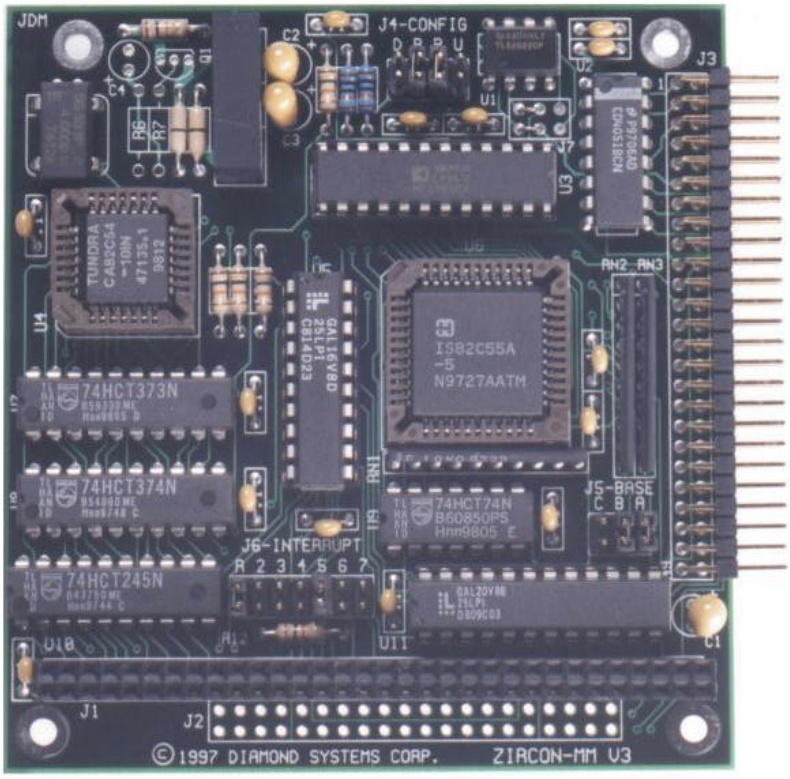
DIAMOND SYSTEMS CORPORATION

ZIRCON-MM

8-Bit Resolution Analog & Digital I/O

PC/104 Module

User Manual V2.11



Diamond Systems Corporation
8430-D Central Ave.
Newark, CA 94560
Tel (510) 456-7800
Fax (510) 45-7878
techinfo@diamondsystems.com
www.diamondsystems.com

TABLE OF CONTENTS

1. GENERAL DESCRIPTION

ZIRCON-MM is a PC/104 format I/O module with 8 analog inputs, 1 analog output, 24 digital I/O lines, 3 16-bit counter/timers (DX version), and 1 interrupt line. Both the analog input and output resolution are 8 bits (1/256). The analog circuitry requires no calibration over the lifetime of the product.

The digital I/O is based on an 82C55 chip with 3 8-bit ports and one configuration register. All I/O lines are TTL and CMOS compatible.

The counter/timers are based on an 82C54 chip with 3 16-bit counter/timers and one configuration register. Two counters are cascaded together to form a 32-bit counter/timer; The input to this combined counter/timer is driven by an on-board 4MHz clock oscillator. The third counter/timer is available at the I/O header. All I/O lines are TTL and CMOS compatible.

ZIRCON-MM is an 8-bit bus module and conforms to the physical and electrical specifications for 8-bit PC/104 modules published in the PC/104 Specification, Version 2.1, July 1994. As it is an 8-bit module, it does not contain the 16-bit expansion bus connector.

One 50-pin right-angle pin headers is provided for all I/O. The pinout of this header is defined at the end of this manual. All digital I/O signals are TTL compatible.

Three versions are available:

ZMM-LC: Contains no counter/timer; all analog I/O is unipolar (positive voltages) only.
ZMM-DX: Contains the counter/timer; analog I/O can be unipolar or bipolar.

ZIRCON-MM operates on +5V power supply only. Model ZMM-DX provides an on-board DC/DC converter to provide +/-5V for the analog circuitry. Model ZMM-LC uses the system +5V and ground for the analog supply.

2. ZIRCON-MM I/O HEADER PINOUT

J3 is the I/O header for Zircon-MM. It is a standard 50-pin dual row male header.

Analog In 7	1	2	Analog In 6
Analog In 5	3	4	Analog In 4
Analog In 3	5	6	Analog In 2
Analog In 1	7	8	Analog In 0
Analog Gnd	9	10	Analog Gnd
Analog Out	11	12	Analog Gnd
Analog Gnd	13	14	Analog Gnd
+5 Analog	15	16	-5 Analog
Analog Gnd	17	18	Analog Gnd
Digital Gnd	19	20	External Trigger
Gate 1/2	21	22	In 0
Gate 0	23	24	Out 0
A7	25	26	A6
A5	27	28	A4
A3	29	30	A2
A1	31	32	A0
C7	33	34	C6
C5	35	36	C4
C3	37	38	C2
C1	39	40	C0
B7	41	42	B6
B5	43	44	B4
B3	45	46	B2
B1	47	48	B0
+5 Digital	49	50	Digital Gnd

Definitions

Analog In 7 - 0	Analog input channels; range depends on configuration
Analog Out	Analog output channel; range depends on configuration
In 0, Gate 0, Out 0	Counter 0 input (clock), gate, and output pins
Gate 1/2	Counters 1 and 2 gate control pin
External Trigger	External Trigger for interrupt-based A/D conversions
A7 - A0	Digital I/O port A (82C55)
B7 - B0	Digital I/O port B (82C55)
C7 - C0	Digital I/O port C (82C55)
+5 Digital, Digital Ground	Access to main PC/104 power supply voltages
+5 Analog	ZMM-DX: +5V from the on-board analog power supply; ZMM-LC+5 Digital
-5 Analog	ZMM-DX: -5V from the on-board analog power supply; ZMM-LC: No connection
Analog Ground	Ground reference for the analog input/output signals

3. BASE ADDRESS CONFIGURATION

The base address is the address of the lowest location in the board's I/O map. Zircon-MM's base address is set with header **J5**, located at the lower right corner of the board. Seven different base addresses are available; see the table below. The default setting is 300 Hex. "Open" means an open position, and "Inst" means a position with a jumper installed. Note that 3 jumpers are never installed, only 0, 1 or 2.

Base Address		**Header J5 Position**		
Hex	**Decimal**	**C**	**B**	**A**
240	576	Open	Open	Open
280	640	Open	Open	Inst
2C0	704	Open	Inst	Open
300	768 (Default)	Open	Inst	Inst
340	832	Inst	Open	Open
380	896	Inst	Open	Inst
3C0	960	Inst	Inst	Open

4. INTERRUPT CONFIGURATION

Interrupt operation is configured with header **J6**, located in the lower center of the board.

One position is used for the PC/104 1KΩ pulldown resistor to enable interrupt sharing. This resistor should be connected if interrupts are used. Only one resistor should be connected to any interrupt line on the bus. Interrupts on Zircon-MM are driven by a tristate driver. When an interrupt is pending, the interrupt line is driven high, and when it is not pending, the output is in high-impedance mode, and the 1KΩ resistor pulls it down to a logic 0 state.

Once an interrupt request is generated by Zircon-MM, it is reset by reading from the A/D converter (read from base + 0 or base + 1).

Interrupts are enabled, disabled, and reset under software control. This header is used only to select the level and the resistor configuration.

Position	**Function**	**Open**	**Jumper**
R	1KΩ Resistor	No pulldown	Pulldown (max 1 per level)
2	IRQ2		
3	IRQ3	Install only one jumper in	
4	IRQ4	any of these 6 locations	
5	IRQ5	to select the interrupt level	
6	IRQ6		
7	IRQ7		

5. ANALOG I/O CONFIGURATION

This section describes how to configure the analog input and analog output full-scale ranges. The full-scale range for the analog inputs is the maximum range that an input signal can swing, still be readable by the board and not cause damage to the board. The full-scale range for the analog output is the maximum range of voltages that the board can generate on the analog output.

Model ZMM-LC has unipolar analog input and output ranges. This means that the analog inputs can only accept positive voltages, and the analog output can only create positive voltages. The input and output ranges are always the same on these models. The analog input/output range on these models can be set to either 0 - 1.25V or 0 - 2.5V.

Model ZMM-DX has both unipolar and bipolar ranges, meaning that the analog inputs can accept both negative and positive input voltages, and the analog output can create both negative and positive voltages. The analog input range on this model can be set to 0-1.25V, 0-2.5V, or 0-5V in unipolar mode and +/-1.25V, +/-2.5V, or +/-5V in bipolar mode.

Header **J4** at the top of the board is used to configure the analog input and output ranges. This 2x4 header had the labels “D R B U” above the four positions. The meanings of these positions are described below. However the simplest way to configure the board is just to refer to the table at the bottom of page 7.

Note: All settings are valid for model ZMM-DX. However, only two settings are valid for ZMM-LC. These settings are marked with an asterisk in the table.

Note: There is an additional header labeled J7 near J4. The jumpers on this header are set at the factory and should not be altered. For ZMM-LC, both jumpers should be in the right positions (over the middle and right pins). For ZMM-DX, both jumpers should be in the left positions (over the middle and left pins). Do not alter the positions of these jumpers, since doing so will prevent the board from functioning properly.

D **Double** the analog input range relative to the analog output range. For example, if the analog range is set to +/-2.5V, inserting a jumper in this position sets the analog input range to +/-5V. The analog output range, however, is still +/-2.5V. Doubling the analog input range is accomplished by inserting a voltage divider in the path of the analog input signal to divide the signal by 2. This means that the signal reaching the A/D converter is 1/2 the signal at the input to the board, so the input signal can swing twice as wide as the A/D full-scale range.

The D position can only be used on model ZMM-DX. On model ZMM-LC, this jumper has no effect, and the analog input and analog output ranges are always the same.

R **Range.** Installing a jumper in this position sets the range to 0-1.25V or +/-1.25V, and removing the jumper sets the range to 0-2.5V or +/-2.5V. On ZMM-DX, the range setting is further modified by the setting of D above.

B **Bipolar** range. On ZMM-DX, installing a jumper in this position will configure the analog inputs and output for bipolar ranges, depending on the position of D and R above.

WARNING: Do not install a jumper in this position on ZMM-LC, as the analog circuitry will not function properly.

U Unipolar range. On ZMM-DX, installing a jumper in this position will configure the analog inputs and output for unipolar ranges, depending on the positions of D and R above.

This jumper must be installed on models ZMM-LC and ZMM-LC2 for the board to function properly.

WARNING: Do not install a jumper in both B and U on ZMM-DX, since doing so will short the -5V supply to Ground.

The table below illustrates all valid settings for these jumpers and the resulting analog input and output ranges.

