos codificados en ASCII, una entrada \overline{DS} (data strobe) para enviar datos de habilitación periódica a la impresora, y una salida \overline{ACK} de reconocimiento de recepción del carácter ASCII.

En este circuito no hay señal para generar la señal \overline{DS} para la impresora, entonces PC₄ es usada con software para generar la señal \overline{DS} . La señal \overline{ACK} es regresada desde el reconocimiento de recepción de datos de la impresora, y se conecta a la entrada \overline{ACK} del 8255A-5.

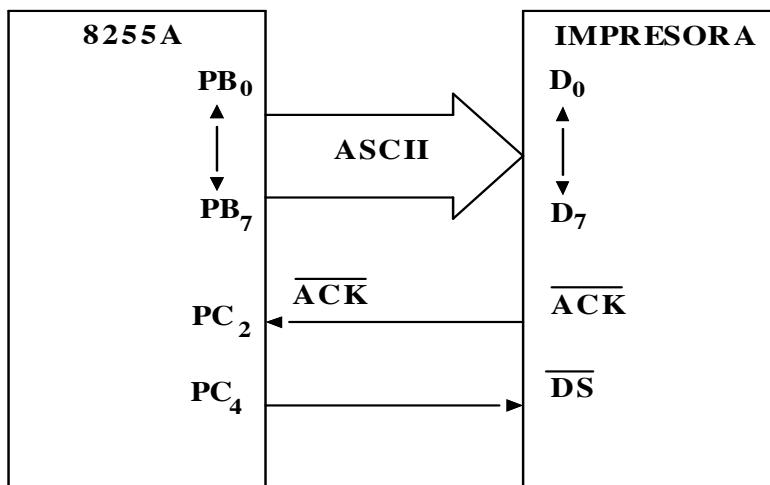


Figura 8 El 8255A conectado a una interfaz de impresora en paralelo que muestra el modo de habilitación periódica de salida .

El Ejemplo 4 lista el software que envía el carácter codificado en ASCII en AH hacia la impresora. El procedimiento primero prueba \overline{OBF} para decidir si la impresora ha retirado el dato del puerto B. Si no ($\overline{OBF}=0$), el procedimiento espera que la señal de \overline{ACK} regrese de la impresora. Si $\overline{OBF}=1$, entonces el procedimiento envía el contenido de AH a la impresora a través del puerto B y también manda la señal \overline{DS} .

Ejemplo 4

;procedimiento que envía el carácter en AH codificado
;en ASCII a la impresora a través del puerto B

```
BIT1 EQU 2H
```

```
PRINT PROC NEAR
```

;revisa que la impresora este lista

```
IN AL,PORTC ;obtener OBF
TEST AL,BIT1 ;probar OBF
JZ PRINT ; si OBF = 0
```

;manda el carácter a la impresora por el puerto B

```
MOV AL,AH
OUT PORTB,AL
```

;manda DS a la impresora

```
MOV AL,8 ;borrar DS
OUT COMMAND,AL
MOV AL,9 ;poner a 1 DS
OUT COMMAND,AL
RET
```

```
PRINT ENDP
```

Modo 2 Operación Bidireccional

En el modo 2, el cual se permite solamente para el grupo A, el puerto A se convierte en bidireccional, permitiendo que los datos sean transmitidos y recibidos a través de los mismos ocho cables. Los datos acarreados bidireccionalmente son útiles al interfazar dos computadoras.

Son también usados para la interfaz estándar paralela de alta velocidad IEEE-488 (canal de instrumentación de propósito general GPIB). La Figura 9 muestra la estructura interna y el diagrama de tiempo para el modo 2 de operación bidireccional.

Definiciones de Señales para el Modo 2 Bidireccional.

1. INTR - Requerimiento de interrupción: Una salida usada para interrumpir al microprocesador en condiciones de entrada o salida.
2. OBF - Reforzador de Salida Lleno: Una salida que indica que el reforzador de salida contiene datos para el canal bidireccional.

3. ACK - Reconocimiento: Una entrada que habilita los reforzadores de tres estados para que los datos puedan aparecer en el puerto A. Si ACK es un 1 lógico, los reforzadores de salida del puerto A están en su estado de alta impedancia.
4. STB - Señal de Habilitación Periódica: Una entrada usada para cargar el amarrador de entrada del puerto A con los datos externos provenientes del canal bidireccional del puerto A.
5. IBF - Reforzador de Entrada Lleno: Una salida usada para indicar que el reforzador de entrada contiene datos provenientes del reforzador bidireccional externo.
6. INTE - Habilitador de Interrupciones: Bit internos (INTE1 e INTE2) que habilitan la terminal INTR. El estado de la terminal INTR es controlado por los bit PC₆ (INTE1) y PC₄ (INTE2) del puerto C.

El Canal Bidireccional. El canal bidireccional se emplea haciendo referencia al puerto A con las instrucciones IN y OUT. Para transmitir datos a través del canal bidireccional, el programa primero prueba la señal OBF para determinar cuando el reforzador de salida esta vacío. Si es así, entonces se envían los datos al reforzador de salida a través de la instrucción OUT. La circuitería externa también monitorea la señal OBF para decidir si el microprocesador ha enviado datos al canal. Tan pronto como la circuitería recibe un 0 lógico en OBF, manda de regreso la señal ACK para retirarlo del reforzador de salida. La señal ACK coloca el bit OBF y también habilita el reforzador de salida de tres estados, entonces los datos pueden ser leídos. El ejemplo 5 lista el procedimiento que transmite el contenido del registro AH a través del puerto bidireccional A.

Ejemplo 5

;procedimiento para transmitir AH a través del
;canal bidireccional del puerto A

```
BIT7 EQU 80H
TRANS PROC NEAR

;revisa OBF
IN AL,PORTC      ;obtener OBF
TEST AL,BIT7      ;verificar OBF
JZ TRANS          ;si OBF = 0

;manda el dato
MOV AL,AH          ;obtener dato
OUT PORTB,AL
RET

TRANS ENDP
```

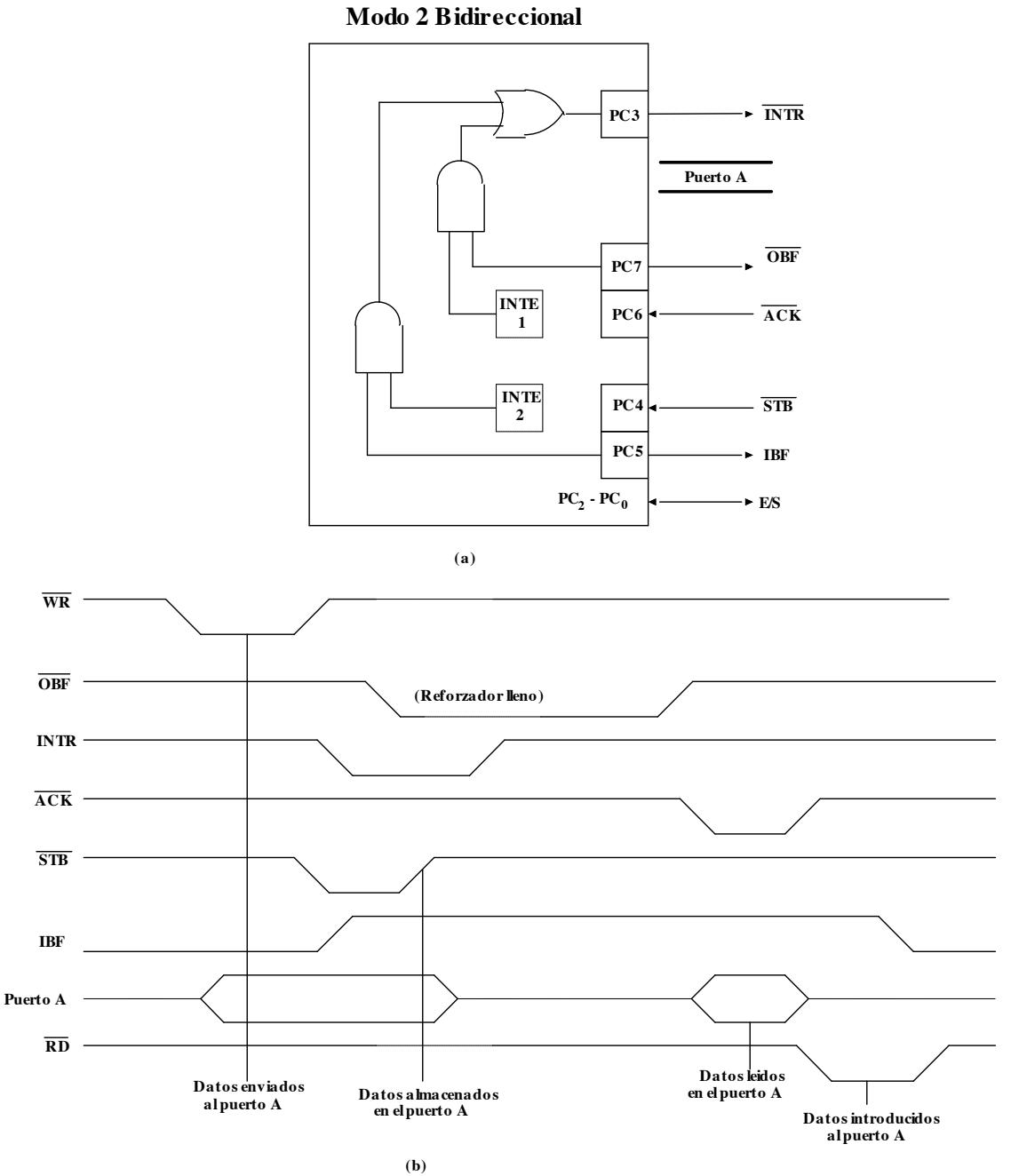


Figura 9 Modo 2 de operación del 8255A. (a) Estructura interna. (b) Diagrama de tiempos.

Para recibir datos a través del canal del puerto bidireccional A, se prueba el bit IBF con software para decidir si los datos han sido enviados al puerto. Si IBF=1, entonces se meten los datos utilizando una instrucción IN. La interfaz externa envía datos al puerto usando la señal STB. Cuando STB se activa, la señal STB cambia a 1 lógico, y el dato en el puerto a se mantiene dentro del puerto en un amarrador. Cuando la instrucción IN se ejecuta, el bit IBF se limpia y el dato en el puerto se traslada a AL. El Ejemplo 6 lista un procedimiento que lee datos del puerto.

Ejemplo 6

;procedimiento que introduce datos del canal
;bidireccional salvándolos en AL

```
BIT5      EQU 20H
READ      PROC NEAR
    ;verificar IBF
    IN  AL,PORTC      ;obtener IBF
    TEST AL,BIT5 ;verificar IBF
    JZ  READ          ;si IBF = 0
    ;leer IBF
    IN  AL,PORTA
    RET
READ      ENDP
```

La terminal INTR (requerimiento de interrupción) puede ser activada en ambas direcciones de datos que fluyen a través del canal. Si INTR se habilita por ambos bit INT, entonces los reforzadores de salida y entrada ocasionan un requerimiento de interrupción. Esto ocurre cuando los datos se envían al reforzador usando STB o cuando los datos se escriben usando OUT.

Sumario de Modos del 8255A-5.

La Figura 10 muestra un resumen gráfico de los tres modos de operación para el 8255A-5. El modo 0 proporciona E/S simple, el modo 1 proporciona E/S con señal de habilitación periódica, y el modo 2 proporciona E/S bidireccional. Como se mencionó anteriormente, estos modos se seleccionan a través del registro de control del 8255A-5.

	Modo 0		Modo 1		Modo 2	
Puerto A	IN	OUT	IN	OUT	I/O	
Puerto B	IN	OUT	IN	OUT	Not used	
0			INTR _B	INTR _B	I/O	
1			IBF _B	OBF _B	I/O	
2			STB _B	ACK _B	I/O	
3	IN	OUT	INTR _A	INTR _A	INTR	
4			STB _A	I/O	STB	
5			IBF _B	I/O	IBF	
6			I/O	ACK _A	ACK	
7			I/O	OBF _A	OBF	

Figura 10 Resumen de las conexiones de los puertos del 8255A.

CMOS Programmable Peripheral Interface

June 1998

Features

- Pin Compatible with NMOS 8255A
- 24 Programmable I/O Pins
- Fully TTL Compatible
- High Speed, No "Wait State" Operation with 5MHz and 8MHz 80C86 and 80C88
- Direct Bit Set/Reset Capability
- Enhanced Control Word Read Capability
- L7 Process
- 2.5mA Drive Capability on All I/O Ports
- Low Standby Power (ICCSB) 10µA

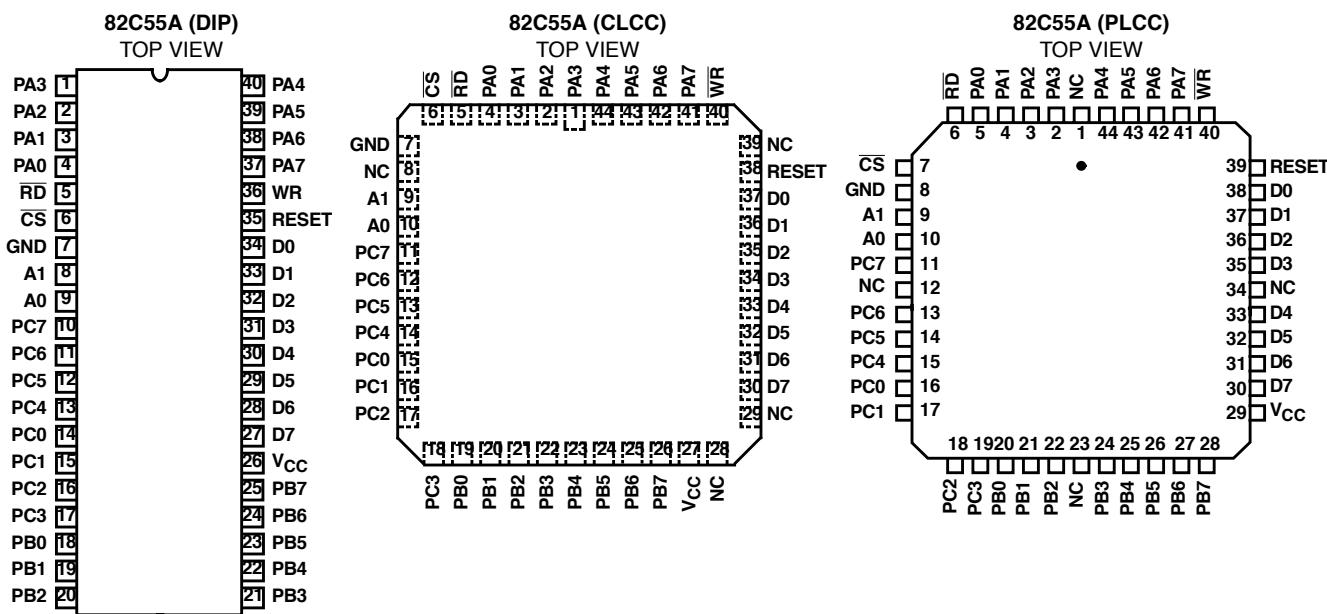
Ordering Information

PART NUMBERS		PACKAGE	TEMPERATURE RANGE	PKG. NO.
5MHz	8MHz			
CP82C55A-5	CP82C55A	40 Ld PDIP	0°C to 70°C	E40.6
IP82C55A-5	IP82C55A		-40°C to 85°C	E40.6
CS82C55A-5	CS82C55A	44 Ld PLCC	0°C to 70°C	N44.65
IS82C55A-5	IS82C55A		-40°C to 85°C	N44.65
CD82C55A-5	CD82C55A	40 Ld CERDIP	0°C to 70°C	F40.6
ID82C55A-5	ID82C55A		-40°C to 85°C	F40.6
MD82C55A-5/B	MD82C55A/B	40 Ld CERDIP	-55°C to 125°C	F40.6
8406601QA	8406602QA	SMD#		F40.6
MR82C55A-5/B	MR82C55A/B	44 Pad CLCC	-55°C to 125°C	J44.A
8406601XA	8406602XA	44 Pad CLCC		J44.A

Description

The Intersil 82C55A is a high performance CMOS version of the industry standard 8255A and is manufactured using a self-aligned silicon gate CMOS process (Scaled SAJ IV). It is a general purpose programmable I/O device which may be used with many different microprocessors. There are 24 I/O pins which may be individually programmed in 2 groups of 12 and used in 3 major modes of operation. The high performance and industry standard configuration of the 82C55A make it compatible with the 80C86, 80C88 and other microprocessors.

Static CMOS circuit design insures low operating power. TTL compatibility over the full military temperature range and bus hold circuitry eliminate the need for pull-up resistors. The Intersil advanced SAJ process results in performance equal to or greater than existing functionally equivalent products at a fraction of the power.

Pinouts

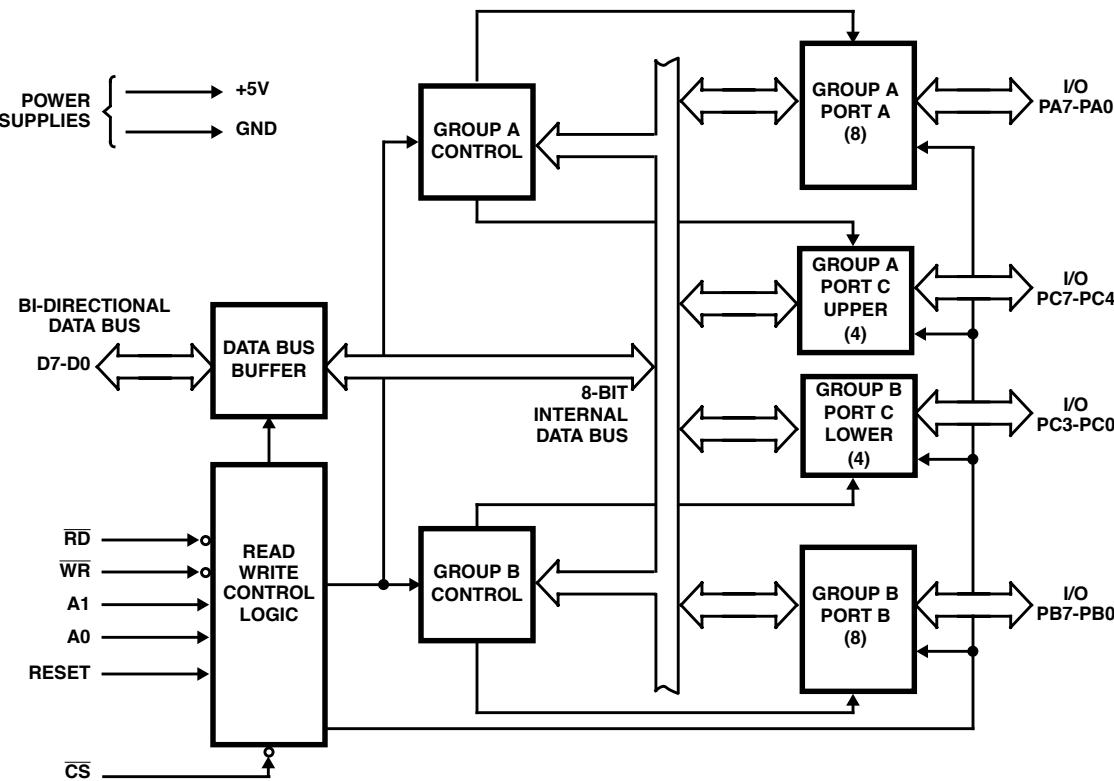
CAUTION: These devices are sensitive to electrostatic discharge; follow proper IC Handling Procedures.

http://www.intersil.com or 407-727-9207 | Copyright © Intersil Corporation 1999

File Number 2969.2

Pin Description

SYMBOL	PIN NUMBER	TYPE	DESCRIPTION
V _{CC}	26		V _{CC} : The +5V power supply pin. A 0.1 μ F capacitor between pins 26 and 7 is recommended for decoupling.
GND	7		GROUND
D0-D7	27-34	I/O	DATA BUS: The Data Bus lines are bidirectional three-state pins connected to the system data bus.
RESET	35	I	RESET: A high on this input clears the control register and all ports (A, B, C) are set to the input mode with the "Bus Hold" circuitry turned on.
CS	6	I	CHIP SELECT: Chip select is an active low input used to enable the 82C55A onto the Data Bus for CPU communications.
RD	5	I	READ: Read is an active low input control signal used by the CPU to read status information or data via the data bus.
WR	36	I	WRITE: Write is an active low input control signal used by the CPU to load control words and data into the 82C55A.
A0-A1	8, 9	I	ADDRESS: These input signals, in conjunction with the RD and WR inputs, control the selection of one of the three ports or the control word register. A0 and A1 are normally connected to the least significant bits of the Address Bus A0, A1.
PA0-PA7	1-4, 37-40	I/O	PORT A: 8-bit input and output port. Both bus hold high and bus hold low circuitry are present on this port.
PB0-PB7	18-25	I/O	PORT B: 8-bit input and output port. Bus hold high circuitry is present on this port.
PC0-PC7	10-17	I/O	PORT C: 8-bit input and output port. Bus hold circuitry is present on this port.

Functional Diagram

Functional Description

Data Bus Buffer

This three-state bi-directional 8-bit buffer is used to interface the 82C55A to the system data bus. Data is transmitted or received by the buffer upon execution of input or output instructions by the CPU. Control words and status information are also transferred through the data bus buffer.

Read/Write and Control Logic

The function of this block is to manage all of the internal and external transfers of both Data and Control or Status words. It accepts inputs from the CPU Address and Control busses and in turn, issues commands to both of the Control Groups.

(CS) Chip Select. A “low” on this input pin enables the communication between the 82C55A and the CPU.

(RD) Read. A “low” on this input pin enables 82C55A to send the data or status information to the CPU on the data bus. In essence, it allows the CPU to “read from” the 82C55A.

(WR) Write. A “low” on this input pin enables the CPU to write data or control words into the 82C55A.

(A0 and A1) Port Select 0 and Port Select 1. These input signals, in conjunction with the RD and WR inputs, control the selection of one of the three ports or the control word register. They are normally connected to the least significant bits of the address bus (A0 and A1).

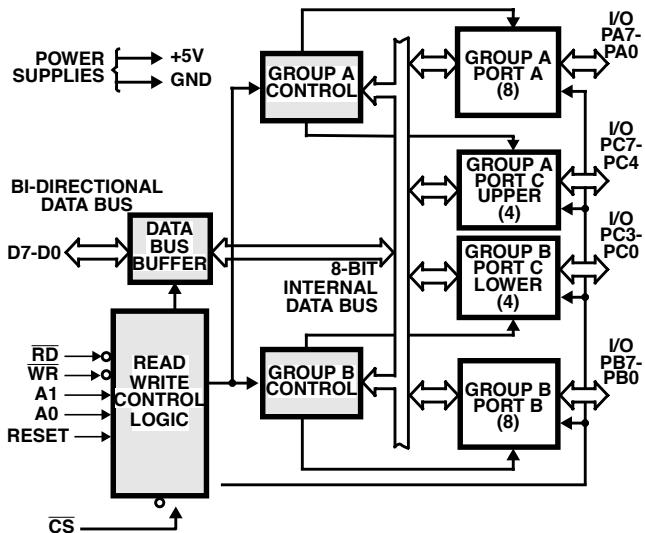


FIGURE 1. 82C55A BLOCK DIAGRAM. DATA BUS BUFFER, READ/WRITE, GROUP A & B CONTROL LOGIC FUNCTIONS

(RESET) Reset. A “high” on this input initializes the control register to 9Bh and all ports (A, B, C) are set to the input mode. “Bus hold” devices internal to the 82C55A will hold the I/O port inputs to a logic “1” state with a maximum hold current of 400 μ A.

Group A and Group B Controls

The functional configuration of each port is programmed by the systems software. In essence, the CPU “outputs” a control word to the 82C55A. The control word contains information such as “mode”, “bit set”, “bit reset”, etc., that initializes the functional configuration of the 82C55A.

Each of the Control blocks (Group A and Group B) accepts “commands” from the Read/Write Control logic, receives “control words” from the internal data bus and issues the proper commands to its associated ports.

Control Group A - Port A and Port C upper (C7 - C4)

Control Group B - Port B and Port C lower (C3 - C0)

The control word register can be both written and read as shown in the “Basic Operation” table. Figure 4 shows the control word format for both Read and Write operations. When the control word is read, bit D7 will always be a logic “1”, as this implies control word mode information.

82C55A BASIC OPERATION

INPUT OPERATION (READ)					
A1	A0	RD	WR	CS	
0	0	0	1	0	Port A → Data Bus
0	1	0	1	0	Port B → Data Bus
1	0	0	1	0	Port C → Data Bus
1	1	0	1	0	Control Word → Data Bus
OUTPUT OPERATION (WRITE)					
0	0	1	0	0	Data Bus → Port A
0	1	1	0	0	Data Bus → Port B
1	0	1	0	0	Data Bus → Port C
1	1	1	0	0	Data Bus → Control
DISABLE FUNCTION					
X	X	X	X	1	Data Bus → Three-State
X	X	1	1	0	Data Bus → Three-State

Ports A, B, and C

The 82C55A contains three 8-bit ports (A, B, and C). All can be configured to a wide variety of functional characteristics by the system software but each has its own special features or "personality" to further enhance the power and flexibility of the 82C55A.

Port A One 8-bit data output latch/buffer and one 8-bit data input latch. Both "pull-up" and "pull-down" bus-hold devices are present on Port A. See Figure 2A.

Port B One 8-bit data input/output latch/buffer and one 8-bit data input buffer. See Figure 2B.

Port C One 8-bit data output latch/buffer and one 8-bit data input buffer (no latch for input). This port can be divided into two 4-bit ports under the mode control. Each 4-bit port contains a 4-bit latch and it can be used for the control signal output and status signal inputs in conjunction with ports A and B. See Figure 2B.

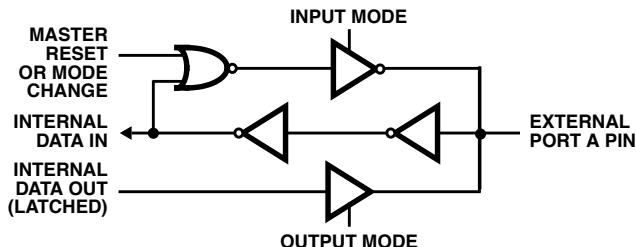


FIGURE 2A. PORT A BUS-HOLD CONFIGURATION

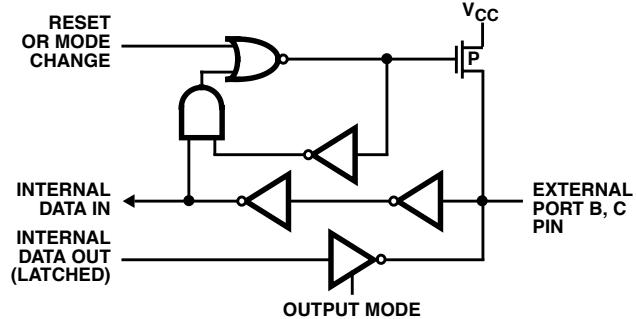


FIGURE 2B. PORT B AND C BUS-HOLD CONFIGURATION

FIGURE 2. BUS-HOLD CONFIGURATION

Operational Description

Mode Selection

There are three basic modes of operation than can be selected by the system software:

- Mode 0 - Basic Input/Output
- Mode 1 - Strobed Input/Output
- Mode 2 - Bi-directional Bus

When the reset input goes "high", all ports will be set to the input mode with all 24 port lines held at a logic "one" level by internal bus hold devices. After the reset is removed, the 82C55A can remain in the input mode with no additional initialization required. This eliminates the need to pullup or pulldown resistors in all-CMOS designs. The control word

register will contain 9Bh. During the execution of the system program, any of the other modes may be selected using a single output instruction. This allows a single 82C55A to service a variety of peripheral devices with a simple software maintenance routine. Any port programmed as an output port is initialized to all zeros when the control word is written.

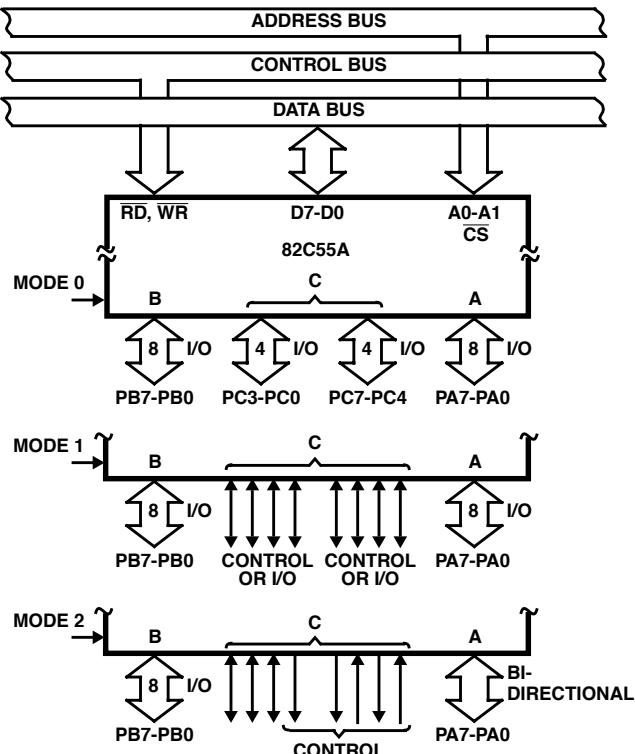


FIGURE 3. BASIC MODE DEFINITIONS AND BUS INTERFACE

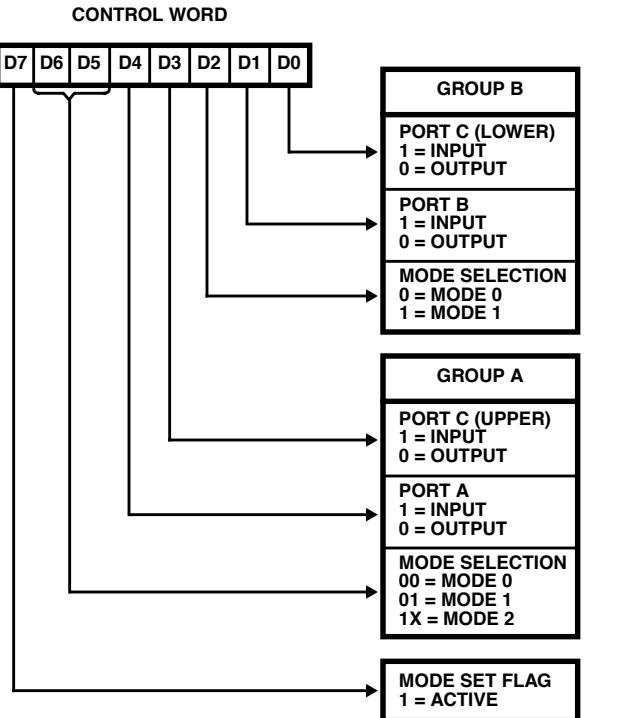


FIGURE 4. MODE DEFINITION FORMAT

The modes for Port A and Port B can be separately defined, while Port C is divided into two portions as required by the Port A and Port B definitions. All of the output registers, including the status flip-flops, will be reset whenever the mode is changed. Modes may be combined so that their functional definition can be "tailored" to almost any I/O structure. For instance: Group B can be programmed in Mode 0 to monitor simple switch closings or display computational results, Group A could be programmed in Mode 1 to monitor a keyboard or tape reader on an interrupt-driven basis.

The mode definitions and possible mode combinations may seem confusing at first, but after a cursory review of the complete device operation a simple, logical I/O approach will surface. The design of the 82C55A has taken into account things such as efficient PC board layout, control signal definition vs. PC layout and complete functional flexibility to support almost any peripheral device with no external logic. Such design represents the maximum use of the available pins.

Single Bit Set/Reset Feature (Figure 5)

Any of the eight bits of Port C can be Set or Reset using a single Output instruction. This feature reduces software requirements in control-based applications.

When Port C is being used as status/control for Port A or B, these bits can be set or reset by using the Bit Set/Reset operation just as if they were output ports.