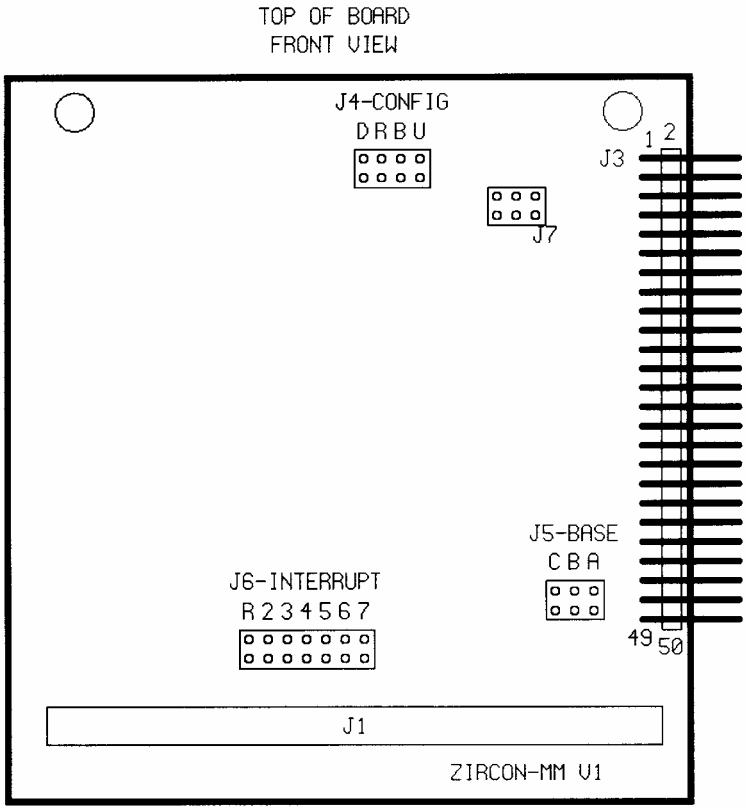
Analog I/O Configuration Settings

	D	R	B	U	Input range	Output range
*	Out	Out	Out	In	0-2.5V	0-2.5V
	Out	Out	In	Out	+/-2.5V	+/-2.5V
*	Out	In	Out	In	0-1.25V	0-1.25V
	Out	In	In	Out	+/-1.25V	+/-1.25V
	In	Out	Out	In	0-5V	0-2.5V
	In	Out	In	Out	+/-5V	+/-2.5V
	In	In	Out	In	0-2.5V	0-1.25V
	In	In	In	Out	+/-2.5V	+/-1.25V

* **These are the only valid settings for ZMM-LC.**

6. ZIRCON-MM BOARD DRAWING

This drawing will help to locate various key features on the board as described in the configuration sections above.



ZIRCON-MM CONFIGURATION DIAGRAM

J1: PC BUS CONNECTION
J3: I/O HEADER (2X25 PINS)
J4: ANALOG I/O RANGE CONFIGURATION
J5: BASE ADDRESS SELECTION
J6: INTERRUPT LEVEL SELECTION

7. REGISTER MAP

Base +	Write	Read
0	Analog Output	A/D register
1	Analog Output (duplicate)	A/D register (duplicate)
2	Start A/D conversion	--
3	Start A/D conversion (duplicate)	--
4	Control register	Status register
5	Control register (duplicate)	Status register (duplicate)
6	--	--
7	--	--
8	Digital I/O port A (82C55)	Digital I/O port A
9	Digital I/O port B	Digital I/O port B
10	Digital I/O port C	Digital I/O port C
11	Digital I/O configuration register	--
12	Counter/timer 0 data (82C54)	Counter/timer 0 data
13	Counter/timer 1 data	Counter/timer 1 data
14	Counter/timer 2 data	Counter/timer 2 data
15	Counter/timer configuration register	Counter/timer configuration register

A/D, D/A, Control, and Status register definitions are shown starting on the next page. For digital I/O and counter/timer register definitions, refer to the 82C55 and 82C54 datasheets, respectively. Counter/timers are not available on model ZMM-LC.

8. REGISTER DEFINITIONS

Base + 0 or 1: D/A Register (Write only)

Both these addresses map to the same physical register on the board.

Bit	7	6	5	4	3	2	1	0
Name	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**

D7 - D0 8-bit digital value for output; D7 = most significant bit, D0 = least significant bit

See **Section 11** for formulas to convert the desired output voltage to the corresponding input output code.

Base + 0 or 1: A/D Register (Read only)

Both these addresses map to the same physical register on the board.

Bit	7	6	5	4	3	2	1	0
Name	**A7**	**A6**	**A5**	**A4**	**A3**	**A2**	**A1**	**A0**

A7 - A0 8-bit digital value from A/D; D7 = most significant bit, D0 = least significant bit

See **Section 10** for formulas to convert this reading to the corresponding input voltage

Base + 2 or 3: Start A/D Conversion (Write only)

Both these addresses map to the same physical register on the board.
Writing to this register starts an A/D conversion. The value written does not matter.

Base + 4 or 5: Control register (Write only)

Both these addresses map to the same physical register on the board.

Bit	7	6	5	4	3	2	1	0
Name	**X**	**X**	**X**	**TrigE**	**TrigC**	**Ch2**	**Ch1**	**Ch0**

X Not used

TrigE Trigger enable for External Trigger
1 External Trigger generates A/D conversions
0 External Trigger is not used

TrigC Trigger enable for Counter 2 output (ZMM-DX only)
1 Counter 2 output generates A/D conversions
0 Counter 2 output is not used

Ch2 - Ch0 A/D channel select:

Ch2	Ch1	Ch0	Input channel
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

Base + 4 or 5: Status register (Read only)

Both these addresses map to the same physical register on the board.

Bit	7	6	5	4	3	2	1	0
Name	**Busy**	**Range**	**Int**	**TrigE**	**TrigC**	**Ch2**	**Ch1**	**Ch0**

Busy A/D converter status:
0 A/D converter busy (A/D conversion in progress)
1 A/D converter idle (A/D conversion complete)

Range A/D Range selection from configuration header J4; see page 7 for details on the meaning of this bit

TrigE, TrigC Readback of these control register bits (see above)

Ch2, Ch1, Ch0 Readback of these control register bits (see above)

Base+8 through Base+11 Read/Write 82C55 Digital I/O Chip Registers

These registers map directly to the 82C55 24-line digital I/O chip.

Base + n, Dir, Function	D7	D6	D5	D4	D3	D2	D1	D0
8, R/W, Port A	A7	A6	A5	A4	A3	A2	A1	A0
9, R/W, Port B	B7	B6	B5	B4	B3	B2	B1	B0
10, R/W, Port C	C7	C6	C5	C4	C3	C2	C1	C0
11, W, Config Register	1	ModeC	ModeA	DirA	DirCH	ModeB	DirB	DirCL

Configuration Register

The configuration register is programmed by writing to Base + 11 using the format below. Once you have set the port directions with this register, you can read and write to the ports as desired.

Bit No.	7	6	5	4	3	2	1	0
Name	1	ModeC	ModeA	DirA	DirCH	ModeB	DirB	DirCL

Definitions:

1 Bit 7 must be set to 1 to indicate port mode set operation.
DirA Direction control for bits A7 – A0: 0 = output, 1 = input
DirB Direction control for bits B7 – B0: 0 = output, 1 = input
DirCL Direction control for bits C3 – C0: 0 = output, 1 = input
DirCH Direction control for bits C7 – C4: 0 = output, 1 = input
ModeA, ModeB, ModeC I/O Mode for each port, 0 or 1

Here is a list of common configuration register values (others are possible):

Configuration Byte					
Hex	**Decimal**	**Port A**	**Port B**	**Port C (both halves)**	
9B	155	Input	Input	Input	(all ports input)
92	146	Input	Input	Output	
99	153	Input	Output	Input	
90	144	Input	Output	Output	
8B	139	Output	Input	Input	
82	130	Output	Input	Output	
89	137	Output	Output	Input	
80	128	Output	Output	Output	(all ports output)

9. DIGITAL I/O OPERATION

Addresses 8 – 11 on the board map to the 82C55 digital I/O chip on Zircon-MM. This chip provides 3 8-bit ports, called A, B, and C, for a total of 24 I/O lines. The chip contains four registers, 1 each for A, B, and C and 1 for control. These four registers are mapped to Ruby-MM-416's I/O map at base + 12 through base + 15. The 24 I/O lines are brought out to pins 25 - 48 on the user I/O header J3.

The I/O lines are standard CMOS logic with ±2.5mA output current capability. If more output current is required, use a buffer chip such as 74F244 or 74ACT244.

Each port's direction is determined through a control register. In normal "Mode 0" operation (the most common operating mode), ports A and B can be independently configured for input or output, and each half of port C can be independently configured for input or output. Output data is latched in the chip, but input data is not latched. When reading a port that is in input mode, the current logic levels of the port at the time of the read operation will be returned. To latch data into the port, you must use Modes 1 or Mode 2. In "Mode 1" and "Mode 2" operation, A and B are again configurable for input/output, but port C is reconfigured to provide control signals for transfer requests and data latching. A complete 82C55 datasheet is included at the end of this manual; please refer to it for programming details.

At power up, hardware reset, or board reset (write to base + 8), the 82C55 is set to all input and the data registers for ports A, B, and C are set to 0.

When a port is set to output mode, the contents of its output register are cleared to 0.

See page 12 for more detail on the 82C55 registers and operation, and see the 82C55 datasheet at the back of this manual for a complete description of the chip.

10. COUNTER/TIMERS (ZMM-DX ONLY)

Addresses 12 – 15 on model ZMM-DX map to an 82C54 counter/timer chip on the board. This chip provides 3 16-bit counter/timers. Each counter/timer has an input pin, a gate pin, and an output pin. The input pin responds to positive edges. A datasheet on the 82C54 is provided in the Appendix. This datasheet provides complete details on the operation and programming of the 82C54.

On ZMM-DX, counters 2 and 1 are cascaded together to create a 32-bit wide timer for A/D conversion timing and interrupts. An on-board oscillator provides a 4MHz clock that drives the input of counter 2. Counter 2's output is connected to counter 1's input. Each counter is generally programmed or Mode 2 operation (rate generator). In this way each counter is a divide-by-n counter, meaning that the output frequency is equal to the input frequency divided by the programmed 16-bit value (the divisor). The formula for determining the output frequency of this counter pair is:

Output frequency = 4Mz / (counter 2 divisor x counter 1 divisor)

The minimum divisor value for each counter is 2. Thus this counter pair can be used to generate a clock between 1MHz (4MHz / (2 x 2)) and .0009313Hz (4MHz / (65535 * 65535), or 1 pulse every 1,074 seconds). The output of counter 1 is generally used as a source for interrupt-based A/D conversions, although it can be used for other interrupt operations as well.

Counter 1's output is used as a trigger source for A/D conversions. The end of an A/D conversion can be used to generate interrupts on the bus. This sequence of operations should be remembered.

When using the counter/timers for interrupt operations (e.g. A/D conversions driven by interrupts), typically the first step is to program counters 2 and 1 for Mode 2 operation, selecting the desired interrupt rate by using the formula above. Then A/D conversions are initiated.

Counter 0 is completely independent. All three control pins are available on the I/O header. Input and gate pins are connected to +5V through 10KΩ pull-up resistors.

11. ANALOG I/O

Zircon-MM contains an 8-bit resolution A/D converter and an 8-bit resolution D/A converter on a single chip. The A/D converter input is fed by an 8 channel analog input multiplexor to expand the number of input channels to 8 single-ended analog voltage inputs. The D/A converter is nonmultiplexed, providing a single analog voltage output.

The A/D and D/A converter chip is factory calibrated for accuracy, and no user trims are provided (or available). Zircon-MM therefore requires no calibration over its entire operating lifetime.

The analog I/O can be set to several different ranges, depending on the model of Zircon-MM. For model ZMM-LC, the analog circuitry is powered from the system +5/Ground supplies, so all analog I/O is limited to positive voltages (0 - 2.5V or 0 - 1.25V ranges). On ZMM-DX, the analog circuitry is powered by an on-board DC/DC converter that provides dedicated +5/-5 supplies to the board. Using these supplies bipolar input and output ranges are available, including +/-2.5V and +/-1.25V.

In addition, on ZMM-DX the input impedance is improved by the addition of a precision op amp between the input multiplexor and the A/D converter. The addition of this op amp further enables the use of a voltage divider network to provide two additional input ranges: +/-5V and 0-5V. The voltage divider network, however, has the side effect of reducing the input impedance of the analog input channels.

On ZMM-LC, the D/A converter is unbuffered and directly drives the output pin on the user I/O header. This output can develop up to 1.25mA of current across a 2KΩ load (at a maximum output voltage of 2.5V). On ZMM-DX, the output is buffered through an op amp as described above, providing up to 30mA of drive current capability.

12. ANALOG INPUT FORMULAS

Zircon-MM uses an 8-bit A/D converter to read analog inputs. The output of this converter ranges from 0 to 255 in unipolar mode (true binary value) and from -128 to +127 in bipolar mode (twos complement value). The formulas below show how to convert the A/D converter reading to the corresponding input voltage. In bipolar mode, the following conversion must be applied before using these formulas, because the twos complement number is only an 8-bit wide value:

if A/D code >= 128 then A/D code = A/D code - 256

This will convert readings in the range +128 to +255 into their proper values of -128 to -1.

Since the A/D converter is 8 bits wide, it can resolve the input voltage signal (detect changes in input voltage) to within $1/2^8$, or 1/256, of the full input range. This minimum detectable voltage difference is equal to 1 least significant bit, or 1 LSB, because it is the change in input voltage required to cause the A/D reading to change by 1 LSB. The value of 1LSB for each input range is shown below.

Converting A/D Converter Values to Equivalent Input Voltages

Input range	1LSB	Conversion formula	Notes
+/-5V	39.1mV	Input voltage = 10V x A/D code / 256	ZMM-DX only
+/-2.5V	19.5mV	Input voltage = 5V x A/D code / 256	ZMM-DX only
+/-1.25V	9.8mV	Input voltage = 2.5V x A/D code / 256	ZMM-DX only
0 - 5V	19.5mV	Input voltage = 5V x A/D code / 256	ZMM-DX only
0 - 2.5V	9.8mV	Input voltage = 2.5V x A/D code / 256	
0 - 1.25V	4.9mV	Input voltage = 1.25V x A/D code / 256	

13. ANALOG OUTPUT FORMULAS

Zircon-MM uses an 8-bit D/A converter to generate analog outputs. In all cases except when the input voltage divider is used on ZMM-DX, the analog output range is identical to the analog input range.

Zircon-MM uses true binary coding for unipolar output ranges and offset binary coding for bipolar ranges. This means that writing a 0 to the D/A will always product an output at negative full scale, and writing 255 to the D/A will always product an output at positive full scale (minus 1 LSB). Because of offset binary coding, an offset must be added to the output code in bipolar mode to produce the desired output voltage.

The table below shows how to calculate the output code needed to generate the desired output voltage for each output voltage range on Zircon-MM. The output code is the 8-bit value written to the D/A converter.

Converting Desired Output Voltages to D/A Converter Codes

Output range	1LSB	Conversion formula	Notes
+/-2.5V	19.5mV	Output code = (Output voltage / 5V) x 256	ZMM-DX only
+/-1.25V	9.8mV	Output code = (Output voltage / 2.5V) x 256	ZMM-DX only
0 - 2.5V	9.8mV	Output code = (Output voltage / 2.5V) x 256	
0 - 1.25V	4.9mV	Output code = (Output voltage / 1.25V) x 256	

14. SPECIFICATIONS

All specifications at 25°C. A/D conversion rate on 386-25MHz CPU.

Analog Input

No. / type of channels	8 single-ended
Resolution	8 bits (1 part in 256)
A/D conversion time	2.6 µs max
Maximum conversion rate	40 KHz typical (interrupt-based single conversions) 64 KHz typical (interrupt-based 8-channel scan)
Relative accuracy	+/-1 LSB max
Offset error	+/-3 LSB max
Full-scale error	-4, +0 LSB max
Input current	+/-300 µA max (ZMM-LC) +/-60 nA max (ZMM-DX, D jumper not installed) +/-25 µA max (ZMM-DX, D jumper installed)

Analog Output

No. / type of channels	1, voltage output
Resolution	8 bits (1 part in 256)
Settling time	4 µs max to +/-1/2 LSB
Offset error	+/-2 LSB max
Full-scale error	+/-2 LSB max
Output current	+/-1.25mA max

Digital I/O

No. of lines	24 (using 82C55 chip)			
Compatibility	TTL / CMOS			
Pull-up resistors	10KΩ on each I/O line			
Output voltage	Logic 1:	3.0V min, 4.6V max;	Logic 0:	0.0V min, 0.4V max
Output current	Logic 1:	-2.5mA max;	Logic 0:	+2.5mA max
Input voltage	Logic 1:	2.0V min, 5.0V max;	Logic 0:	0.0V min, 0.8V max

Counter/Timers

No. of counter/timers	3
Width	16 bits
Compatibility	TTL / CMOS
Maximum clock frequency	10 MHz
On-board clock oscillator	4MHz
Configuration	Counters 1 and 2 cascaded for interrupts; Counter 0 free

General

Dimensions	3.55” x 3.775” (PC/104 standard)
Operating temperature	0 - 70°C
Power supply	+5VDC ±10%
Current consumption (typical, all outputs open)	ZMM-DX: 127mA ZMM-LC: 100mA
PC/104 bus	8-bit bus used; 16-bit header footprint on board for passthrough

HARRIS SEMICONDUCTOR

82C54

March 1997

CMOS Programmable Interval Timer

Features

- **8MHz to 12MHz Clock Input Frequency**
- **Compatible with NMOS 8254**
  - **Enhanced Version of NMOS 8253**
- **Three Independent 16-Bit Counters**
- **Six Programmable Counter Modes**
- **Status Read Back Command**
- **Binary or BCD Counting**
- **Fully TTL Compatible**
- **Single 5V Power Supply**
- **Low Power**
  - **ICCSB 10μA**
  - **ICCOP . . . . . .10mA at 8MHz**
- **Operating Temperature Ranges**
  - **C82C54 . . . . . .0^oC to $+70^oC$**
  - **I82C54 . . . . . .-40^oC to $+85^oC$**
  - **M82C54 . . . . . .-55^oC to $+125^oC$**

Description

The Harris 82C54 is a high performance CMOS Programmable Interval Timer manufactured using an advanced 2 micron CMOS process.

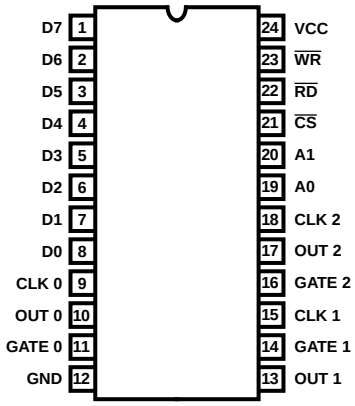
The 82C54 has three independently programmable and functional 16-bit counters, each capable of handling clock input frequencies of up to 8MHz (82C54) or 10MHz (82C54-10) or 12MHz (82C54-12).

The high speed and industry standard configuration of the 82C54 make it compatible with the Harris 80C86, 80C88, and 80C286 CMOS microprocessors along with many other industry standard processors. Six programmable timer modes allow the 82C54 to be used as an event counter, elapsed time indicator, programmable one-shot, and many other applications. Static CMOS circuit design insures low power operation.

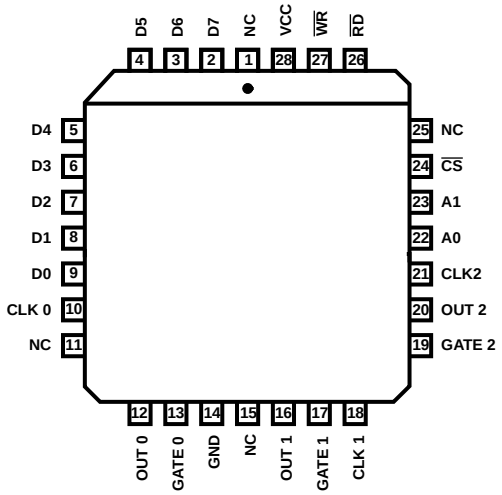
The Harris advanced CMOS process results in a significant reduction in power with performance equal to or greater than existing equivalent products.

Pinouts

82C54 (PDIP, CERDIP, SOIC)
TOP VIEW



82C54 (PLCC/CLCC)
TOP VIEW



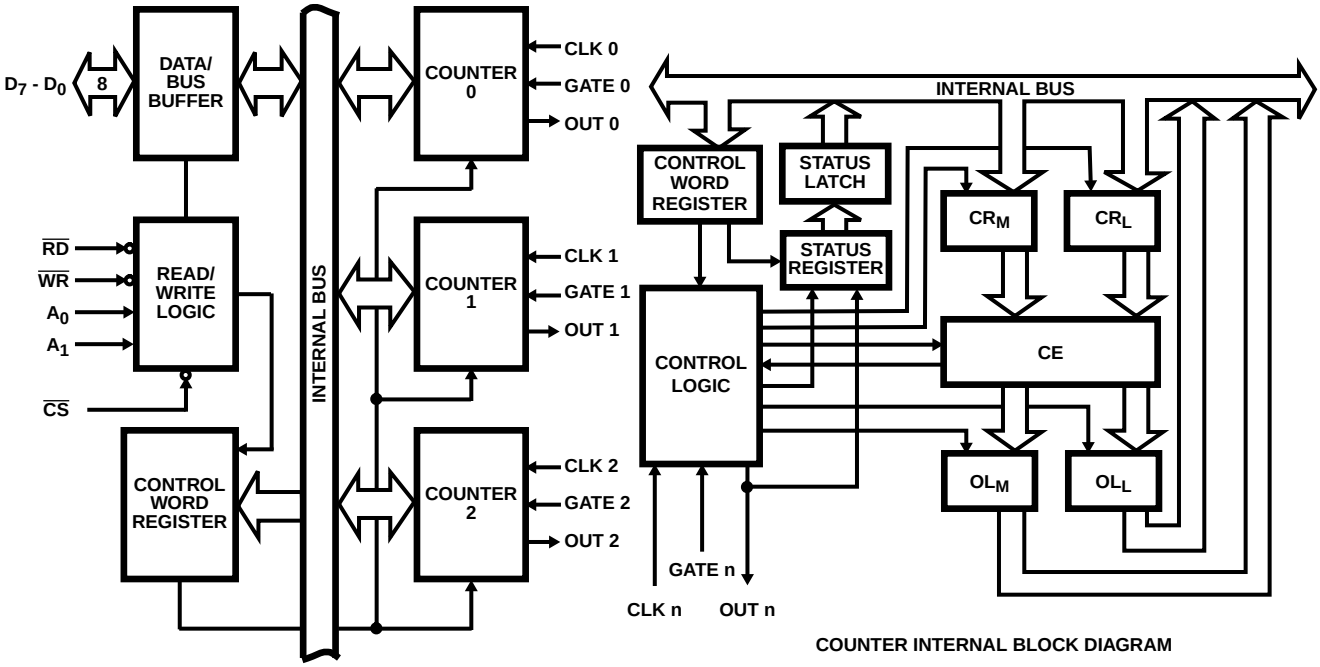
CAUTION: These devices are sensitive to electrostatic discharge. Users should follow proper IC Handling Procedures.