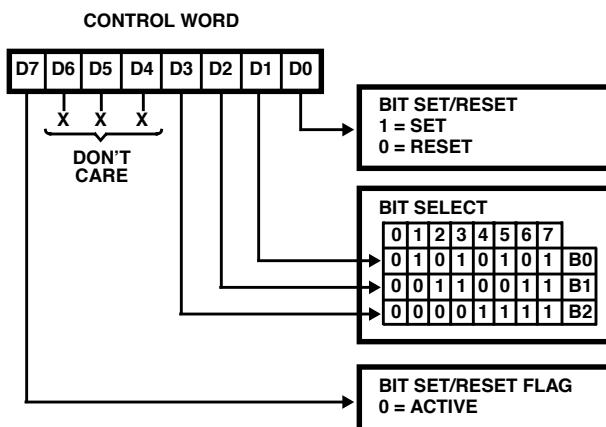


FIGURE 5. BIT SET/RESET FORMAT

Interrupt Control Functions

When the 82C55A is programmed to operate in mode 1 or mode 2, control signals are provided that can be used as interrupt request inputs to the CPU. The interrupt request signals, generated from port C, can be inhibited or enabled by setting or resetting the associated INTE flip-flop, using the bit set/reset function of port C.

This function allows the programmer to enable or disable a CPU interrupt by a specific I/O device without affecting any other device in the interrupt structure.

INTE Flip-Flop Definition

(BIT-SET)-INTE is SET - Interrupt Enable

(BIT-RESET)-INTE is Reset - Interrupt Disable

NOTE: All Mask flip-flops are automatically reset during mode selection and device Reset.

Operating Modes

Mode 0 (Basic Input/Output). This functional configuration provides simple input and output operations for each of the three ports. No handshaking is required, data is simply written to or read from a specific port.

Mode 0 Basic Functional Definitions:

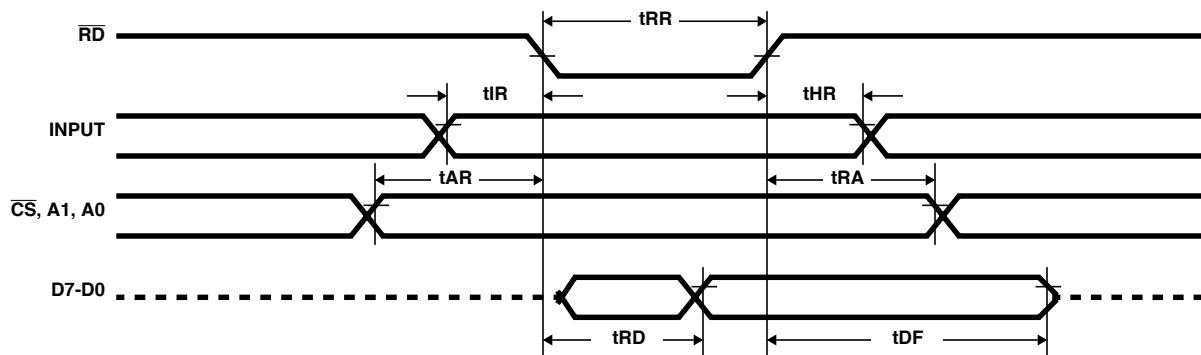
- Two 8-bit ports and two 4-bit ports
- Any Port can be input or output
- Outputs are latched
- Inputs are not latched
- 16 different Input/Output configurations possible

MODE 0 PORT DEFINITION

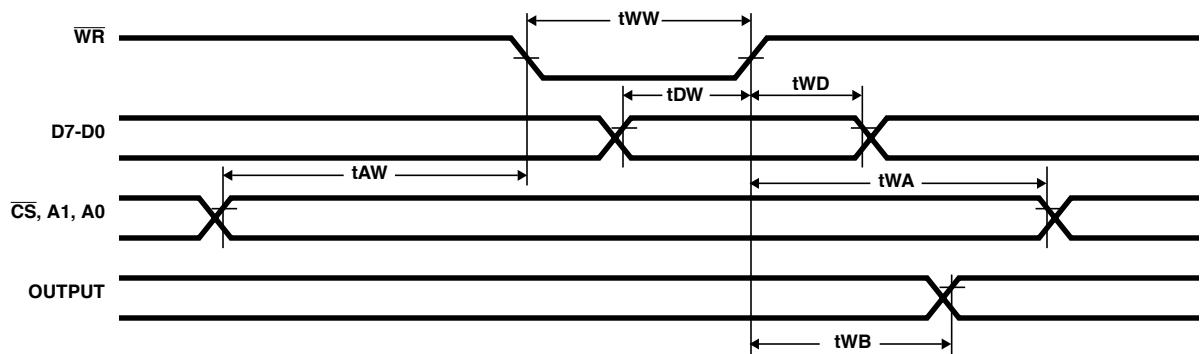
A			B		GROUP A		#	GROUP B	
D4	D3	D1	D0	PORTC (Upper)	PORTC (Lower)	PORT B	PORT C (Lower)		
0	0	0	0	Output	Output	0	Output	Output	
0	0	0	1	Output	Output	1	Output	Input	
0	0	1	0	Output	Output	2	Input	Output	
0	0	1	1	Output	Output	3	Input	Input	
0	1	0	0	Output	Input	4	Output	Output	
0	1	0	1	Output	Input	5	Output	Input	
0	1	1	0	Output	Input	6	Input	Output	
0	1	1	1	Output	Input	7	Input	Input	
1	0	0	0	Input	Output	8	Output	Output	
1	0	0	1	Input	Output	9	Output	Input	
1	0	1	0	Input	Output	10	Input	Output	
1	0	1	1	Input	Output	11	Input	Input	
1	1	0	0	Input	Input	12	Output	Output	
1	1	0	1	Input	Input	13	Output	Input	
1	1	1	0	Input	Input	14	Input	Output	
1	1	1	1	Input	Input	15	Input	Input	

82C55A

Mode 0 (Basic Input)

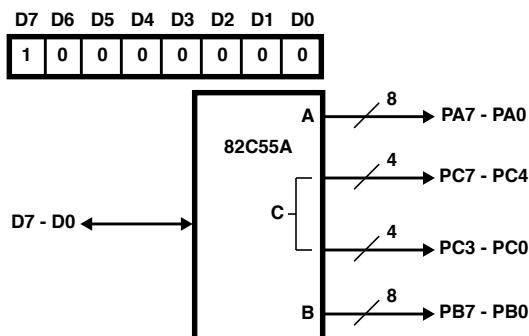


Mode 0 (Basic Output)

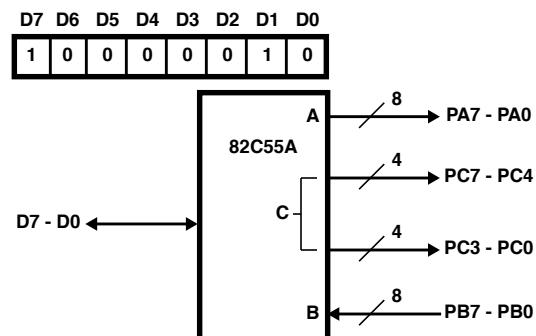


Mode 0 Configurations

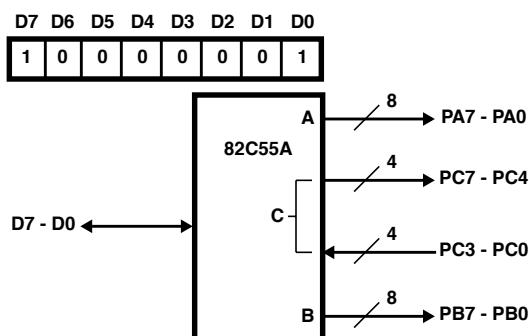
CONTROL WORD #0



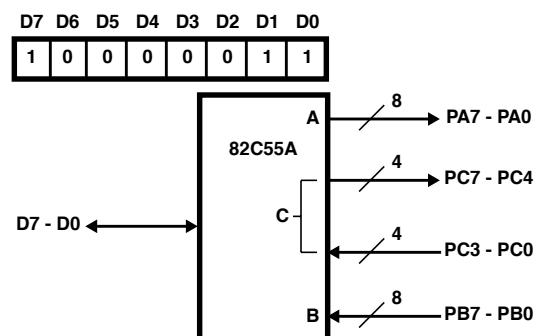
CONTROL WORD #2



CONTROL WORD #1



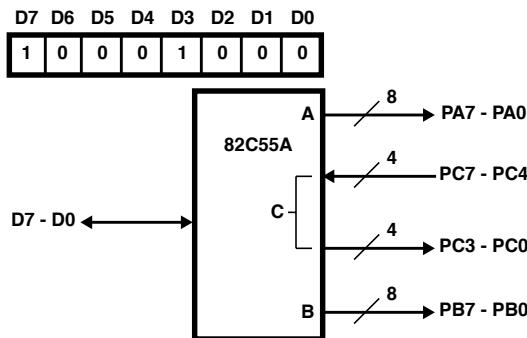
CONTROL WORD #3



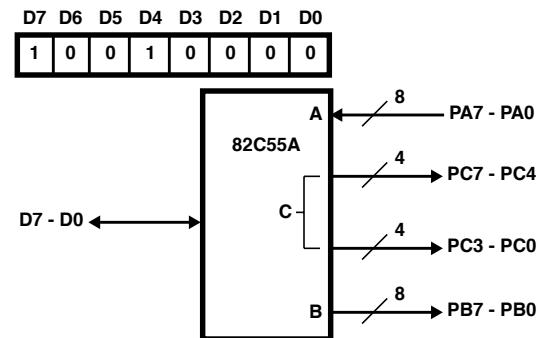
82C55A

Mode 0 Configurations (Continued)

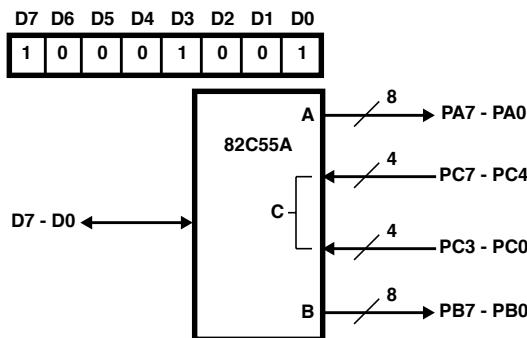
CONTROL WORD #4



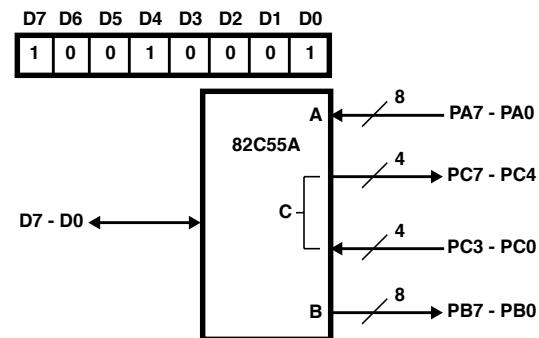
CONTROL WORD #8



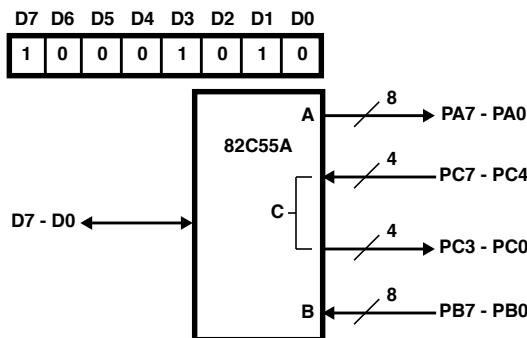
CONTROL WORD #5



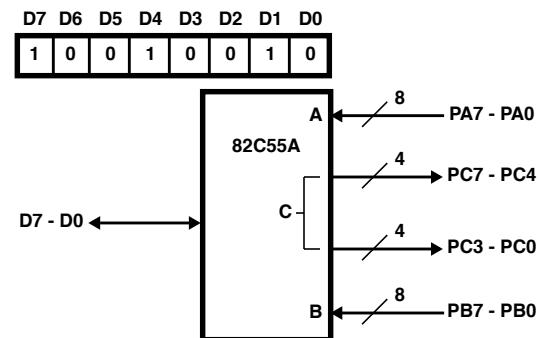
CONTROL WORD #9



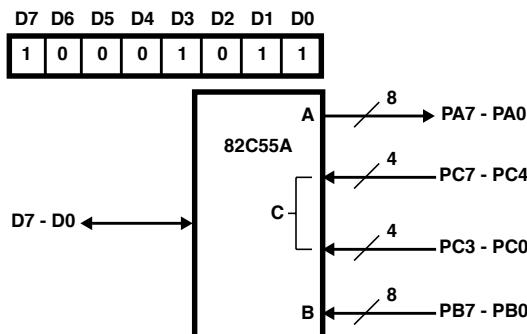
CONTROL WORD #6



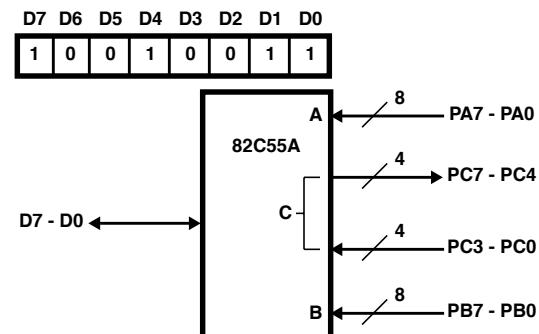
CONTROL WORD #10

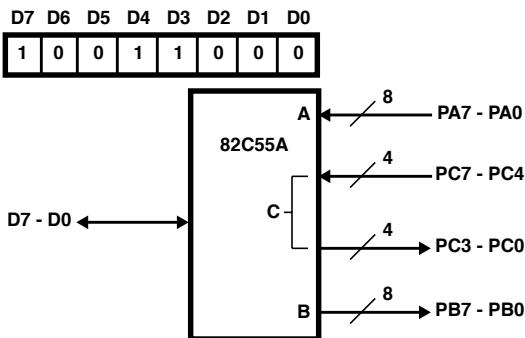
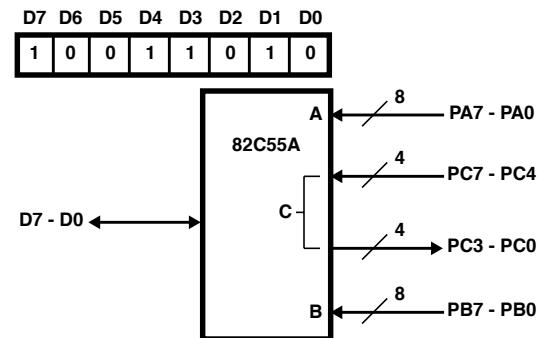
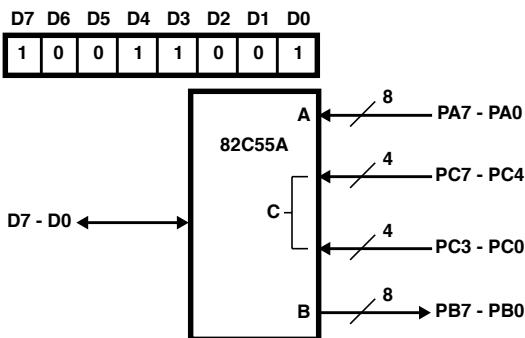
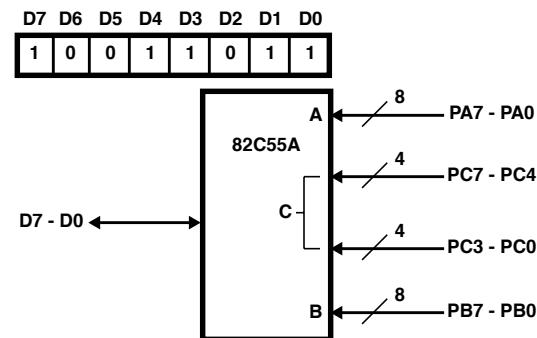


CONTROL WORD #7



CONTROL WORD #11



Mode 0 Configurations (Continued)**CONTROL WORD #12****CONTROL WORD #14****CONTROL WORD #13****CONTROL WORD #15****Operating Modes**

Mode 1 - (Strobed Input/Output). This functional configuration provides a means for transferring I/O data to or from a specified port in conjunction with strobes or "hand shaking" signals. In mode 1, port A and port B use the lines on port C to generate or accept these "hand shaking" signals.