File Number **2970.1**

Ordering Information

PART NUMBERS			TEMPERATURE RANGE	PACKAGE	PKG. NO.
8MHz	10MHz	12MHz			
CP82C54	CP82C54-10	CP82C54-12	0^oC to $+70^oC$	24 Lead PDIP	E24.6
IP82C54	IP82C54-10	IP82C54-12	-40^oC to $+85^oC$	24 Lead PDIP	E24.6
CS82C54	CS82C54-10	CS82C54-12	0^oC to $+70^oC$	28 Lead PLCC	N28.45
IS82C54	IS82C54-10	IS82C54-12	-40^oC to $+85^oC$	28 Lead PLCC	N28.45
CD82C54	CD82C54-10	CD82C54-12	0^oC to $+70^oC$	24 Lead CERDIP	F24.6
ID82C54	ID82C54-10	ID82C54-12	-40^oC to $+85^oC$	24 Lead CERDIP	F24.6
MD82C54/B	MD82C54-10/B	MD82C54-12/B	-55^oC to $+125^oC$	24 Lead CERDIP	F24.6
MR82C54/B	MR82C54-10/B	MR82C54-12/B	-55^oC to $+125^oC$	28 Lead CLCC	J28.A
SMD # 8406501JA	-	8406502JA	-55^oC to $+125^oC$	24 Lead CERDIP	F24.6
SMD# 84065013A	-	84065023A	-55^oC to $+125^oC$	28 Lead CLCC	J28.A
CM82C54	CM82C54-10	CM82C54-12	0^oC to $+70^oC$	24 Lead SOIC	M24.3

Functional Diagram



COUNTER INTERNAL BLOCK DIAGRAM

Pin Description

SYMBOL	DIP PIN NUMBER	TYPE	DEFINITION
D7 - D0	1 - 8	I/O	DATA: Bi-directional three-state data bus lines, connected to system data bus.
CLK 0	9	I	CLOCK 0: Clock input of Counter 0.
OUT 0	10	O	OUT 0: Output of Counter 0.
GATE 0	11	I	GATE 0: Gate input of Counter 0.
GND	12		GROUND: Power supply connection.
OUT 1	13	O	OUT 1: Output of Counter 1.
GATE 1	14	I	GATE 1: Gate input of Counter 1.
CLK 1	15	I	CLOCK 1: Clock input of Counter 1.
GATE 2	16	I	GATE 2: Gate input of Counter 2.
OUT 2	17	O	OUT 2: Output of Counter 2.

Pin Description (Continued)

SYMBOL	DIP PIN NUMBER	TYPE	DEFINITION
CLK 2	18	I	CLOCK 2: Clock input of Counter 2.
A0, A1	19 - 20	I	ADDRESS: Select inputs for one of the three counters or Control Word Register for read/write operations. Normally connected to the system address bus.

A1	A0	SELECTS
0	0	Counter 0
0	1	Counter 1
1	0	Counter 2
1	1	Control Word Register

$\overline{CS}$	21	I	CHIP SELECT: A low on this input enables the 82C54 to respond to $\overline{RD}$ and $\overline{WR}$ signals. $\overline{RD}$ and $\overline{WR}$ are ignored otherwise.
$\overline{RD}$	22	I	READ: This input is low during CPU read operations.
$\overline{WR}$	23	I	WRITE: This input is low during CPU write operations.
V_{CC}	24		V_{CC}: The +5V power supply pin. A 0.1µF capacitor between pins VCC and GND is recommended for decoupling.

Functional Description

General

The 82C54 is a programmable interval timer/counter designed for use with microcomputer systems. It is a general purpose, multi-timing element that can be treated as an array of I/O ports in the system software.

The 82C54 solves one of the most common problems in any microcomputer system, the generation of accurate time delays under software control. Instead of setting up timing loops in software, the programmer configures the 82C54 to match his requirements and programs one of the counters for the desired delay. After the desired delay, the 82C54 will interrupt the CPU. Software overhead is minimal and variable length delays can easily be accommodated.

Some of the other computer/timer functions common to microcomputers which can be implemented with the 82C54 are:

- Real time clock
- Event counter
- Digital one-shot
- Programmable rate generator
- Square wave generator
- Binary rate multiplier
- Complex waveform generator
- Complex motor controller

Data Bus Buffer

This three-state, bi-directional, 8-bit buffer is used to interface the 82C54 to the system bus (see Figure 1).

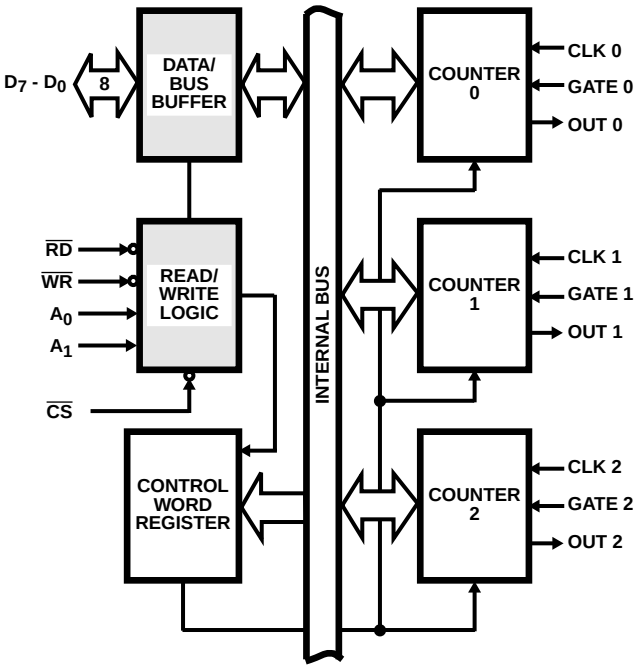


FIGURE 1. DATA BUS BUFFER AND READ/WRITE LOGIC FUNCTIONS

Read/Write Logic

The Read/Write Logic accepts inputs from the system bus and generates control signals for the other functional blocks of the 82C54. A1 and A0 select one of the three counters or the Control Word Register to be read from/written into. A “low” on the $\overline{RD}$ input tells the 82C54 that the CPU is reading one of the counters. A “low” on the $\overline{WR}$ input tells the 82C54 that the CPU is writing either a Control Word or an initial count. Both $\overline{RD}$ and $\overline{WR}$ are qualified by $\overline{CS}$; $\overline{RD}$ and $\overline{WR}$ are ignored unless the 82C54 has been selected by holding $\overline{CS}$ low.

Control Word Register

The Control Word Register (Figure 2) is selected by the Read/Write Logic when A1, A0 = 11. If the CPU then does a write operation to the 82C54, the data is stored in the Control Word Register and is interpreted as a Control Word used to define the Counter operation.

The Control Word Register can only be written to; status information is available with the Read-Back Command.

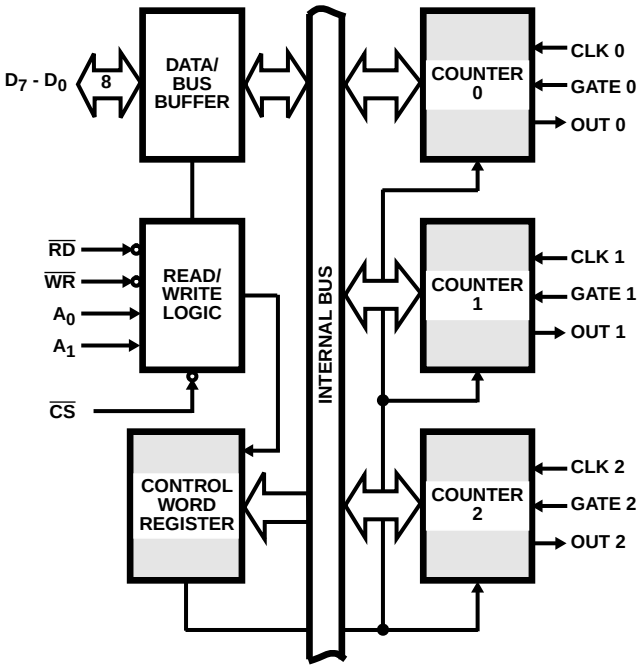


FIGURE 2. CONTROL WORD REGISTER AND COUNTER FUNCTIONS

Counter 0, Counter 1, Counter 2

These three functional blocks are identical in operation, so only a single Counter will be described. The internal block diagram of a signal counter is shown in Figure 3. The counters are fully independent. Each Counter may operate in a different Mode.

The Control Word Register is shown in the figure; it is not part of the Counter itself, but its contents determine how the Counter operates.

The status register, shown in the figure, when latched, contains the current contents of the Control Word Register and status of the output and null count flag. (See detailed explanation of the Read-Back command.)

The actual counter is labeled CE (for Counting Element). It is a 16-bit presettable synchronous down counter.

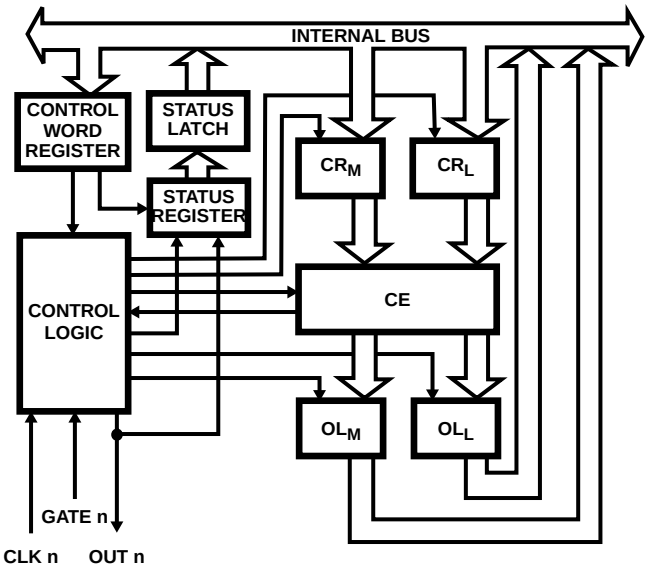


FIGURE 3. COUNTER INTERNAL BLOCK DIAGRAM

OLM and OLL are two 8-bit latches. OL stands for “Output Latch”; the subscripts M and L for “Most significant byte” and “Least significant byte”, respectively. Both are normally referred to as one unit and called just OL. These latches normally “follow” the CE, but if a suitable Counter Latch Command is sent to the 82C54, the latches “latch” the present count until read by the CPU and then return to “following” the CE. One latch at a time is enabled by the counter’s Control Logic to drive the internal bus. This is how the 16-bit Counter communicates over the 8-bit internal bus. Note that the CE itself cannot be read; whenever you read the count, it is the OL that is being read.

Similarly, there are two 8-bit registers called CRM and CRL (for “Count Register”). Both are normally referred to as one unit and called just CR. When a new count is written to the Counter, the count is stored in the CR and later transferred to the CE. The Control Logic allows one register at a time to be loaded from the internal bus. Both bytes are transferred to the CE simultaneously. CRM and CRL are cleared when the Counter is programmed for one byte counts (either most significant byte only or least significant byte only) the other byte will be zero. Note that the CE cannot be written into; whenever a count is written, it is written into the CR.

The Control Logic is also shown in the diagram. CLK n, GATE n, and OUT n are all connected to the outside world through the Control Logic.

82C54 System Interface

The 82C54 is treated by the system software as an array of peripheral I/O ports; three are counters and the fourth is a control register for MODE programming.

Basically, the select inputs A0, A1 connect to the A0, A1 address bus signals of the CPU. The $\overline{CS}$ can be derived directly from the address bus using a linear select method or it can be connected to the output of a decoder.