Mode 1 Basic Function Definitions:

- Two Groups (Group A and Group B)
- Each group contains one 8-bit port and one 4-bit control/data port
- The 8-bit data port can be either input or output. Both inputs and outputs are latched.
- The 4-bit port is used for control and status of the 8-bit port.

Input Control Signal Definition

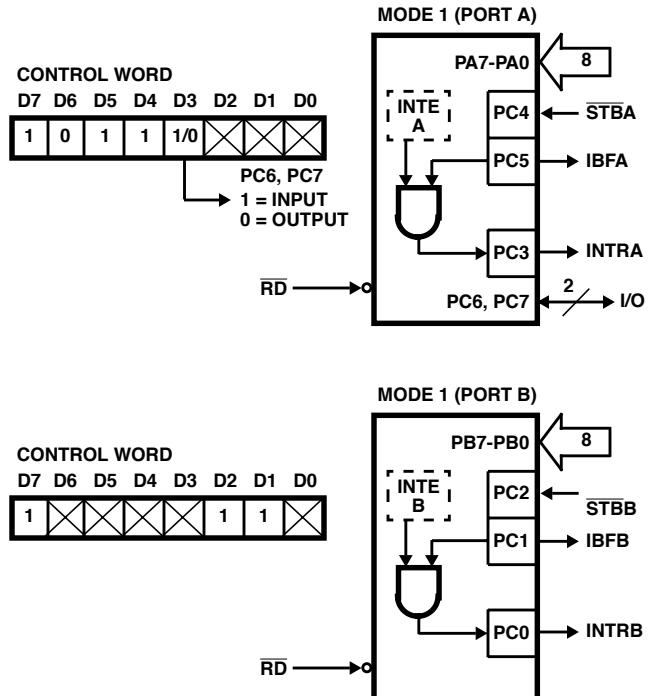
(Figures 6 and 7)

STB (Strobe Input)

A "low" on this input loads data into the input latch.

IBF (Input Buffer Full F/F)

A "high" on this output indicates that the data has been loaded into the input latch: in essence, an acknowledgement. IBF is set by STB input being low and is reset by the rising edge of the RD input.

**FIGURE 6. MODE 1 INPUT**

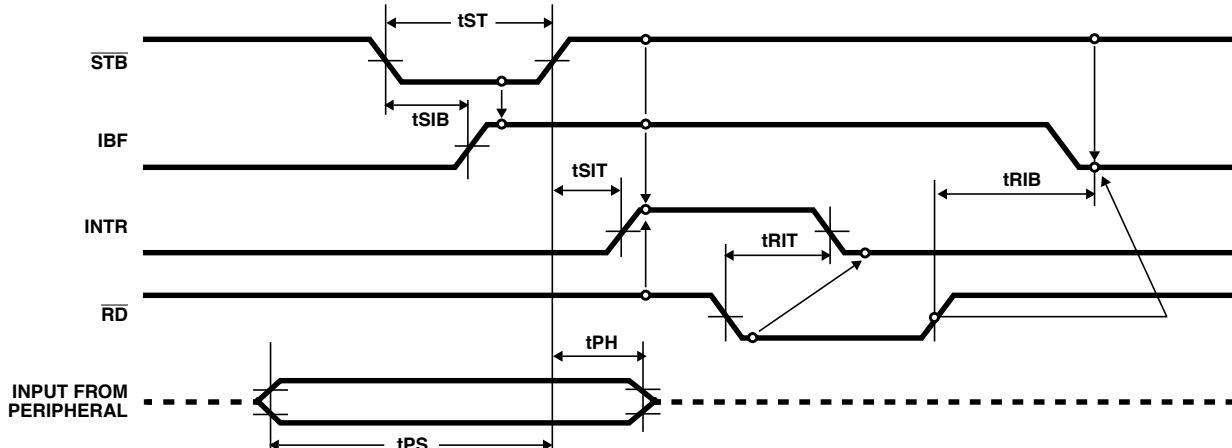


FIGURE 7. MODE 1 (STROBED INPUT)

INTR (Interrupt Request)

A “high” on this output can be used to interrupt the CPU when an input device is requesting service. INTR is set by the condition: STB is a “one”, IBF is a “one” and INTE is a “one”. It is reset by the falling edge of RD. This procedure allows an input device to request service from the CPU by simply strobing its data into the port.

INTE A

Controlled by bit set/reset of PC4.

INTE B

Controlled by bit set/reset of PC2.

Output Control Signal Definition

(Figure 8 and 9)

OBF - Output Buffer Full F/F). The \overline{OBF} output will go “low” to indicate that the CPU has written data out to be specified port. This does not mean valid data is sent out of the part at this time since \overline{OBF} can go true before data is available. Data is guaranteed valid at the rising edge of \overline{OBF} , (See Note 1). The \overline{OBF} F/F will be set by the rising edge of the WR input and reset by ACK input being low.

ACK - Acknowledge Input). A “low” on this input informs the 82C55A that the data from Port A or Port B is ready to be accepted. In essence, a response from the peripheral device indicating that it is ready to accept data, (See Note 1).

INTR - (Interrupt Request). A “high” on this output can be used to interrupt the CPU when an output device has accepted data transmitted by the CPU. INTR is set when ACK is a “one”, OBF is a “one” and INTE is a “one”. It is reset by the falling edge of WR.

INTE A

Controlled by Bit Set/Reset of PC6.

INTE B

Controlled by Bit Set/Reset of PC2.

NOTE:

1. To strobe data into the peripheral device, the user must operate the strobe line in a hand shaking mode. The user needs to send \overline{OBF} to the peripheral device, generates an ACK from the peripheral device and then latch data into the peripheral device on the rising edge of \overline{OBF} .

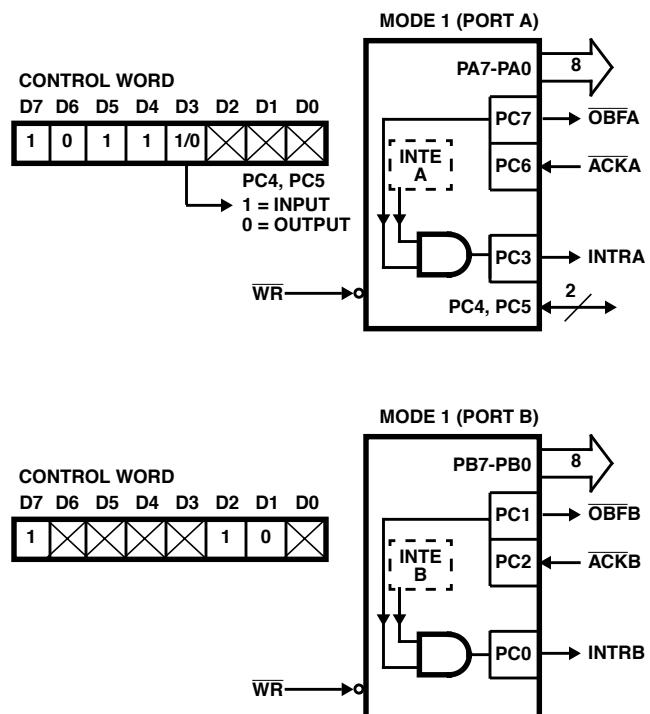


FIGURE 8. MODE 1 OUTPUT

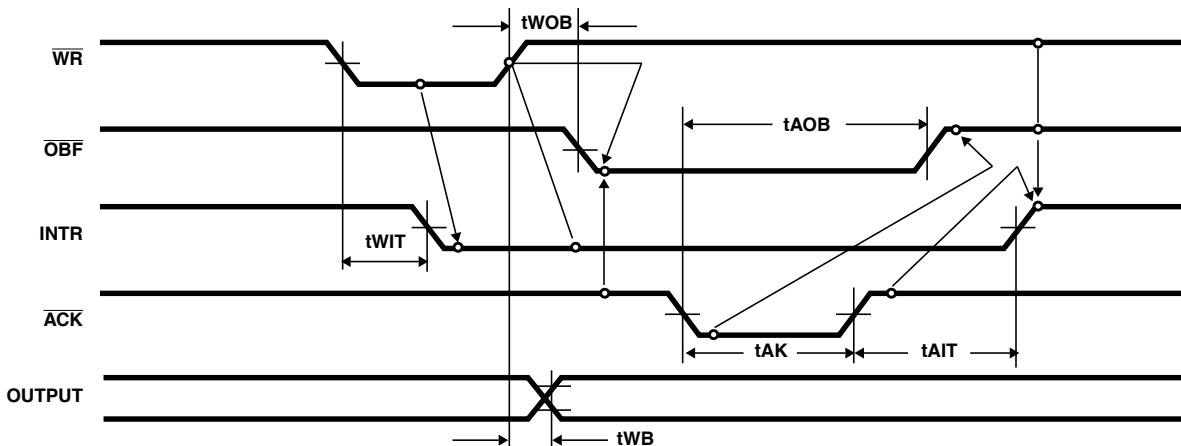
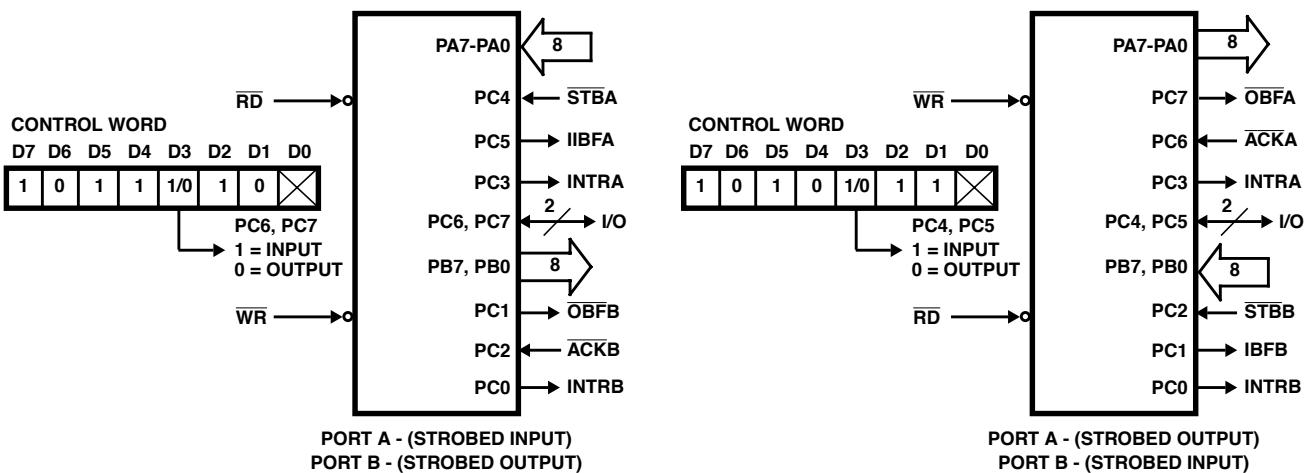


FIGURE 9. MODE 1 (STROBED OUTPUT)



Combinations of Mode 1: Port A and Port B can be individually defined as input or output in Mode 1 to support a wide variety of strobed I/O applications.

FIGURE 10. COMBINATIONS OF MODE 1

Operating Modes

Mode 2 (Strobed Bi-Directional Bus I/O)

The functional configuration provides a means for communicating with a peripheral device or structure on a single 8-bit bus for both transmitting and receiving data (bi-directional bus I/O). "Hand shaking" signals are provided to maintain proper bus flow discipline similar to Mode 1. Interrupt generation and enable/disable functions are also available.

Mode 2 Basic Functional Definitions:

- Used in Group A only
- One 8-bit, bi-directional bus Port (Port A) and a 5-bit control Port (Port C)
- Both inputs and outputs are latched
- The 5-bit control port (Port C) is used for control and status for the 8-bit, bi-directional bus port (Port A)

Bi-Directional Bus I/O Control Signal Definition

(Figures 11, 12, 13, 14)

INTR - (Interrupt Request). A high on this output can be used to interrupt the CPU for both input or output operations.

Output Operations

OBF - (Output Buffer Full). The OBF output will go "low" to indicate that the CPU has written data out to port A.

ACK - (Acknowledge). A "low" on this input enables the three-state output buffer of port A to send out the data. Otherwise, the output buffer will be in the high impedance state.

INTE 1 - (The INTE flip-flop associated with OBF). Controlled by bit set/reset of PC4.

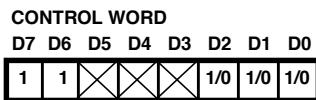
Input Operations

STB - (Strobe Input). A "low" on this input loads data into the input latch.

IBF - (Input Buffer Full F/F). A "high" on this output indicates that data has been loaded into the input latch.

INTE 2 - (The INTE flip-flop associated with IBF). Controlled by bit set/reset of PC4.

82C55A



PC2-PC0
 1 = INPUT
 0 = OUTPUT

PORT B
 1 = INPUT
 0 = OUTPUT

GROUP B MODE
 0 = MODE 0
 1 = MODE 1

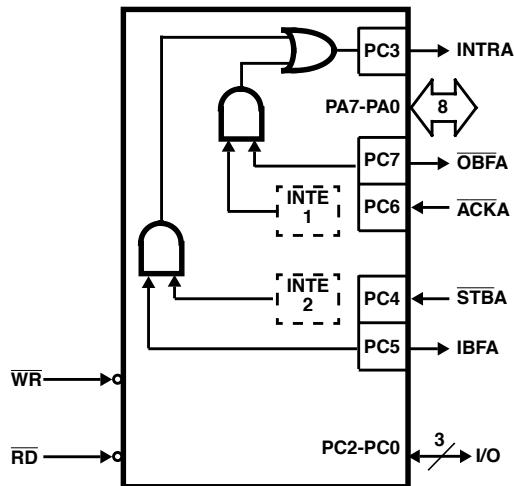
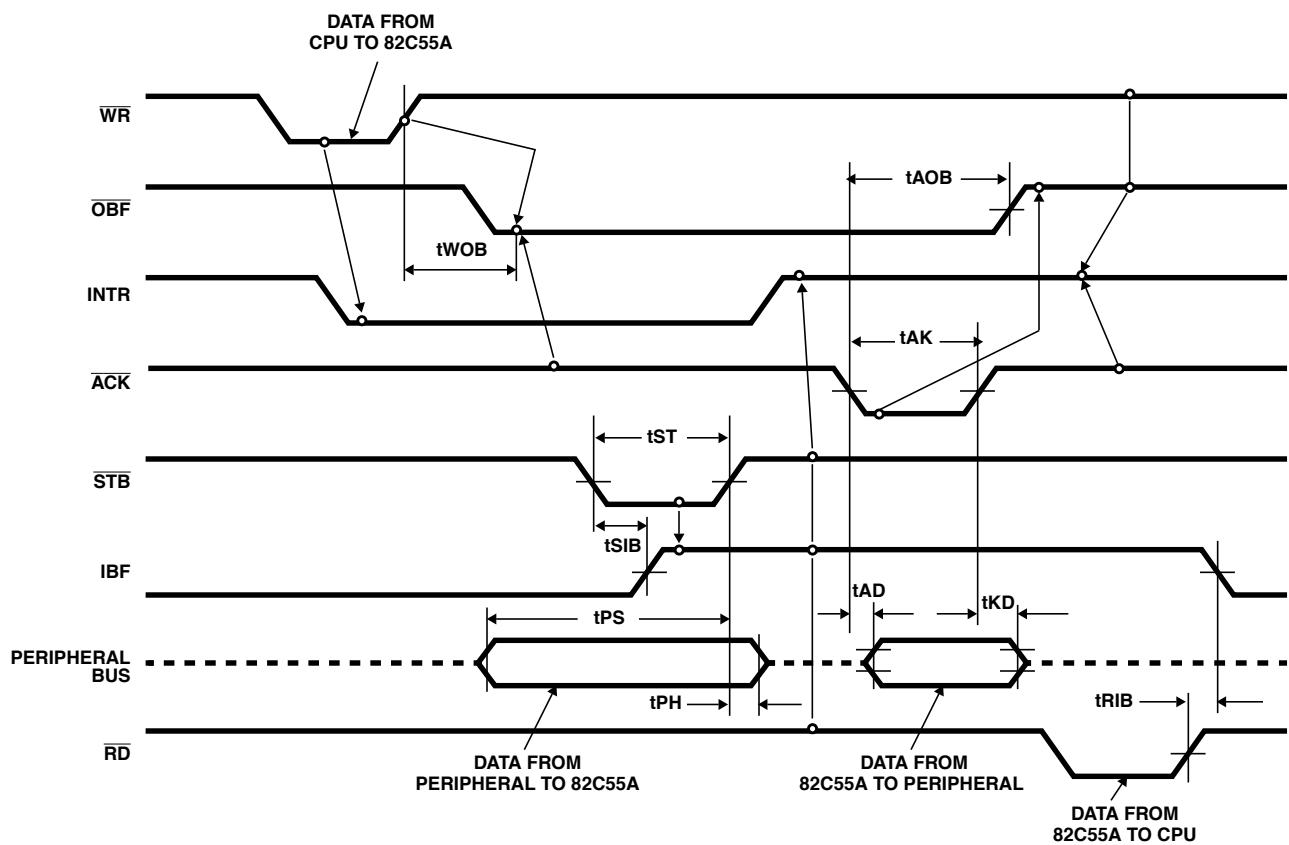