Operational Description

General

After power-up, the state of the 82C54 is undefined. The Mode, count value, and output of all Counters are undefined.

How each Counter operates is determined when it is programmed. Each Counter must be programmed before it can be used. Unused counters need not be programmed.

Programming the 82C54

Counters are programmed by writing a Control Word and then an initial count.

All Control Words are written into the Control Word Register, which is selected when A1, A0 = 11. The Control Word specifies which Counter is being programmed.

By contrast, initial counts are written into the Counters, not the Control Word Register. The A1, A0 inputs are used to select the Counter to be written into. The format of the initial count is determined by the Control Word used.

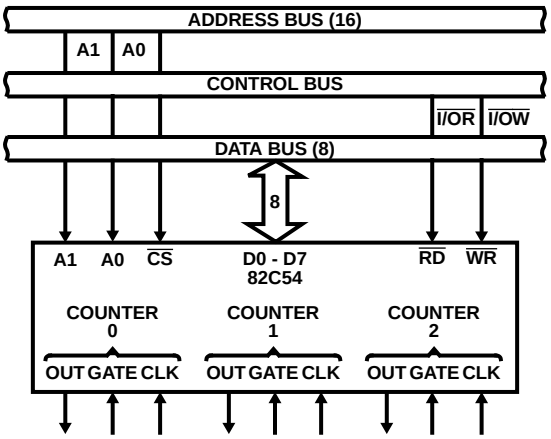


FIGURE 4. 82C54 SYSTEM INTERFACE

Write Operations

The programming procedure for the 82C54 is very flexible. Only two conventions need to be remembered:

1. For Each Counter, the Control Word must be written before the initial count is written.
2. The initial count must follow the count format specified in the Control Word (least significant byte only, most significant byte only, or least significant byte and then most significant byte).

Since the Control Word Register and the three Counters have separate addresses (selected by the A1, A0 inputs), and each Control Word specifies the Counter it applies to (SC0, SC1 bits), no special instruction sequence is required. Any programming sequence that follows the conventions above is acceptable.

Control Word Format

A1, A0 = 11; $\overline{CS}$ = 0; $\overline{RD}$ = 1; $\overline{WR}$ = 0

D7	D6	D5	D4	D3	D2	D1	D0
SC1	SC0	RW1	RW0	M2	M1	M0	BCD

SC - Select Counter

SC1	SC0	
0	0	Select Counter 0
0	1	Select Counter 1
1	0	Select Counter 2
1	1	Read-Back Command (See Read Operations)

RW - Read/Write

RW1	RW0	
0	0	Counter Latch Command (See Read Operations)
0	1	Read/Write least significant byte only.
1	0	Read/Write most significant byte only.
1	1	Read/Write least significant byte first, then most significant byte.

M - Mode

M2	M1	M0	
0	0	0	Mode 0
0	0	1	Mode 1
X	1	0	Mode 2
X	1	1	Mode 3
1	0	0	Mode 4
1	0	1	Mode 5

BCD - Binary Coded Decimal

0	Binary Counter 16-bit
1	Binary Coded Decimal (BCD) Counter (4 Decades)

NOTE: Don't Care bits (X) should be 0 to insure compatibility with future products.

Possible Programming Sequence

	A1	A0
Control Word - Counter 0	1	1
LSB of Count - Counter 0	0	0
MSB of Count - Counter 0	0	0
Control Word - Counter 1	1	1
LSB of Count - Counter 1	0	1
MSB of Count - Counter 1	0	1
Control Word - Counter 2	1	1
LSB of Count - Counter 2	1	0
MSB of Count - Counter 2	1	0

Possible Programming Sequence

	A1	A0
Control Word - Counter 0	1	1
Control Word - Counter 1	1	1
Control Word - Counter 2	1	1
LSB of Count - Counter 2	1	0

Possible Programming Sequence (Continued)

	A1	A0
LSB of Count - Counter 1	0	1
LSB of Count - Counter 0	0	0
MSB of Count - Counter 0	0	0
MSB of Count - Counter 1	0	1
MSB of Count - Counter 2	1	0

Possible Programming Sequence

	A1	A0
Control Word - Counter 2	1	1
Control Word - Counter 1	1	1
Control Word - Counter 0	1	1
LSB of Count - Counter 2	1	0
MSB of Count - Counter 2	1	0
LSB of Count - Counter 1	0	1
MSB of Count - Counter 1	0	1
LSB of Count - Counter 0	0	0
MSB of Count - Counter 0	0	0

Possible Programming Sequence

	A1	A0
Control Word - Counter 1	1	1
Control Word - Counter 0	1	1
LSB of Count - Counter 1	0	1
Control Word - Counter 2	1	1
LSB of Count - Counter 0	0	0
MSB of Count - Counter 1	0	1
LSB of Count - Counter 2	1	0
MSB of Count - Counter 0	0	0
MSB of Count - Counter 2	1	0

NOTE: In all four examples, all counters are programmed to Read/Write two-byte counts. These are only four of many programming sequences.

A new initial count may be written to a Counter at any time without affecting the Counter's programmed Mode in any way. Counting will be affected as described in the Mode definitions. The new count must follow the programmed count format.

If a Counter is programmed to read/write two-byte counts, the following precaution applies. A program must not transfer control between writing the first and second byte to another routine which also writes into that same Counter. Otherwise, the Counter will be loaded with an incorrect count.

Read Operations

It is often desirable to read the value of a Counter without disturbing the count in progress. This is easily done in the 82C54.

There are three possible methods for reading the Counters. The first is through the Read-Back command, which is explained later. The second is a simple read operation of the Counter, which is selected with the A1, A0 inputs. The only requirement is that the CLK input of the selected Counter must be inhibited by using either the GATE input or external logic. Otherwise, the count may be in process of changing when it is read, giving an undefined result.

Counter Latch Command

The other method for reading the Counters involves a special software command called the "Counter Latch Command". Like a Control Word, this command is written to the Control Word Register, which is selected when A1, A0 = 11. Also, like a Control Word, the SC0, SC1 bits select one of the three Counters, but two other bits, D5 and D4, distinguish this command from a Control Word.

A1, A0 = 11; $\overline{CS}$ = 0; $\overline{RD}$ = 1; $\overline{WR}$ = 0

D7	D6	D5	D4	D3	D2	D1	D0
SC1	SC0	0	0	X	X	X	X

SC1, SC0 - specify counter to be latched

SC1	SC0	COUNTER
0	0	0
0	1	1
1	0	2
1	1	Read-Back Command

D5, D4 - 00 designates Counter Latch Command, X - Don't Care.
NOTE: Don't Care bits (X) should be 0 to insure compatibility with future products.

The selected Counter's output latch (OL) latches the count when the Counter Latch Command is received. This count is held in the latch until it is read by the CPU (or until the Counter is reprogrammed). The count is then unlatched automatically and the OL returns to "following" the counting element (CE). This allows reading the contents of the Counters "on the fly" without affecting counting in progress. Multiple Counter Latch Commands may be used to latch more than one Counter. Each latched Counter's OL holds its count until read. Counter Latch Commands do not affect the programmed Mode of the Counter in any way.

If a Counter is latched and then, some time later, latched again before the count is read, the second Counter Latch Command is ignored. The count read will be the count at the time the first Counter Latch Command was issued.

With either method, the count must be read according to the programmed format; specifically, if the Counter is programmed for two byte counts, two bytes must be read. The two bytes do not have to be read one right after the other; read or write or programming operations of other Counters may be inserted between them.

Another feature of the 82C54 is that reads and writes of the same Counter may be interleaved; for example, if the Counter is programmed for two byte counts, the following sequence is valid.

1. Read least significant byte.
2. Write new least significant byte.
3. Read most significant byte.
4. Write new most significant byte.

If a counter is programmed to read or write two-byte counts, the following precaution applies: A program MUST NOT transfer control between reading the first and second byte to another routine which also reads from that same Counter. Otherwise, an incorrect count will be read.

Read-Back Command

The read-back command allows the user to check the count value, programmed Mode, and current state of the OUT pin and Null Count flag of the selected counter(s).

The command is written into the Control Word Register and has the format shown in Figure 5. The command applies to the counters selected by setting their corresponding bits D3, D2, D1 = 1.

A0, A1 = 11; $\overline{CS}$ = 0; $\overline{RD}$ = 1; $\overline{WR}$ = 0

D7	D6	D5	D4	D3	D2	D1	D0
1	1	$\overline{COUNT}$	$\overline{STATUS}$	CNT 2	CNT 1	CNT 0	0

D5: 0 = Latch count of selected Counter (s)
D4: 0 = Latch status of selected Counter(s)
D3: 1 = Select Counter 2
D2: 1 = Select Counter 1
D1: 1 = Select Counter 0
D0: Reserved for future expansion; Must be 0

FIGURE 5. READ-BACK COMMAND FORMAT

The read-back command may be used to latch multiple counter output latches (OL) by setting the COUNT bit D5 = 0 and selecting the desired counter(s). This signal command is functionally equivalent to several counter latch commands, one for each counter latched. Each counter's latched count is held until it is read (or the counter is reprogrammed). That counter is automatically unlatched when read, but other counters remain latched until they are read. If multiple count read-back commands are issued to the same counter without reading the count, all but the first are ignored; i.e., the count which will be read is the count at the time the first read-back command was issued.

The read-back command may also be used to latch status information of selected counter(s) by setting STATUS bit D4 = 0. Status must be latched to be read; status of a counter is accessed by a read from that counter.

The counter status format is shown in Figure 6. Bits D5 through D0 contain the counter's programmed Mode exactly as written in the last Mode Control Word. OUTPUT bit D7 contains the current state of the OUT pin. This allows the user to monitor the counter's output via software, possibly eliminating some hardware from a system.

D7	D6	D5	D4	D3	D2	D1	D0
OUTPUT	NULL COUNT	RW1	RW0	M2	M1	M0	BCD

D7: 1 = Out pin is 1
0 = Out pin is 0
D6: 1 = Null count
0 = Count available for reading
D5 - D0 = Counter programmed mode (See Control Word Formats)

FIGURE 6. STATUS BYTE

NULL COUNT bit D6 indicates when the last count written to the counter register (CR) has been loaded into the counting element (CE). The exact time this happens depends on the Mode of the counter and is described in the Mode Definitions, but until the counter is loaded into the counting element (CE), it can't be read from the counter. If the count is latched or read before this time, the count value will not reflect the new count just written. The operation of Null Count is shown below.

THIS ACTION: CAUSES:

A. Write to the control word register:(1) . . . . . Null Count = 1
B. Write to the count register (CR):(2) . . . . . Null Count = 1
C. New count is loaded into CE (CR - CE) . . . . . Null Count = 0

(1) Only the counter specified by the control word will have its null count set to 1. Null count bits of other counters are unaffected.

(2) If the counter is programmed for two-byte counts (least significant byte then most significant byte) null count goes to 1 when the second byte is written.

If multiple status latch operations of the counter(s) are performed without reading the status, all but the first are ignored; i.e., the status that will be read is the status of the counter at the time the first status read-back command was issued.