FIGURE 11. MODE CONTROL WORD

FIGURE 12. MODE 2



NOTE: Any sequence where \overline{WR} occurs before \overline{ACK} and \overline{STB} occurs before RD is permissible. ($INTR = IBF \cdot MASK \cdot \overline{STB} \cdot RD \div \overline{OBF} \cdot MASK \cdot \overline{ACK} \cdot \overline{WR}$)

FIGURE 13. MODE 2 (BI-DIRECTIONAL)

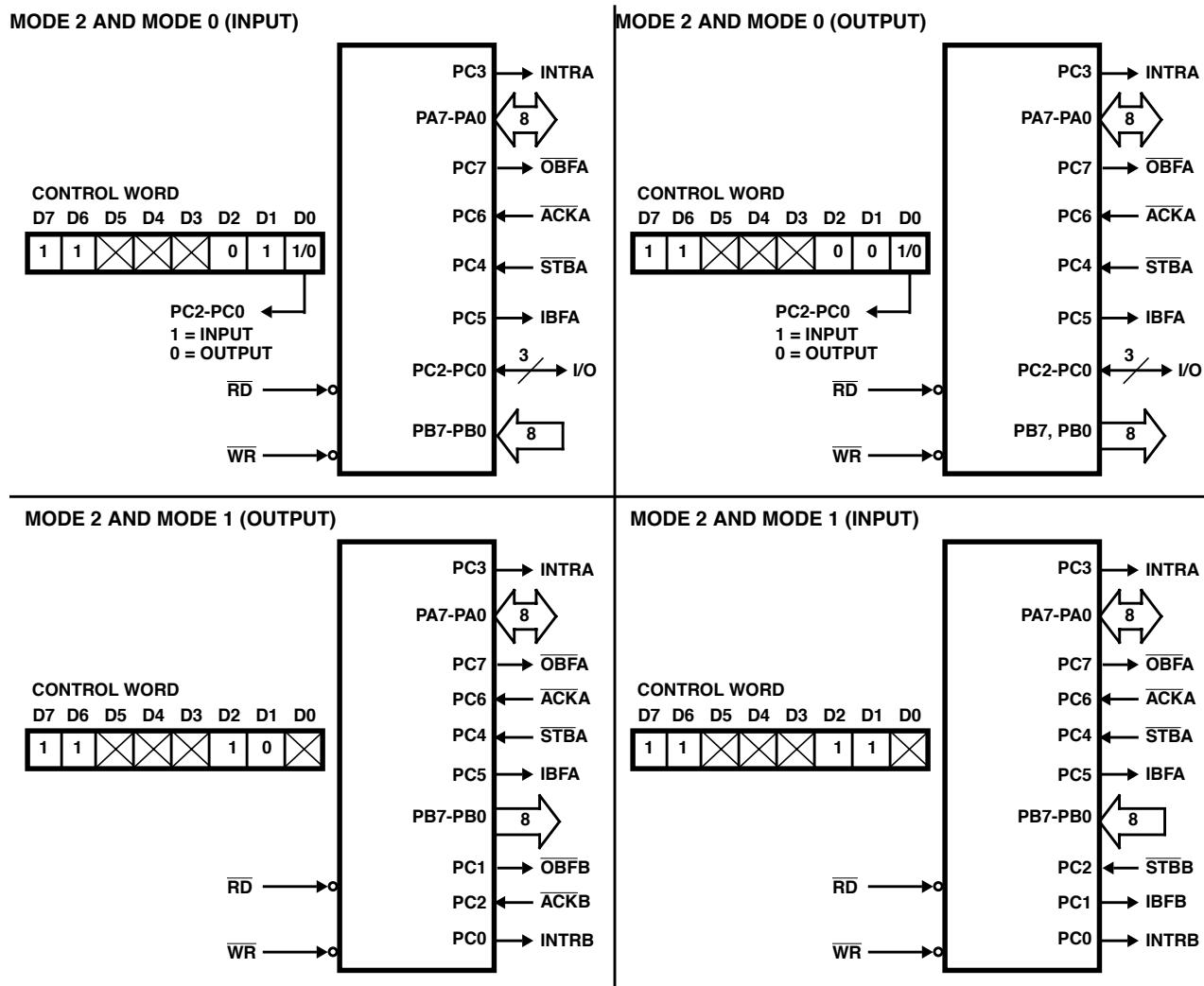


FIGURE 14. MODE 2 COMBINATIONS

	Mode Definition Summary				
	Mode 0		Mode 1		Mode 2
	In	Out	In	Out	Group A Only
PA0	In	Out	In	Out	
PA1	In	Out	In	Out	
PA2	In	Out	In	Out	
PA3	In	Out	In	Out	
PA4	In	Out	In	Out	
PA5	In	Out	In	Out	
PA6	In	Out	In	Out	
PA7	In	Out	In	Out	
PB0	In	Out	In	Out	
PB1	In	Out	In	Out	
PB2	In	Out	In	Out	
PB3	In	Out	In	Out	
PB4	In	Out	In	Out	
PB5	In	Out	In	Out	
PB6	In	Out	In	Out	
PB7	In	Out	In	Out	
PC0	In	Out	INTRB	INTRB	I/O
PC1	In	Out	IBFB	OBFB	I/O
PC2	In	Out	STBB	ACKB	I/O
PC3	In	Out	INTRA	INTRA	INTRA
PC4	In	Out	STBA	I/O	STBA
PC5	In	Out	IBFA	I/O	IBFA
PC6	In	Out	I/O	ACKA	ACKA
PC7	In	Out	I/O	OBFA	OBFA

Special Mode Combination Considerations

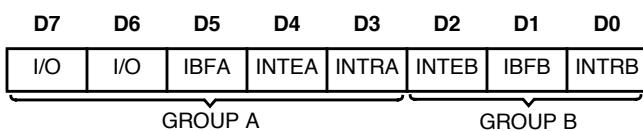
There are several combinations of modes possible. For any combination, some or all of Port C lines are used for control or status. The remaining bits are either inputs or outputs as defined by a "Set Mode" command.

During a read of Port C, the state of all the Port C lines, except the $\overline{\text{ACK}}$ and $\overline{\text{STB}}$ lines, will be placed on the data bus. In place of the $\overline{\text{ACK}}$ and $\overline{\text{STB}}$ line states, flag status will appear on the data bus in the PC2, PC4, and PC6 bit positions as illustrated by Figure 17.

Through a “Write Port C” command, only the Port C pins programmed as outputs in a Mode 0 group can be written. No other pins can be affected by a “Write Port C” command, nor can the interrupt enable flags be accessed. To write to any Port C output programmed as an output in Mode 1 group or to change an interrupt enable flag, the “Set/Reset Port C Bit” command must be used.

With a “Set/Reset Port Cea Bit” command, any Port C line programmed as an output (including IBF and OBF) can be written, or an interrupt enable flag can be either set or reset. Port C lines programmed as inputs, including ACK and STB lines, associated with Port C fare not affected by a “Set/Reset Port C Bit” command. Writing to the corresponding Port C bit positions of the ACK and STB lines with the “Set Reset Port C Bit” command will affect the Group A and Group B interrupt enable flags, as illustrated in Figure 17.

INPUT CONFIGURATION



OUTPUT CONFIGURATION

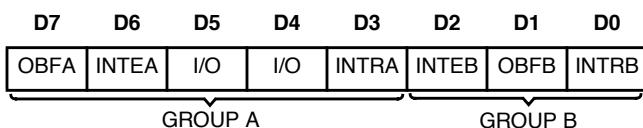


FIGURE 15. MODE 1 STATUS WORD FORMAT

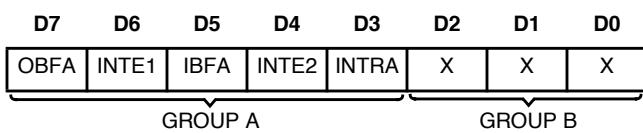


FIGURE 16. MODE 6 STATUS WORD FORMAT

Current Drive Capability

Any output on Port A, B or C can sink or source 2.5mA. This feature allows the 82C55A to directly drive Darlington type drivers and high-voltage displays that require such sink or source current.

Reading Port C Status (Figures 15 and 16)

In Mode 0, Port C transfers data to or from the peripheral device. When the 82C55A is programmed to function in Modes 1 or 2, Port C generates or accepts "hand shaking" signals with the peripheral device. Reading the contents of Port C allows the programmer to test or verify the "status" of each peripheral device and change the program flow accordingly.

There is not special instruction to read the status information from Port C. A normal read operation of Port C is executed to perform this function.

INTERRUPT ENABLE FLAG	POSITION	ALTERNATE PORT C PIN SIGNAL (MODE)
INTE B	PC2	ACKB (Output Mode 1) or STBB (Input Mode 1)
INTE A2	PC4	STBA (Input Mode 1 or Mode 2)
INTE A1	PC6	ACKA (Output Mode 1 or Mode 2)

FIGURE 17. INTERRUPT ENABLE FLAGS IN MODES 1 AND 2

Applications of the 82C55A

The 82C55A is a very powerful tool for interfacing peripheral equipment to the microcomputer system. It represents the optimum use of available pins and flexible enough to interface almost any I/O device without the need for additional external logic.

Each peripheral device in a microcomputer system usually has a "service routine" associated with it. The routine manages the software interface between the device and the CPU. The functional definition of the 82C55A is programmed by the I/O service routine and becomes an extension of the system software. By examining the I/O devices interface characteristics for both data transfer and timing, and matching this information to the examples and tables in the detailed operational description, a control word can easily be developed to initialize the 82C55A to exactly "fit" the application. Figures 18 through 24 present a few examples of typical applications of the 82C55A.

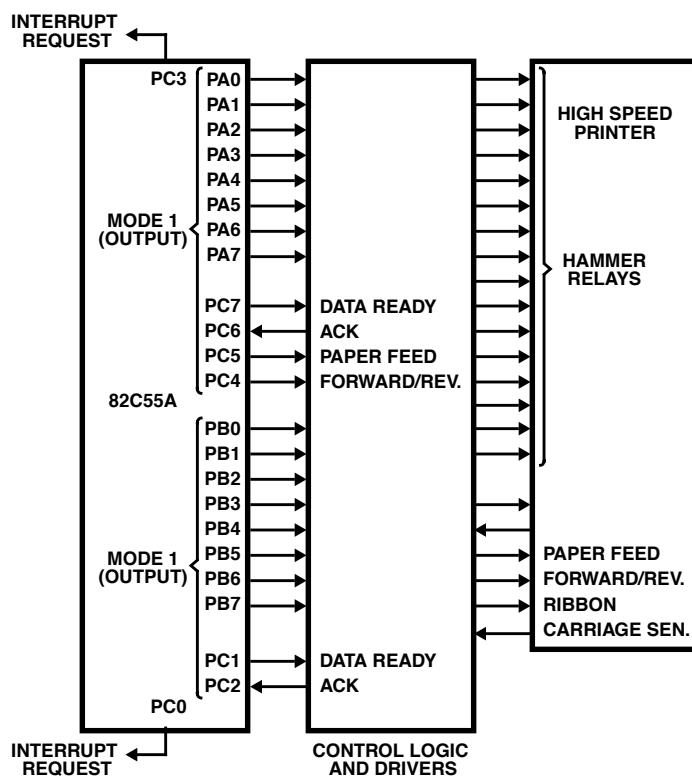
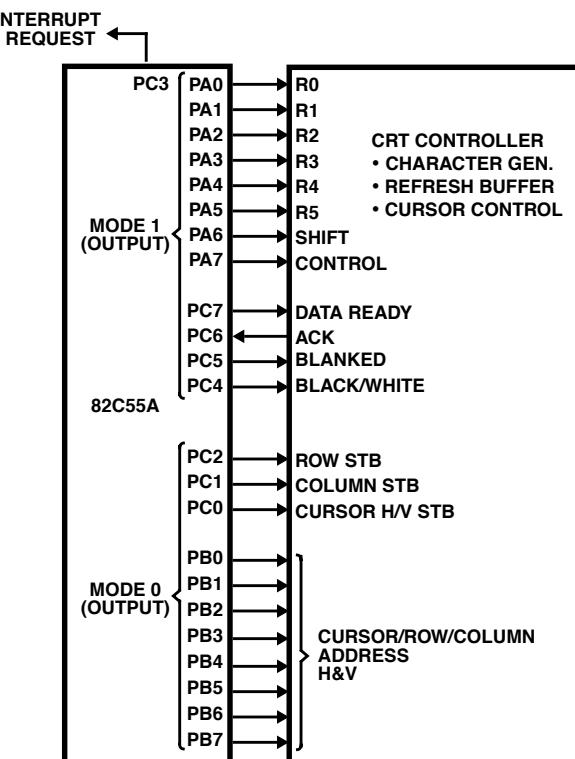
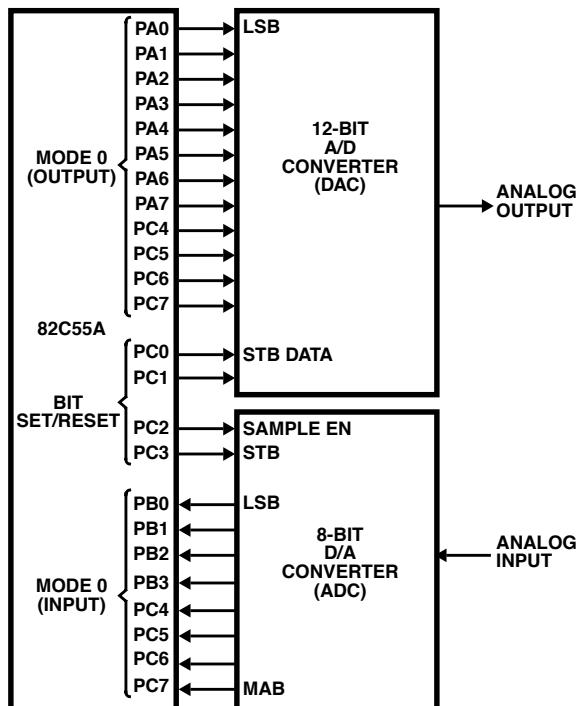
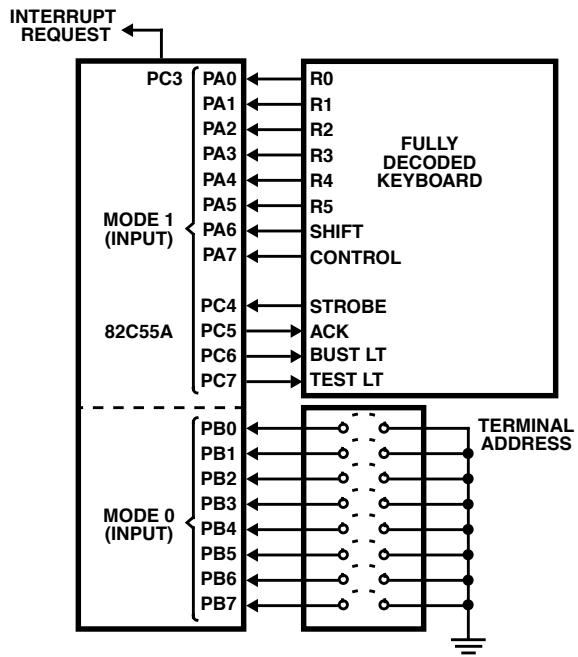
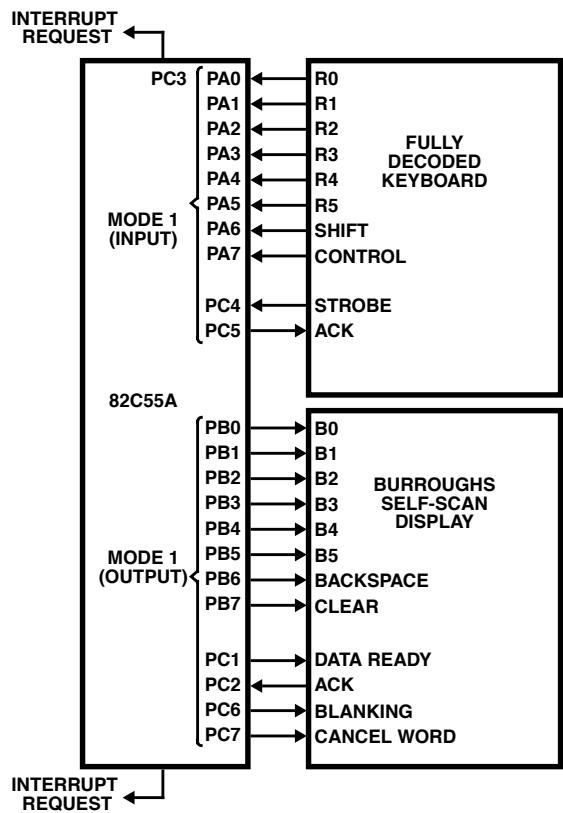


FIGURE 18. PRINTER INTERFACE



82C55A

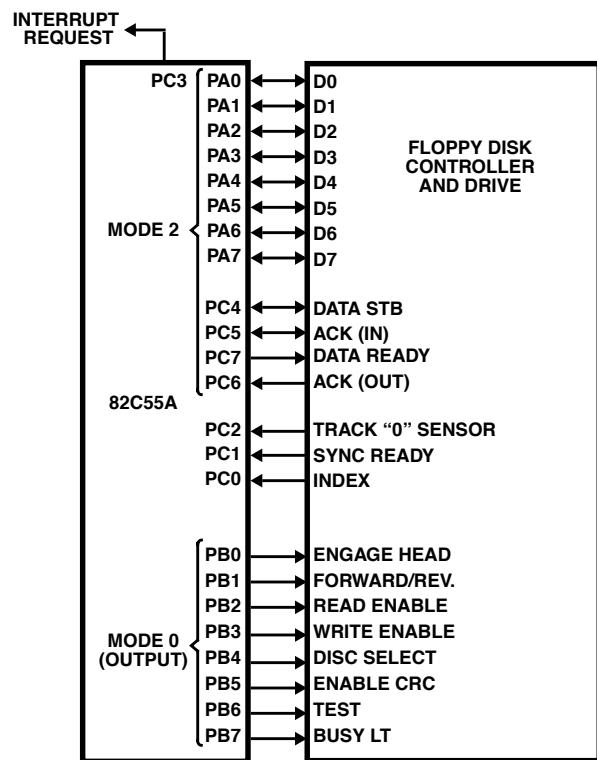


FIGURE 23. BASIC FLOPPY DISC INTERFACE

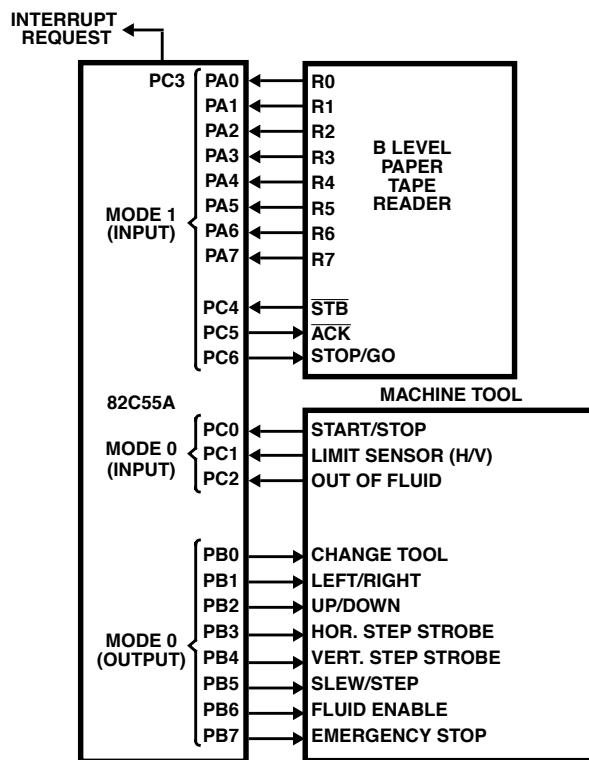


FIGURE 24. MACHINE TOOL CONTROLLER INTERFACE

Absolute Maximum Ratings $T_A = 25^\circ\text{C}$

Supply Voltage	+8.0V
Input, Output or I/O Voltage	GND-0.5V to $V_{CC}+0.5V$
ESD Classification	Class 1

Operating Conditions

Voltage Range	+4.5V to 5.5V
Operating Temperature Range	-40°C to 85°C
C82C55A	0°C to 70°C
I82C55A	-55°C to 125°C

Thermal Information

	θ_{JA}	θ_{JC}
CERDIP Package	50°C/W	10°C/W
CLCC Package	65°C/W	14°C/W
PDIP Package	50°C/W	N/A
PLCC Package	46°C/W	N/A
Maximum Storage Temperature Range	-65°C to 150°C	
Maximum Junction Temperature		
CDIP Package	175°C	
PDIP Package	150°C	
Maximum Lead Temperature (Soldering 10s)	300°C	(PLCC Lead Tips Only)

Die Characteristics

Gate Count	1000 Gates
------------------	------------

CAUTION: Stresses above those listed in "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress only rating and operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied.

NOTE:

1. θ_{JA} is measured with the component mounted on an evaluation PC board in free air.

Electrical Specifications $V_{CC} = 5.0V \pm 10\%$; $T_A = 0^\circ\text{C}$ to $+70^\circ\text{C}$ (C82C55A);
 $T_A = -40^\circ\text{C}$ to $+85^\circ\text{C}$ (I82C55A);
 $T_A = -55^\circ\text{C}$ to $+125^\circ\text{C}$ (M82C55A)

SYMBOL	PARAMETER	LIMITS		UNITS	TEST CONDITIONS
		MIN	MAX		
V_{IH}	Logical One Input Voltage	2.0 2.2	-	V	I82C55A, C82C55A, M82C55A
V_{IL}	Logical Zero Input Voltage	-	0.8	V	
V_{OH}	Logical One Output Voltage	3.0 $V_{CC}-0.4$	-	V	$I_{OH} = -2.5\text{mA}$, $I_{OH} = -100\mu\text{A}$
V_{OL}	Logical Zero Output Voltage	-	0.4	V	$I_{OL} + 2.5\text{mA}$
I_I	Input Leakage Current	-1.0	+1.0	μA	$V_{IN} = V_{CC}$ or GND, DIP Pins: 5, 6, 8, 9, 35, 36
I_O	I/O Pin Leakage Current	-10	+10	μA	$VO = V_{CC}$ or GND DIP Pins: 27 - 34
I_{BHH}	Bus Hold High Current	-50	-400	μA	$VO = 3.0\text{V}$. Ports A, B, C
I_{BHL}	Bus Hold Low Current	50	400	μA	$VO = 1.0\text{V}$. Port A ONLY
I_{DAR}	Darlington Drive Current	-2.5	Note 2, 4	mA	Ports A, B, C. Test Condition 3
I_{CCSB}	Standby Power Supply Current	-	10	μA	$V_{CC} = 5.5\text{V}$, $V_{IN} = V_{CC}$ or GND. Output Open
I_{CCOP}	Operating Power Supply Current	-	1	mA/MHz	$T_A = +25^\circ\text{C}$, $V_{CC} = 5.0\text{V}$, Typical (See Note 3)

NOTES:

2. No internal current limiting exists on Port Outputs. A resistor must be added externally to limit the current.
3. ICCOP = 1mA/MHz of Peripheral Read/Write cycle time. (Example: 1.0 μs I/O Read/Write cycle time = 1mA).
4. Tested as V_{OH} at -2.5mA.

Capacitance $T_A = 25^\circ\text{C}$

SYMBOL	PARAMETER	TYPICAL	UNITS	TEST CONDITIONS
C_{IN}	Input Capacitance	10	pF	FREQ = 1MHz, All Measurements are referenced to device GND
$C_{I/O}$	I/O Capacitance	20	pF	

82C55A

AC Electrical Specifications $V_{CC} = +5V \pm 10\%$, $GND = 0V$; $T_A = -55^{\circ}C$ to $+125^{\circ}C$ (M82C55A) (M82C55A-5);
 $T_A = -40^{\circ}C$ to $+85^{\circ}C$ (I82C55A) (I82C55A-5);
 $T_A = 0^{\circ}C$ to $+70^{\circ}C$ (C82C55A) (C82C55A-5)

SYMBOL	PARAMETER	82C55A-5		82C55A		UNITS	TEST CONDITIONS
		MIN	MAX	MIN	MAX		
READ TIMING							
(1) tAR	Address Stable Before \overline{RD}	0	-	0	-	ns	
(2) tRA	Address Stable After \overline{RD}	0	-	0	-	ns	
(3) tRR	\overline{RD} Pulse Width	250	-	150	-	ns	
(4) tRD	Data Valid From \overline{RD}	-	200	-	120	ns	1
(5) tDF	Data Float After \overline{RD}	10	75	10	75	ns	2
(6) tRV	Time Between \overline{RD} s and/or \overline{WR} s	300	-	300	-	ns	
WRITE TIMING							
(7) tAW	Address Stable Before \overline{WR}	0	-	0	-	ns	
(8) tWA	Address Stable After \overline{WR}	20	-	20	-	ns	
(9) tWW	\overline{WR} Pulse Width	100	-	100	-	ns	
(10) tDW	Data Valid to \overline{WR} High	100	-	100	-	ns	
(11) tWD	Data Valid After \overline{WR} High	30	-	30	-	ns	
OTHER TIMING							
(12) tWB	$\overline{WR} = 1$ to Output	-	350	-	350	ns	1
(13) tIR	Peripheral Data Before \overline{RD}	0	-	0	-	ns	
(14) tHR	Peripheral Data After \overline{RD}	0	-	0	-	ns	
(15) tAK	ACK Pulse Width	200	-	200	-	ns	
(16) tST	STB Pulse Width	100	-	100	-	ns	
(17) tPS	Peripheral Data Before STB High	20	-	20	-	ns	
(18) tPH	Peripheral Data After STB High	50	-	50	-	ns	
(19) tAD	ACK = 0 to Output	-	175	-	175	ns	1
(20) tKD	ACK = 1 to Output Float	20	250	20	250	ns	2
(21) tWOB	$\overline{WR} = 1$ to OBF = 0	-	150	-	150	ns	1
(22) tAOB	ACK = 0 to OBF = 1	-	150	-	150	ns	1
(23) tSIB	STB = 0 to IBF = 1	-	150	-	150	ns	1
(24) tRIB	$\overline{RD} = 1$ to IBF = 0	-	150	-	150	ns	1
(25) tTRIT	$\overline{RD} = 0$ to INTR = 0	-	200	-	200	ns	1
(26) tSIT	STB = 1 to INTR = 1	-	150	-	150	ns	1
(27) tAIT	ACK = 1 to INTR = 1	-	150	-	150	ns	1
(28) tWIT	$\overline{WR} = 0$ to INTR = 0	-	200	-	200	ns	1
(29) tRES	Reset Pulse Width	500	-	500	-	ns	1, (Note)

NOTE: Period of initial Reset pulse after power-on must be at least 50 μ sec. Subsequent Reset pulses may be 500ns minimum.

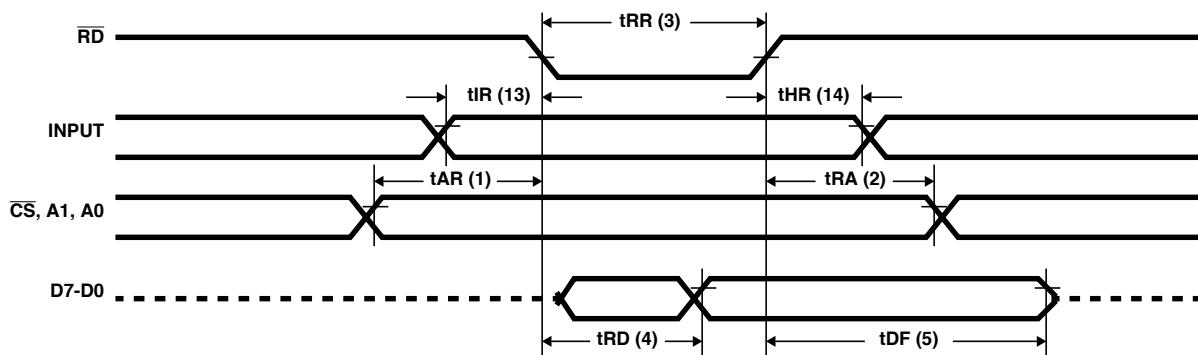
Timing Waveforms

FIGURE 25. MODE 0 (BASIC INPUT)

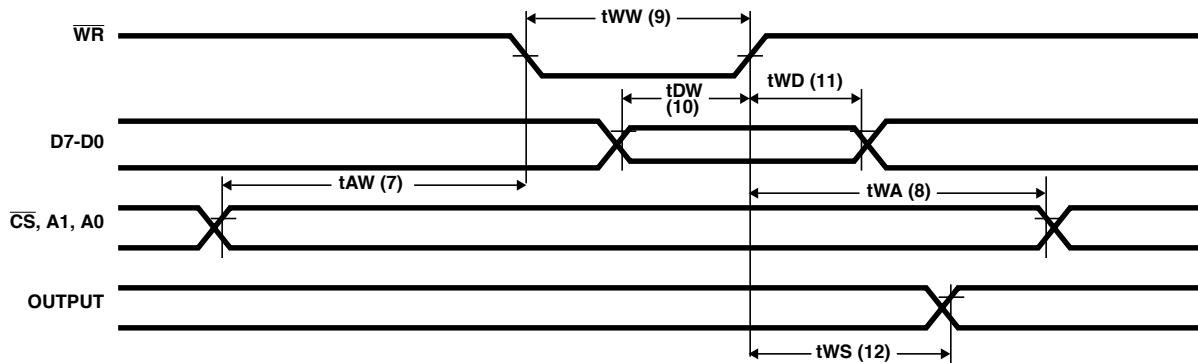


FIGURE 26. MODE 0 (BASIC OUTPUT)