COMMANDS								DESCRIPTION	RESULT
D7	D6	D5	D4	D3	D2	D1	D0		
1	1	0	0	0	0	1	0	Read-Back Count and Status of Counter 0	Count and Status Latched for Counter 0
1	1	1	0	0	1	0	0	Read-Back Status of Counter 1	Status Latched for Counter 1
1	1	1	0	1	1	0	0	Read-Back Status of Counters 2, 1	Status Latched for Counter 2, But Not Counter 1
1	1	0	1	1	0	0	0	Read-Back Count of Counter 2	Count Latched for Counter 2
1	1	0	0	0	1	0	0	Read-Back Count and Status of Counter 1	Count Latched for Counter 1, But Not Status
1	1	1	0	0	0	1	0	Read-Back Status of Counter 1	Command Ignored, Status Already Latched for Counter 1

FIGURE 7. READ-BACK COMMAND EXAMPLE

Both count and status of the selected counter(s) may be latched simultaneously by setting both COUNT and STATUS bits D5, D4 = 0. This is functionally the same as issuing two separate read-back commands at once, and the above discussions apply here also. Specifically, if multiple count and/or status read-back commands are issued to the same counter(s) without any intervening reads, all but the first are ignored. This is illustrated in Figure 7.

If both count and status of a counter are latched, the first read operation of that counter will return latched status, regardless of which was latched first. The next one or two reads (depending on whether the counter is programmed for one or two type counts) return latched count. Subsequent reads return unlatched count.

$\overline{CS}$	$\overline{RD}$	$\overline{WR}$	A1	A0	
0	1	0	0	0	Write into Counter 0
0	1	0	0	1	Write into Counter 1
0	1	0	1	0	Write into Counter 2
0	1	0	1	1	Write Control Word
0	0	1	0	0	Read from Counter 0
0	0	1	0	1	Read from Counter 1
0	0	1	1	0	Read from Counter 2
0	0	1	1	1	No-Operation (Three-State)
1	X	X	X	X	No-Operation (Three-State)
0	1	1	X	X	No-Operation (Three-State)

FIGURE 8. READ/WRITE OPERATIONS SUMMARY

Mode Definitions

The following are defined for use in describing the operation of the 82C54.

CLK PULSE:

A rising edge, then a falling edge, in that order, of a Counter's CLK input.

TRIGGER:

A rising edge of a Counter's Gate input.

COUNTER LOADING:

The transfer of a count from the CR to the CE (See "Functional Description")

Mode 0: Interrupt on Terminal Count

Mode 0 is typically used for event counting. After the Control Word is written, OUT is initially low, and will remain low until the Counter reaches zero. OUT then goes high and remains high until a new count or a new Mode 0 Control Word is written to the Counter.

GATE = 1 enables counting; GATE = 0 disables counting. GATE has no effect on OUT.

After the Control Word and initial count are written to a Counter, the initial count will be loaded on the next CLK pulse. This CLK pulse does not decrement the count, so for an initial count of N, OUT does not go high until N + 1 CLK pulses after the initial count is written.

If a new count is written to the Counter it will be loaded on the next CLK pulse and counting will continue from the new count. If a two-byte count is written, the following happens:

(1) Writing the first byte disables counting. Out is set low immediately (no clock pulse required).

(2) Writing the second byte allows the new count to be loaded on the next CLK pulse.

This allows the counting sequence to be synchronized by software. Again OUT does not go high until N + 1 CLK pulses after the new count of N is written.

If an initial count is written while GATE = 0, it will still be loaded on the next CLK pulse. When GATE goes high, OUT will go high N CLK pulses later; no CLK pulse is needed to load the counter as this has already been done.

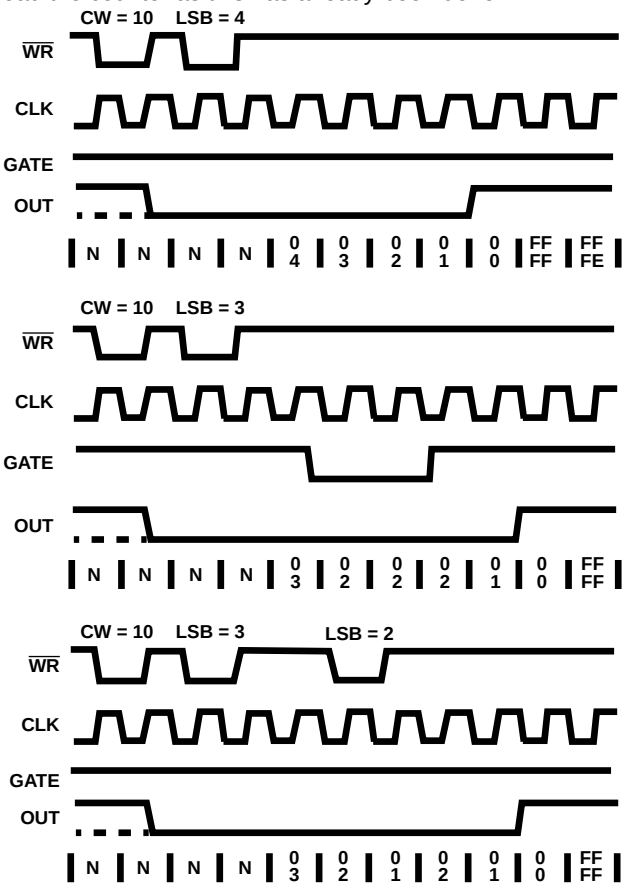


FIGURE 9. MODE 0

NOTES: The following conventions apply to all mode timing diagrams.

1. Counters are programmed for binary (not BCD) counting and for reading/writing least significant byte (LSB) only.
2. The counter is always selected ($\overline{CS}$ always low).
3. CW stands for "Control Word"; CW = 10 means a control word of 10, Hex is written to the counter.
4. LSB stands for Least significant "byte" of count.
5. Numbers below diagrams are count values. The lower number is the least significant byte. The upper number is the most significant byte. Since the counter is programmed to read/write LSB only, the most significant byte cannot be read.
6. N stands for an undefined count.
7. Vertical lines show transitions between count values.

Mode 1: Hardware Retriggerable One-Shot

OUT will be initially high. OUT will go low on the CLK pulse following a trigger to begin the one-shot pulse, and will remain low until the Counter reaches zero. OUT will then go high and remain high until the CLK pulse after the next trigger.

After writing the Control Word and initial count, the Counter is armed. A trigger results in loading the Counter and setting OUT low on the next CLK pulse, thus starting the one-shot pulse N CLK cycles in duration. The one-shot is retriggerable, hence OUT will remain low for N CLK pulses after any trigger. The one-shot pulse can be repeated without rewriting the same count into the counter. GATE has no effect on OUT.

If a new count is written to the Counter during a one-shot pulse, the current one-shot is not affected unless the Counter is retriggerable. In that case, the Counter is loaded with the new count and the one-shot pulse continues until the new count expires.

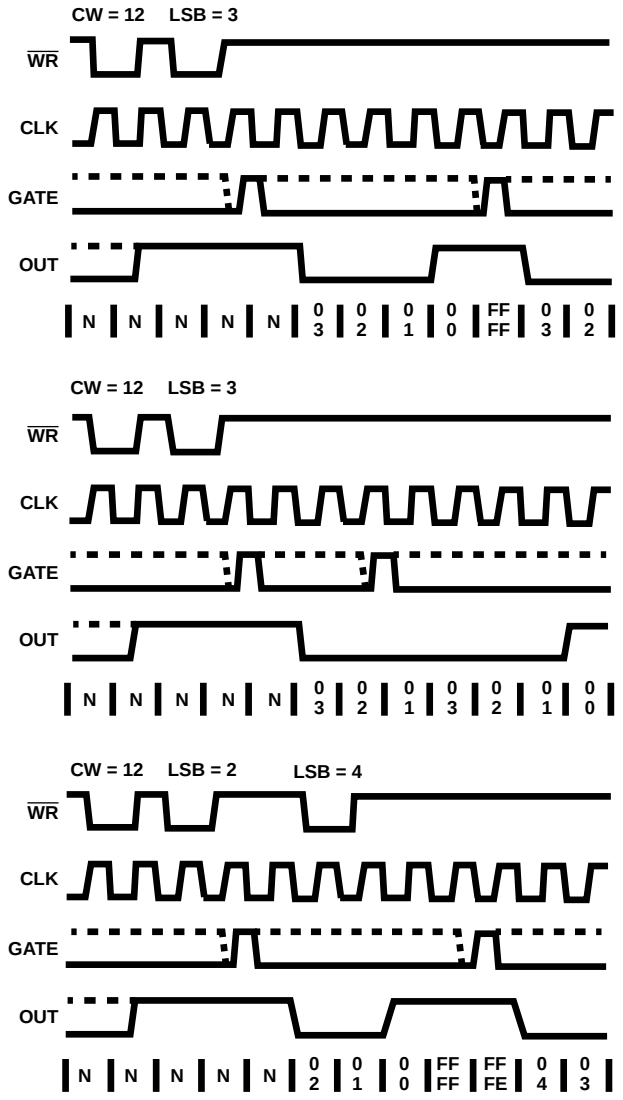


FIGURE 10. MODE 1

Mode 2: Rate Generator

This Mode functions like a divide-by-N counter. It is typically used to generate a Real Time Clock Interrupt. OUT will initially be high. When the initial count has decremented to 1, OUT goes low for one CLK pulse. OUT then goes high again, the Counter reloads the initial count and the process is repeated. Mode 2 is periodic; the same sequence is repeated indefinitely. For an initial count of N, the sequence repeats every N CLK cycles.

GATE = 1 enables counting; GATE = 0 disables counting. If GATE goes low during an output pulse, OUT is set high immediately. A trigger reloads the Counter with the initial count on the next CLK pulse; OUT goes low N CLK pulses after the trigger. Thus the GATE input can be used to synchronize the Counter.

After writing a Control Word and initial count, the Counter will be loaded on the next CLK pulse. OUT goes low N CLK pulses after the initial count is written. This allows the Counter to be synchronized by software also.

Writing a new count while counting does not affect the current counting sequence. If a trigger is received after writing a new count but before the end of the current period, the Counter will be loaded with the new count on the next CLK pulse and counting will continue from the end of the current counting cycle.

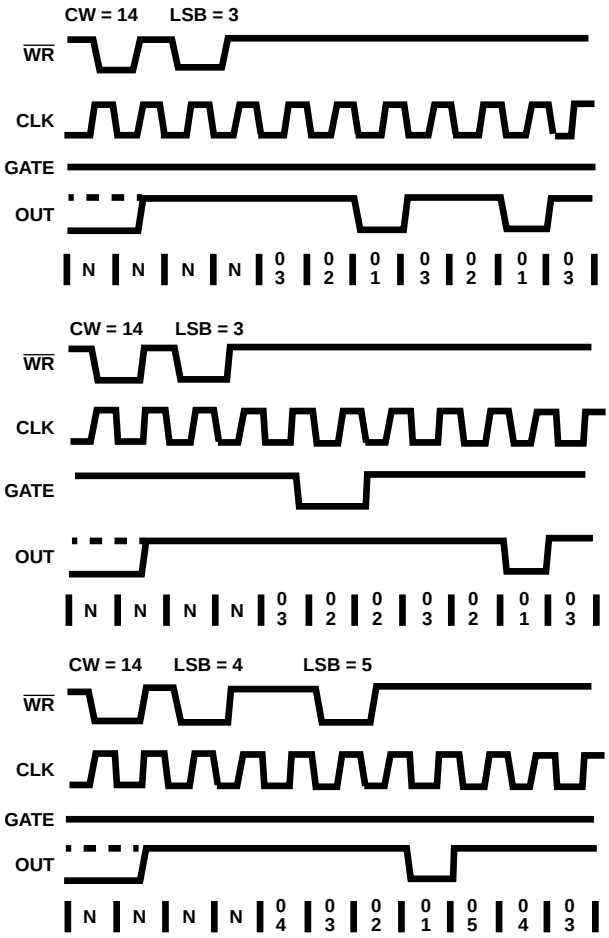


FIGURE 11. MODE 2

Mode 3: Square Wave Mode

Mode 3 is typically used for Baud rate generation. Mode 3 is similar to Mode 2 except for the duty cycle of OUT. OUT will initially be high. When half the initial count has expired, OUT goes low for the remainder of the count. Mode 3 is periodic; the sequence above is repeated indefinitely. An initial count of N results in a square wave with a period of N CLK cycles.

GATE = 1 enables counting; GATE = 0 disables counting. If GATE goes low while OUT is low, OUT is set high immediately; no CLK pulse is required. A trigger reloads the Counter with the initial count on the next CLK pulse. Thus the GATE input can be used to synchronize the Counter.

After writing a Control Word and initial count, the Counter will be loaded on the next CLK pulse. This allows the Counter to be synchronized by software also.

Writing a new count while counting does not affect the current counting sequence. If a trigger is received after writing a new count but before the end of the current half-cycle of the square wave, the Counter will be loaded with the new count on the next CLK pulse and counting will continue from the new count. Otherwise, the new count will be loaded at the end of the current half-cycle.

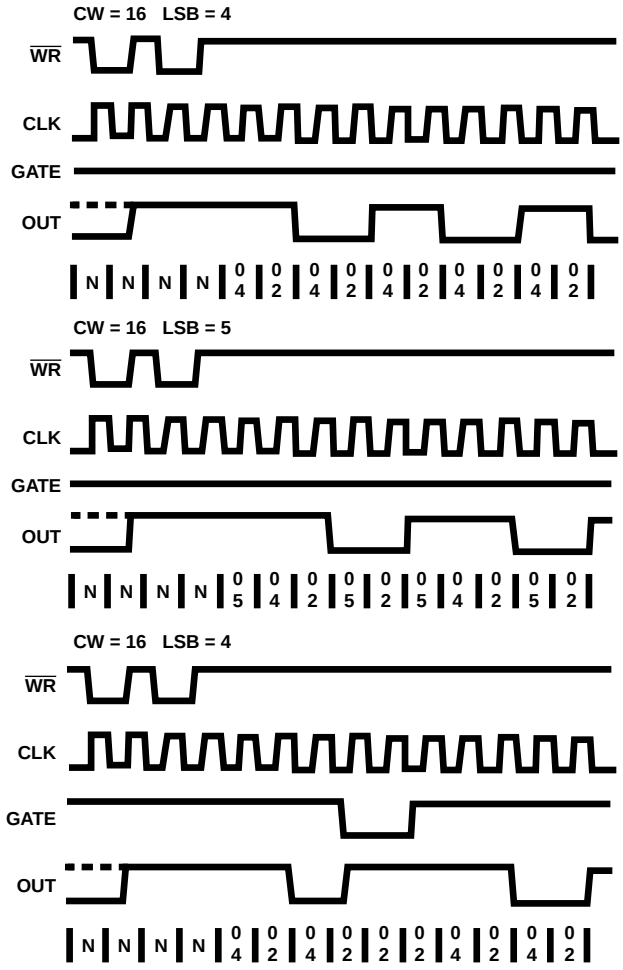


FIGURE 12. MODE 3

Mode 3 is Implemented as Follows:

EVEN COUNTS: OUT is initially high. The initial count is loaded on one CLK pulse and then is decremented by two on succeeding CLK pulses. When the count expires, OUT changes value and the Counter is reloaded with the initial count. The above process is repeated indefinitely.

ODD COUNTS: OUT is initially high. The initial count is loaded on one CLK pulse, decremented by one on the next CLK pulse, and then decremented by two on succeeding CLK pulses. When the count expires, OUT goes low and the Counter is reloaded with the initial count. The count is decremented by three on the next CLK pulse, and then by two on succeeding CLK pulses. When the count expires, OUT goes high again and the Counter is reloaded with the initial count. The above process is repeated indefinitely. So for odd counts, OUT will be high for (N + 1)/2 counts and low for (N - 1)/2 counts.

Mode 4: Software Triggered Mode

OUT will be initially high. When the initial count expires, OUT will go low for one CLK pulse then go high again. The counting sequence is “Triggered” by writing the initial count.

GATE = 1 enables counting; GATE = 0 disables counting. GATE has no effect on OUT.

After writing a Control Word and initial count, the Counter will be loaded on the next CLK pulse. This CLK pulse does not decrement the count, so for an initial count of N, OUT does not strobe low until N + 1 CLK pulses after the initial count is written.

If a new count is written during counting, it will be loaded on the next CLK pulse and counting will continue from the new count. If a two-byte count is written, the following happens:

(1) Writing the first byte has no effect on counting.

(2) Writing the second byte allows the new count to be loaded on the next CLK pulse.

This allows the sequence to be “retriggered” by software. OUT strobes low N + 1 CLK pulses after the new count of N is written.

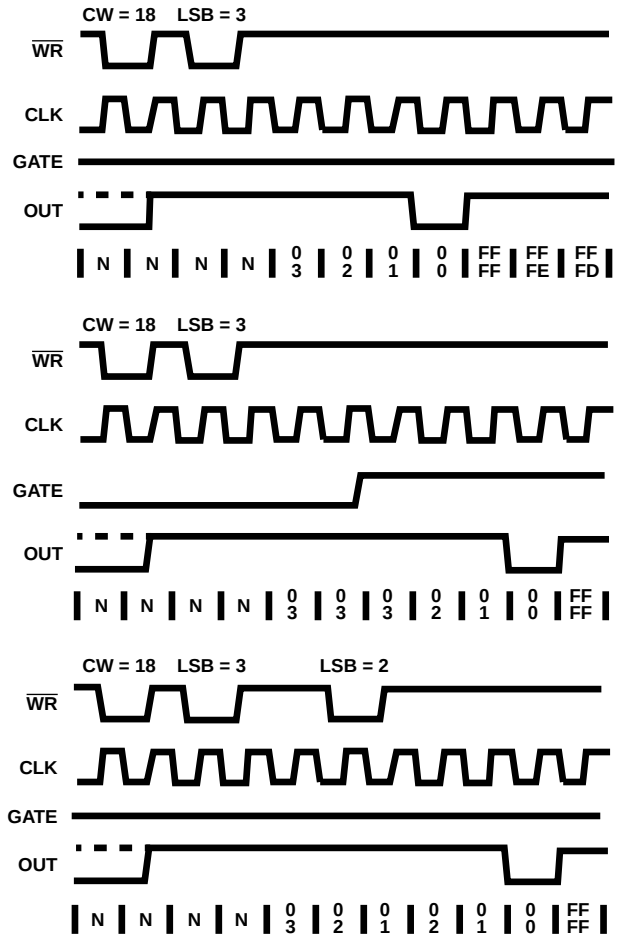


FIGURE 13. MODE 4

Mode 5: Hardware Triggered Strobe (Retriggerable)

OUT will initially be high. Counting is triggered by a rising edge of GATE. When the initial count has expired, OUT will go low for one CLK pulse and then go high again.

After writing the Control Word and initial count, the counter will not be loaded until the CLK pulse after a trigger. This CLK pulse does not decrement the count, so for an initial count of N, OUT does not strobe low until N + 1 CLK pulses after trigger.

A trigger results in the Counter being loaded with the initial count on the next CLK pulse. The counting sequence is triggerable. OUT will not strobe low for N + 1 CLK pulses after any trigger GATE has no effect on OUT.

If a new count is written during counting, the current counting sequence will not be affected. If a trigger occurs after the new count is written but before the current count expires, the Counter will be loaded with new count on the next CLK pulse and counting will continue from there.

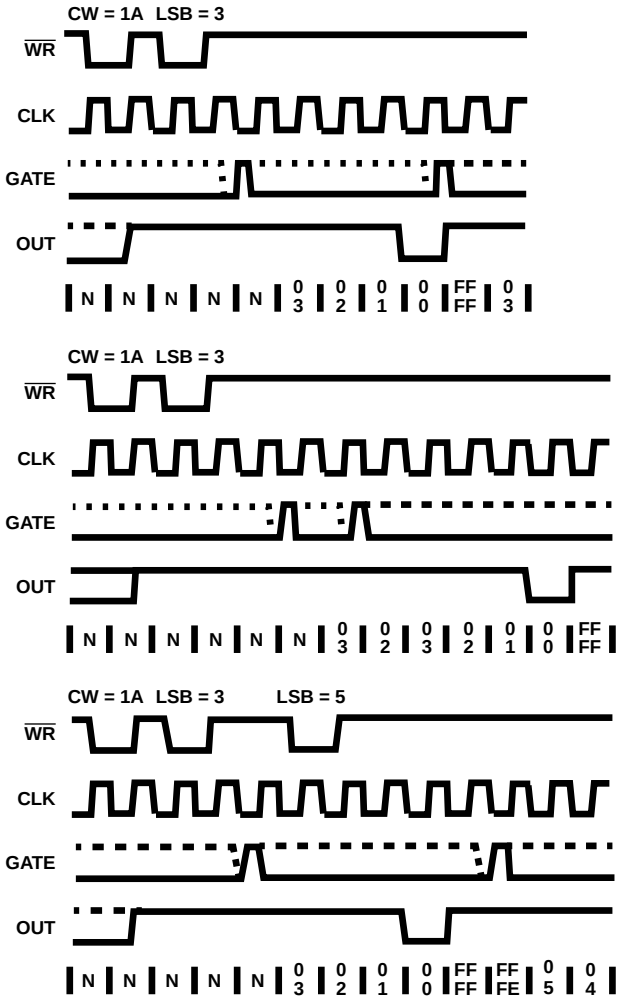


FIGURE 14. MODE 5

Operation Common to All Modes

Programming

When a Control Word is written to a Counter, all Control Logic, is immediately reset and OUT goes to a known initial state; no CLK pulses are required for this.

Gate

The GATE input is always sampled on the rising edge of CLK. In Modes 0, 2, 3 and 4 the GATE input is level sensitive, and logic level is sampled on the rising edge of CLK. In modes 1, 2, 3 and 5 the GATE input is rising-edge sensitive. In these Modes, a rising edge of Gate (trigger) sets an edge-sensitive flip-flop in the Counter. This flip-flop is then sampled on the next rising edge of CLK. The flip-flop is reset immediately after it is sampled. In this way, a trigger will be detected no matter when it occurs - a high logic level does not have to be maintained until the next rising edge of CLK. Note that in Modes 2 and 3, the GATE input is both edge- and level-sensitive.

Counter

New counts are loaded and Counters are decremented on the falling edge of CLK.

The largest possible initial count is 0; this is equivalent to 2^{16} for binary counting and 10^4 for BCD counting.

The counter does not stop when it reaches zero. In Modes 0, 1, 4, and 5 the Counter “wraps around” to the highest count, either FFFF hex for binary counting or 9999 for BCD counting, and continues counting. Modes 2 and 3 are periodic; the Counter reloads itself with the initial count and continues counting from there.

SIGNAL STATUS MODES	LOW OR GOING LOW	RISING	HIGH
0	Disables Counting	-	Enables Counting
1	-	1) Initiates Counting 2) Resets output after next clock	-
2	1) Disables counting 2) Sets output immediately high	Initiates Counting	Enables Counting
3	1) Disables counting 2) Sets output immediately high	Initiates Counting	Enables Counting
4	1) Disables Counting	-	Enables Counting
5	-	Initiates Counting	-

FIGURE 15. GATE PIN OPERATIONS SUMMARY

MODE	MIN COUNT	MAX COUNT
0	1	0
1	1	0
2	2	0
3	2	0
4	1	0
5	1	0

NOTE: 0 is equivalent to 2^{16} for binary counting and 10^4 for BCD counting.