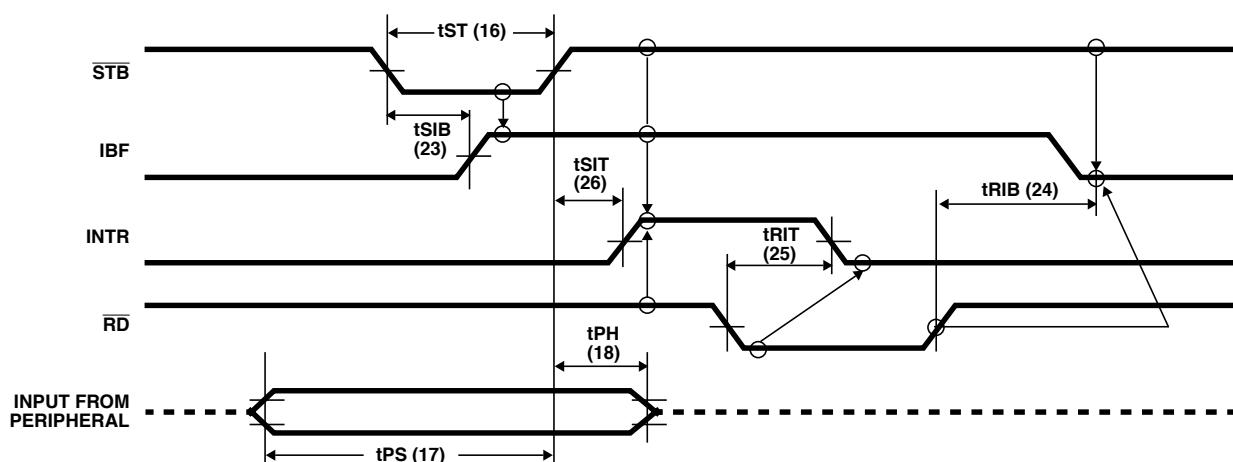


FIGURE 27. MODE 1 (STROBED INPUT)

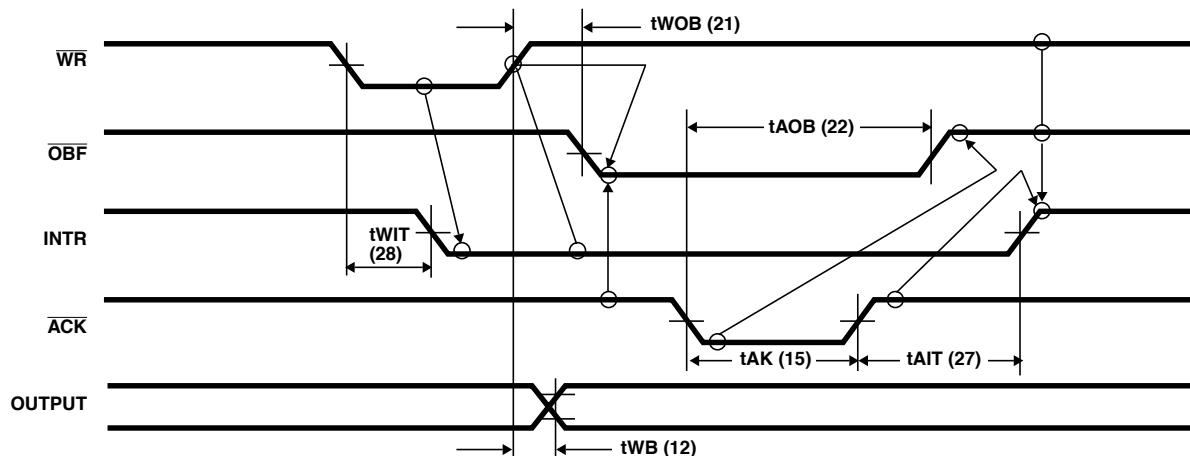
Timing Waveforms (Continued)

FIGURE 28. MODE 1 (STROBED OUTPUT)

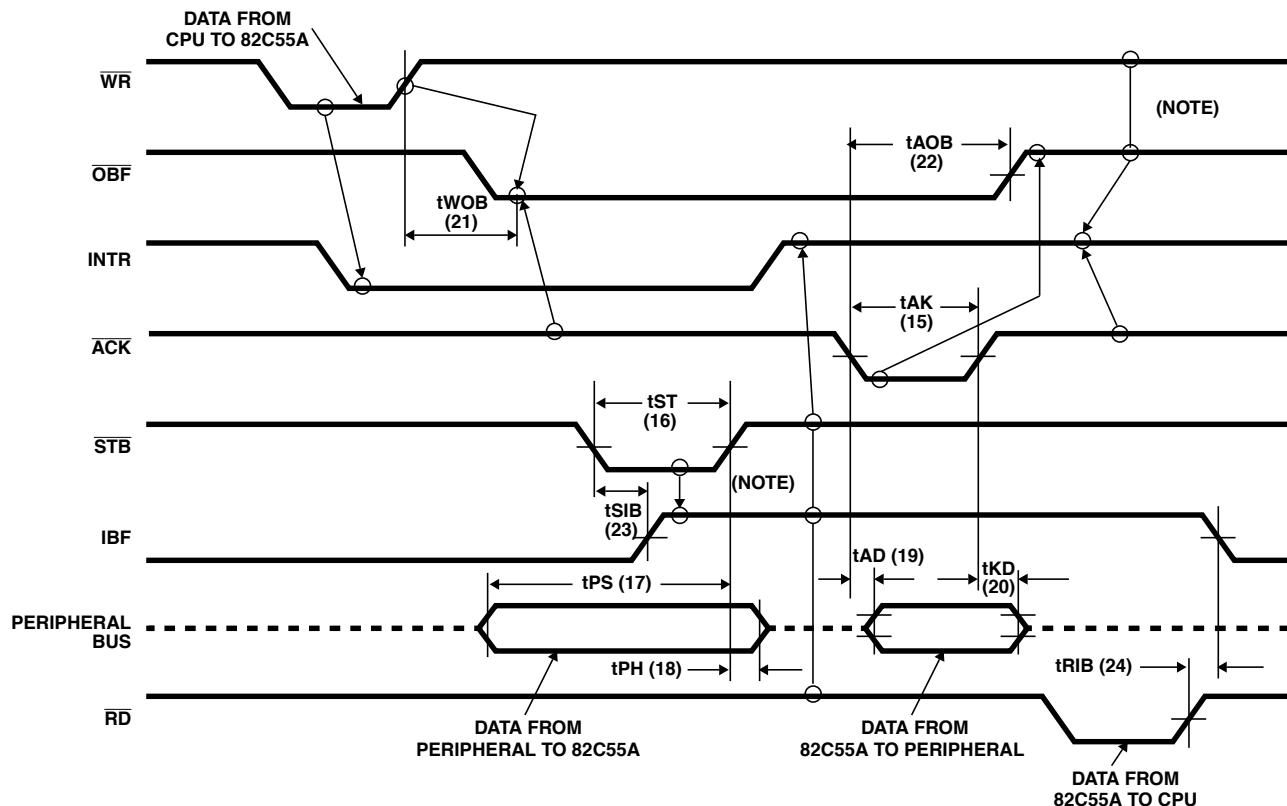


FIGURE 29. MODE 2 (BI-DIRECTIONAL)

NOTE: Any sequence where \overline{WR} occurs before \overline{ACK} and \overline{STB} occurs before \overline{RD} is permissible. ($INTR = IBF \cdot \overline{MASK} \cdot STB \cdot \overline{RD} \cdot \overline{OBF} \cdot \overline{MASK} \cdot ACK \cdot \overline{WR}$)

Timing Waveforms (Continued)

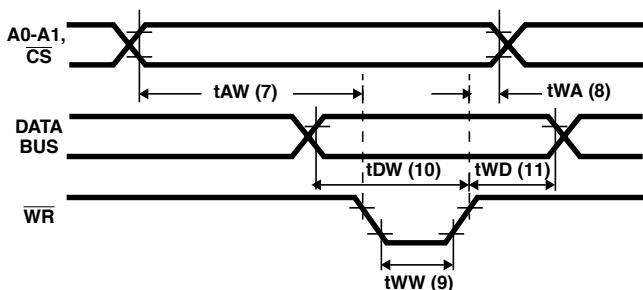


FIGURE 30. WRITE TIMING

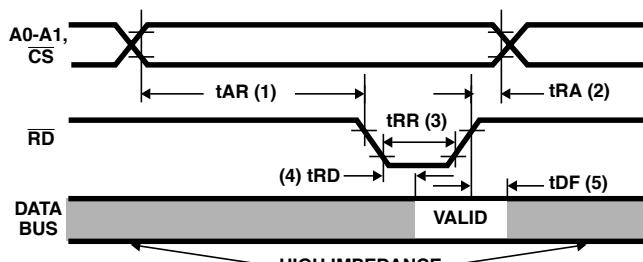
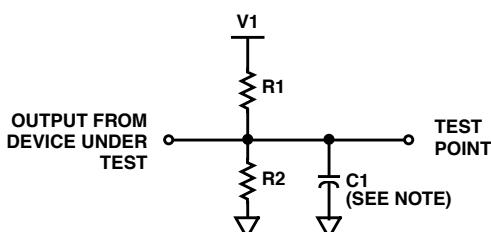


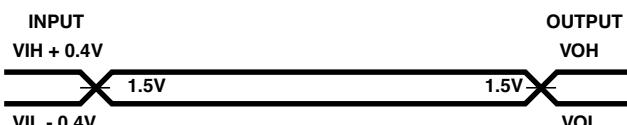
FIGURE 31. READ TIMING

AC Test Circuit



NOTE: Includes STRAY and JIG Capacitance

AC Testing Input, Output Waveforms



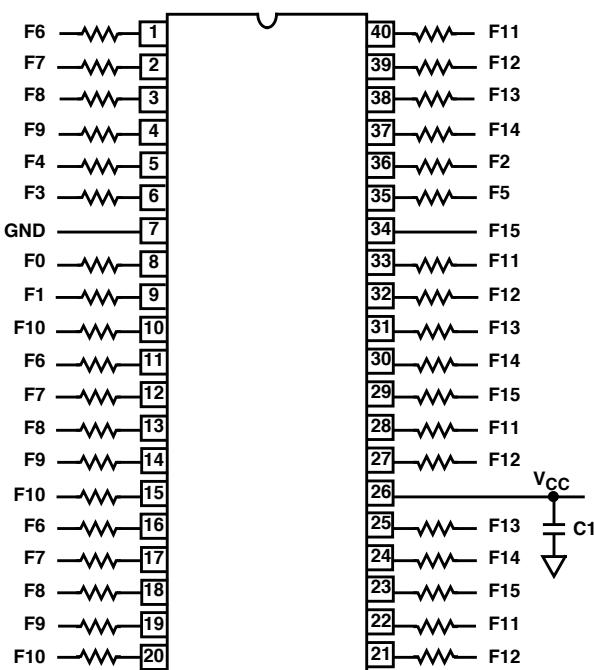
AC Testing: All AC Parameters tested as per test circuits. Input RISE and FALL times are driven at 1ns/V

TEST CONDITION DEFINITION TABLE

TEST CONDITION	V1	R1	R2	C1
1	1.7V	523Ω	Open	150pF
2	V_{CC}	2kΩ	1.7kΩ	50pF
3	1.5V	750Ω	Open	50pF

Burn-In Circuits

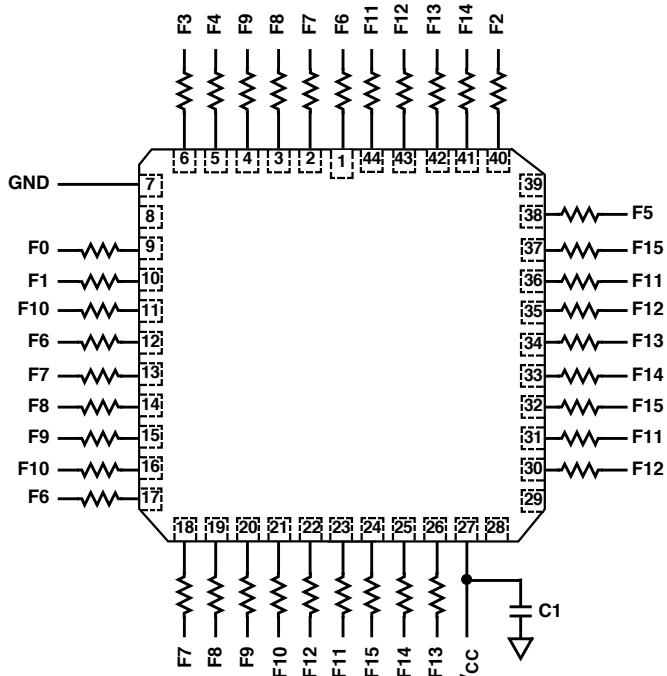
MD82C55A CERDIP



NOTES:

1. $V_{CC} = 5.5V \pm 0.5V$
 2. $VIH = 4.5V \pm 10\%$
 3. $VIL = -0.2V$ to $0.4V$
 4. $GND = 0V$

MR82C55A CLCC



NOTES:

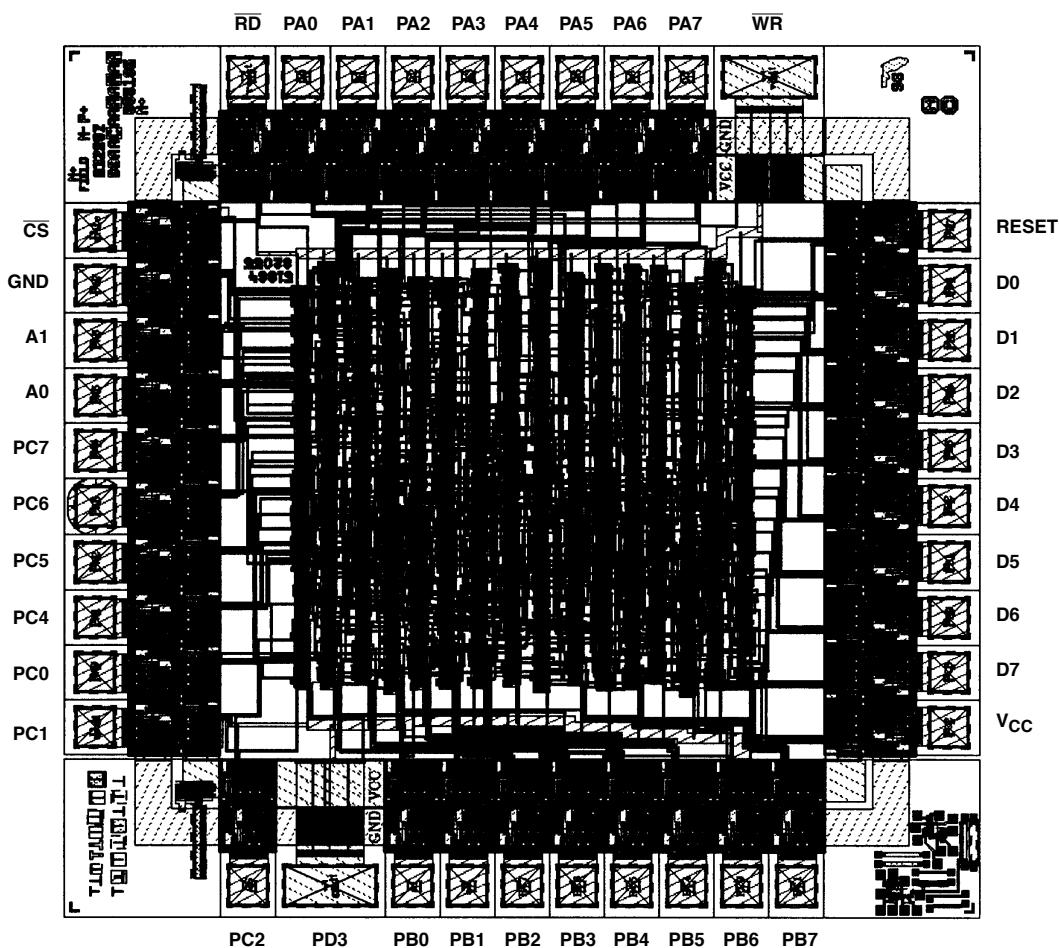
1. $C_1 = 0.01\mu F$ minimum
 2. All resistors are $47k\Omega \pm 5\%$
 3. $f_0 = 100\text{kHz} \pm 10\%$
 4. $f_1 = f_0 \div 2$; $f_2 = f_1 \div 2$; ...; $f_{15} = f_{14} \div 2$

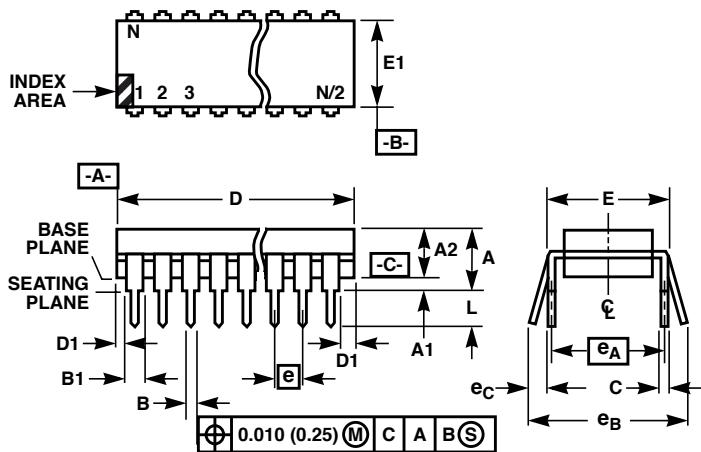
Die Characteristics**DIE DIMENSIONS:**95 x 100 x 19 \pm 1mils**METALLIZATION:**

Type: Silicon - Aluminum

Thickness: 11kÅ \pm 1kÅ**GLASSIVATION:**Type: SiO₂Thickness: 8kÅ \pm 1kÅ**WORST CASE CURRENT DENSITY:** 0.78×10^5 A/cm²***Metallization Mask Layout***

82C55A



Dual-In-Line Plastic Packages (PDIP)

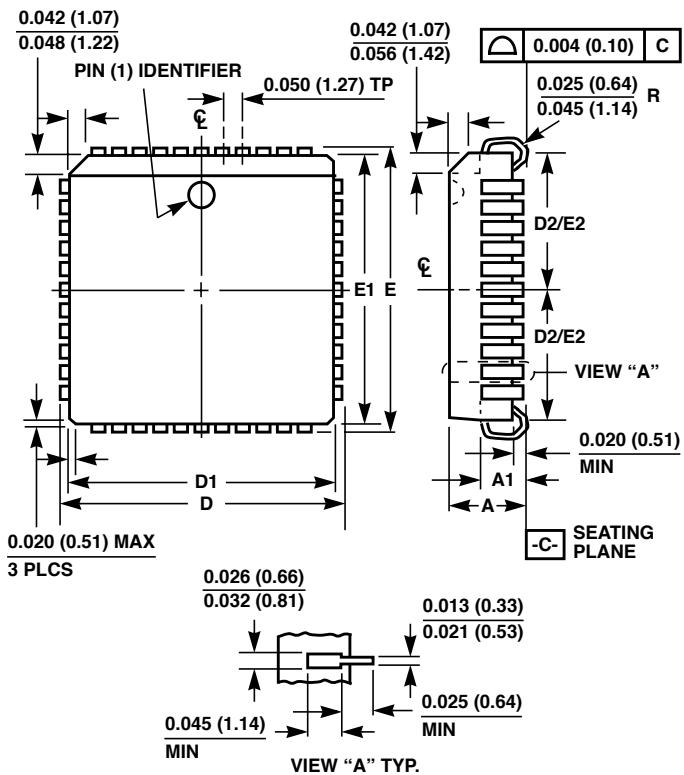
NOTES:

1. Controlling Dimensions: INCH. In case of conflict between English and Metric dimensions, the inch dimensions control.
2. Dimensioning and tolerancing per ANSI Y14.5M-1982.
3. Symbols are defined in the "MO Series Symbol List" in Section 2.2 of Publication No. 95.
4. Dimensions A, A1 and L are measured with the package seated in JEDEC seating plane gauge GS-3.
5. D, D1, and E1 dimensions do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.010 inch (0.25mm).
6. E and e_A are measured with the leads constrained to be perpendicular to datum $-C-$.
7. e_B and e_C are measured at the lead tips with the leads unconstrained. e_C must be zero or greater.
8. B1 maximum dimensions do not include dambar protrusions. Dambar protrusions shall not exceed 0.010 inch (0.25mm).
9. N is the maximum number of terminal positions.
10. Corner leads (1, N, N/2 and N/2 + 1) for E8.3, E16.3, E18.3, E28.3, E42.6 will have a B1 dimension of 0.030 - 0.045 inch (0.76 - 1.14mm).

**E40.6 (JEDEC MS-011-AC ISSUE B)
40 LEAD DUAL-IN-LINE PLASTIC PACKAGE**

SYMBOL	INCHES		MILLIMETERS		NOTES
	MIN	MAX	MIN	MAX	
A	-	0.250	-	6.35	4
A1	0.015	-	0.39	-	4
A2	0.125	0.195	3.18	4.95	-
B	0.014	0.022	0.356	0.558	-
B1	0.030	0.070	0.77	1.77	8
C	0.008	0.015	0.204	0.381	-
D	1.980	2.095	50.3	53.2	5
D1	0.005	-	0.13	-	5
E	0.600	0.625	15.24	15.87	6
E1	0.485	0.580	12.32	14.73	5
e	0.100 BSC		2.54 BSC		-
e_A	0.600 BSC		15.24 BSC		6
e_B	-	0.700	-	17.78	7
L	0.115	0.200	2.93	5.08	4
N	40		40		9

Rev. 0 12/93

Plastic Leaded Chip Carrier Packages (PLCC)

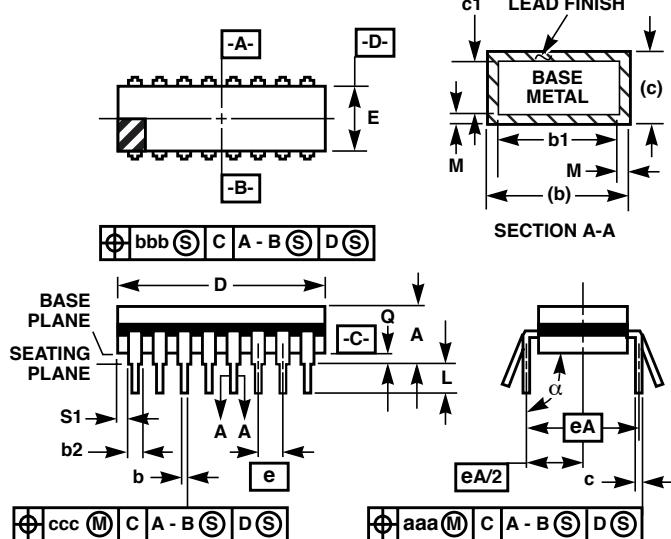
N44.65 (JEDEC MS-018AC ISSUE A)
44 LEAD PLASTIC LEADED CHIP CARRIER PACKAGE

SYM-BOL	INCHES		MILLIMETERS		NOTES
	MIN	MAX	MIN	MAX	
A	0.165	0.180	4.20	4.57	-
A1	0.090	0.120	2.29	3.04	-
D	0.685	0.695	17.40	17.65	-
D1	0.650	0.656	16.51	16.66	3
D2	0.291	0.319	7.40	8.10	4, 5
E	0.685	0.695	17.40	17.65	-
E1	0.650	0.656	16.51	16.66	3
E2	0.291	0.319	7.40	8.10	4, 5
N	44		44		6

Rev. 2 11/97

NOTES:

1. Controlling dimension: INCH. Converted millimeter dimensions are not necessarily exact.
2. Dimensions and tolerancing per ANSI Y14.5M-1982.
3. Dimensions D1 and E1 do not include mold protrusions. Allowable mold protrusion is 0.010 inch (0.25mm) per side. Dimensions D1 and E1 include mold mismatch and are measured at the extreme material condition at the body parting line.
4. To be measured at seating plane [-C-] contact point.
5. Centerline to be determined where center leads exit plastic body.
6. "N" is the number of terminal positions.

Ceramic Dual-In-Line Frit Seal Packages (CERDIP)

NOTES:

- Index area: A notch or a pin one identification mark shall be located adjacent to pin one and shall be located within the shaded area shown. The manufacturer's identification shall not be used as a pin one identification mark.
- The maximum limits of lead dimensions b and c or M shall be measured at the centroid of the finished lead surfaces, when solder dip or tin plate lead finish is applied.
- Dimensions b1 and c1 apply to lead base metal only. Dimension M applies to lead plating and finish thickness.
- Corner leads (1, N, N/2, and N/2+1) may be configured with a partial lead paddle. For this configuration dimension b3 replaces dimension b2.
- This dimension allows for off-center lid, meniscus, and glass overrun.
- Dimension Q shall be measured from the seating plane to the base plane.
- Measure dimension S1 at all four corners.
- N is the maximum number of terminal positions.
- Dimensioning and tolerancing per ANSI Y14.5M - 1982.
- Controlling dimension: INCH.

**F40.6 MIL-STD-1835 GDIP1-T40 (D-5, CONFIGURATION A)
40 LEAD CERAMIC DUAL-IN-LINE FRIT SEAL PACKAGE**

SYMBOL	INCHES		MILLIMETERS		NOTES
	MIN	MAX	MIN	MAX	
A	-	0.225	-	5.72	-
b	0.014	0.026	0.36	0.66	2
b1	0.014	0.023	0.36	0.58	3
b2	0.045	0.065	1.14	1.65	-
b3	0.023	0.045	0.58	1.14	4
c	0.008	0.018	0.20	0.46	2
c1	0.008	0.015	0.20	0.38	3
D	-	2.096	-	53.24	5
E	0.510	0.620	12.95	15.75	5
e	0.100 BSC		2.54 BSC		-
eA	0.600 BSC		15.24 BSC		-
eA/2	0.300 BSC		7.62 BSC		-
L	0.125	0.200	3.18	5.08	-
Q	0.015	0.070	0.38	1.78	6
S1	0.005	-	0.13	-	7
α	90°	105°	90°	105°	-
aaa	-	0.015	-	0.38	-
bbb	-	0.030	-	0.76	-
ccc	-	0.010	-	0.25	-
M	-	0.0015	-	0.038	2,3
N	40		40		8

Rev. 0 4/94

All Intersil semiconductor products are manufactured, assembled and tested under **ISO9000** quality systems certification.

Intersil products are sold by description only. Intersil Corporation reserves the right to make changes in circuit design and/or specifications at any time without notice. Accordingly, the reader is cautioned to verify that data sheets are current before placing orders. Information furnished by Intersil is believed to be accurate and reliable. However, no responsibility is assumed by Intersil or its subsidiaries for its use; nor for any infringements of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of Intersil or its subsidiaries.

For information regarding Intersil Corporation and its products, see web site <http://www.intersil.com>

Sales Office Headquarters**NORTH AMERICA**

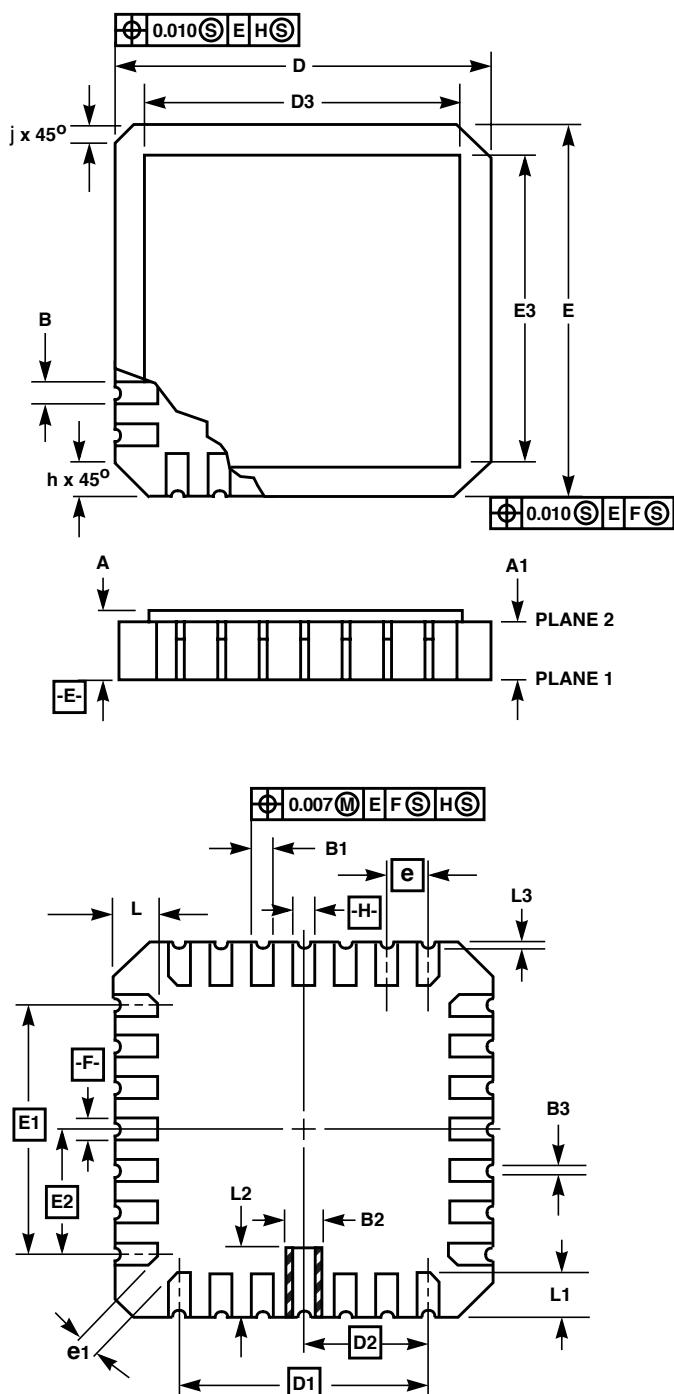
Intersil Corporation
P. O. Box 883, Mail Stop 53-204
Melbourne, FL 32902
TEL: (407) 724-7000
FAX: (407) 724-7240

EUROPE

Intersil SA
Mercure Center
100, Rue de la Fusée
1130 Brussels, Belgium
TEL: (32) 2.724.2111
FAX: (32) 2.724.22.05

ASIA

Intersil (Taiwan) Ltd.
Taiwan Limited
7F-6, No. 101 Fu Hsing North Road
Taipei, Taiwan
Republic of China
TEL: (886) 2 2716 9310
FAX: (886) 2 2715 3029

Ceramic Leadless Chip Carrier Packages (CLCC)

**J44.A MIL-STD-1835 CQCC1-N44 (C-5)
44 PAD CERAMIC LEADLESS CHIP CARRIER PACKAGE**

SYMBOL	INCHES		MILLIMETERS		NOTES
	MIN	MAX	MIN	MAX	
A	0.064	0.120	1.63	3.05	6, 7
A1	0.054	0.088	1.37	2.24	-
B	0.033	0.039	0.84	0.99	4
B1	0.022	0.028	0.56	0.71	2, 4
B2	0.072 REF		1.83 REF		-
B3	0.006	0.022	0.15	0.56	-
D	0.640	0.662	16.26	16.81	-
D1	0.500 BSC		12.70 BSC		-
D2	0.250 BSC		6.35 BSC		-
D3	-	0.662	-	16.81	2
E	0.640	0.662	16.26	16.81	-
E1	0.500 BSC		12.70 BSC		-
E2	0.250 BSC		6.35 BSC		-
E3	-	0.662	-	16.81	2
e	0.050 BSC		1.27 BSC		-
e1	0.015	-	0.38	-	2
h	0.040 REF		1.02 REF		5
j	0.020 REF		0.51 REF		5
L	0.045	0.055	1.14	1.40	-
L1	0.045	0.055	1.14	1.40	-
L2	0.075	0.095	1.90	2.41	-
L3	0.003	0.015	0.08	0.38	-
ND	11		11		3
NE	11		11		3
N	44		44		3

Rev. 0 5/18/94

NOTES:

1. Metallized castellations shall be connected to plane 1 terminals and extend toward plane 2 across at least two layers of ceramic or completely across all of the ceramic layers to make electrical connection with the optional plane 2 terminals.
2. Unless otherwise specified, a minimum clearance of 0.015 inch (0.38mm) shall be maintained between all metallized features (e.g., lid, castellations, terminals, thermal pads, etc.)
3. Symbol "N" is the maximum number of terminals. Symbols "ND" and "NE" are the number of terminals along the sides of length "D" and "E", respectively.
4. The required plane 1 terminals and optional plane 2 terminals (if used) shall be electrically connected.
5. The corner shape (square, notch, radius, etc.) may vary at the manufacturer's option, from that shown on the drawing.
6. Chip carriers shall be constructed of a minimum of two ceramic layers.
7. Dimension "A" controls the overall package thickness. The maximum "A" dimension is package height before being solder dipped.
8. Dimensioning and tolerancing per ANSI Y14.5M-1982.
9. Controlling dimension: INCH.