FIGURE 16. MINIMUM AND MAXIMUM INITIAL COUNTS

82C55A

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^oC to 70^oC	E40.6
IP82C55A-5	IP82C55A		-40^oC to 85^oC	E40.6
CS82C55A-5	CS82C55A	44 Ld PLCC	0^oC to 70^oC	N44.65
IS82C55A-5	IS82C55A		-40^oC to 85^oC	N44.65
CD82C55A-5	CD82C55A	40 Ld CERDIP	0^oC to 70^oC	F40.6
ID82C55A-5	ID82C55A		-40^oC to 85^oC	F40.6
MD82C55A-5/B	MD82C55A/B		-55^oC to 125^oC	F40.6
8406601QA	8406602QA	SMD#		F40.6
MR82C55A-5/B	MR82C55A/B	44 Pad CLCC	-55^oC to 125^oC	J44.A
8406601XA	8406602XA	SMD#		J44.A

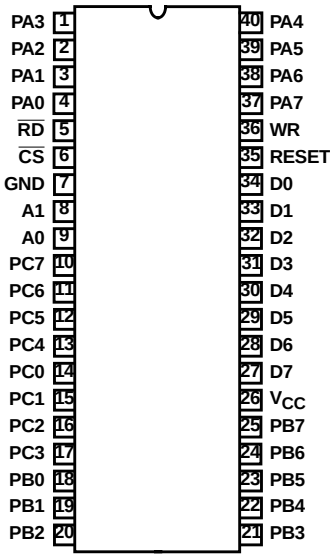
Description

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 SAJI 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.

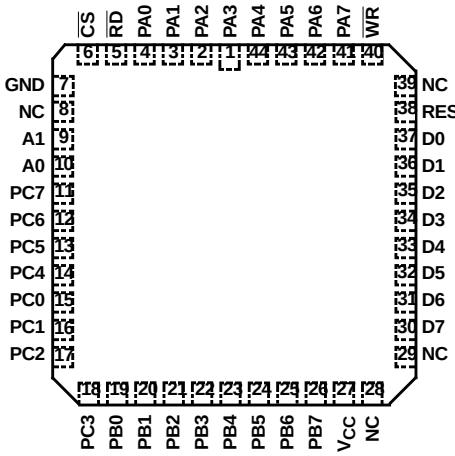
Static CMOS circuit design insures low operating power. TTL compatibility over the full military temperature range and bus hold circuitry eliminate the need for pull-up resistors. The Intersil advanced SAJI process results in performance equal to or greater than existing functionally equivalent products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

Pinouts

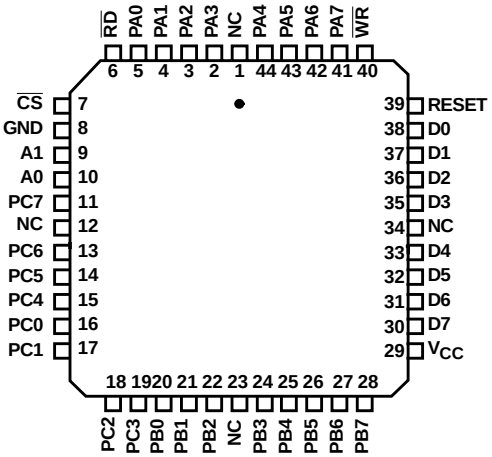
82C55A (DIP)
TOP VIEW



82C55A (CLCC)
TOP VIEW



82C55A (PLCC)
TOP VIEW

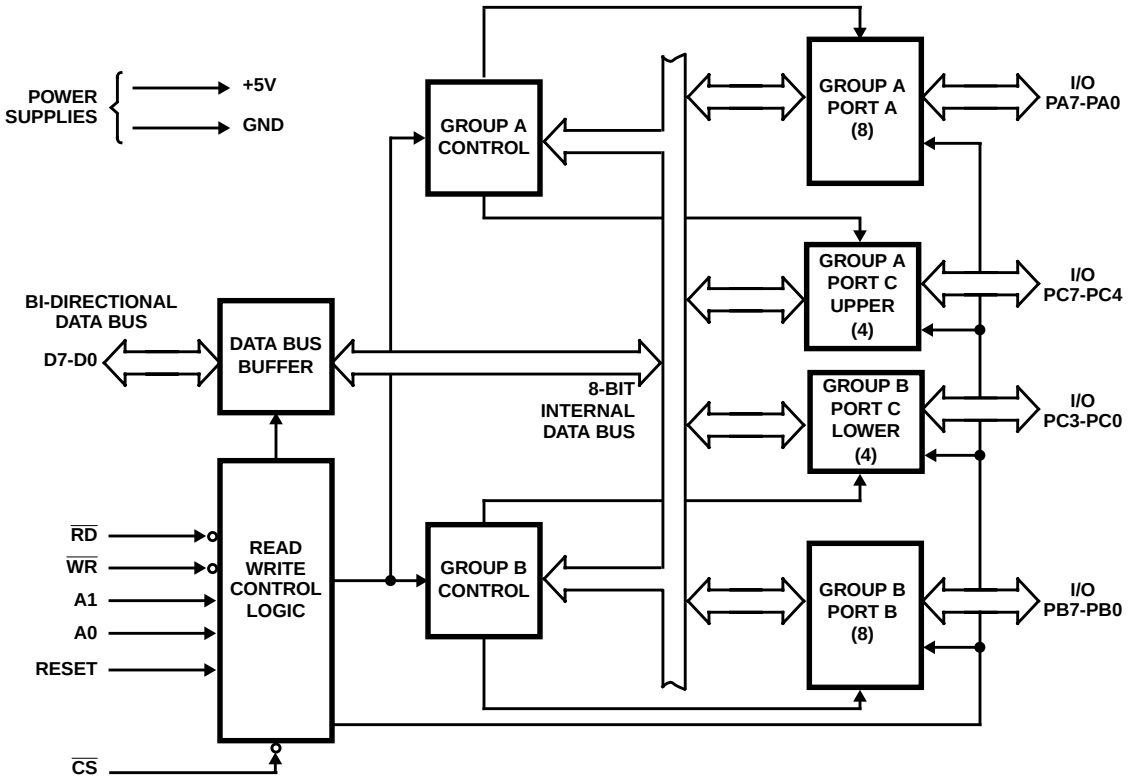


File Number **2969.2**

Pin Description

SYMBOL	PIN NUMBER	TYPE	DESCRIPTION
V_{CC}	26		V_{CC}: The +5V power supply pin. A 0.1µF 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.
$\overline{CS}$	6	I	CHIP SELECT: Chip select is an active low input used to enable the 82C55A onto the Data Bus for CPU communications.
$\overline{RD}$	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.
$\overline{WR}$	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 $\overline{RD}$ and $\overline{WR}$ 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 communcation 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).

82C55A BASIC OPERATION

A1	A0	RD	WR	CS	INPUT OPERATION (READ)
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

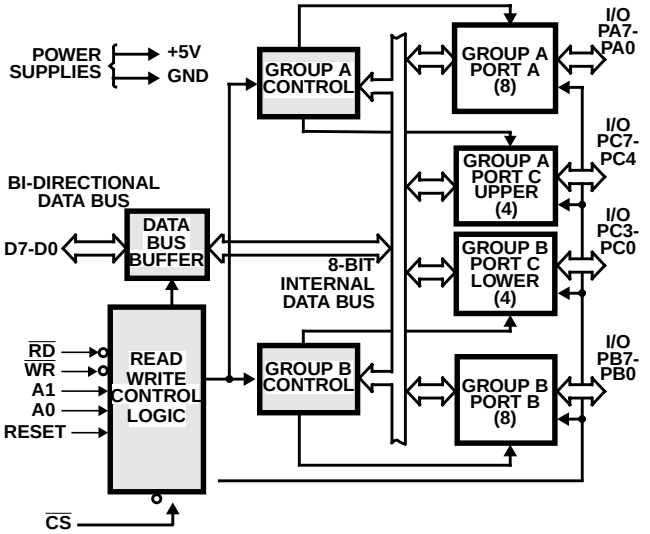


FIGURE 1. 82C55A BLOCK DIAGRAM. DATA BUS BUFFER, READ/WRITE, GROUP A & B CONTROL LOGIC FUNCTIONS

(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 400µA.

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.

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.

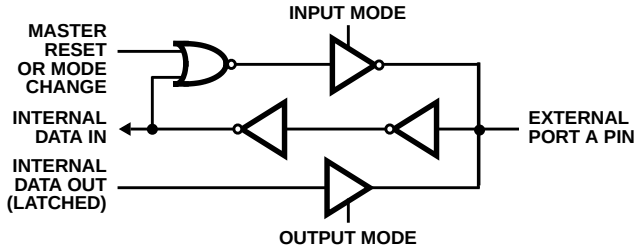


FIGURE 2A. PORT A BUS-HOLD CONFIGURATION

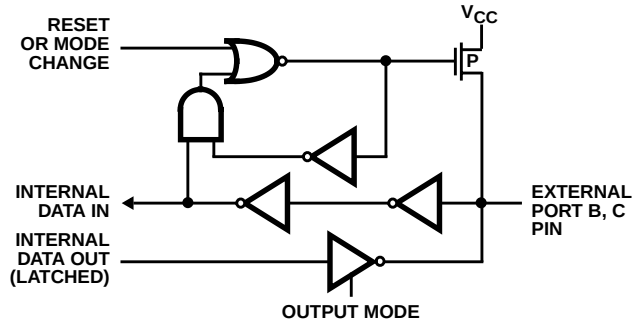


FIGURE 2B. PORT B AND C BUS-HOLD CONFIGURATION

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 pull-down 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.

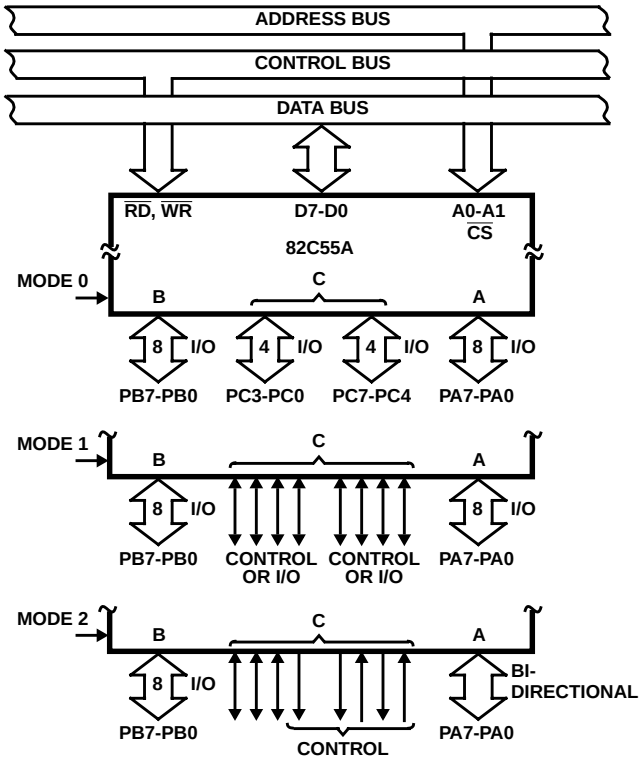


FIGURE 3. BASIC MODE DEFINITIONS AND BUS INTERFACE

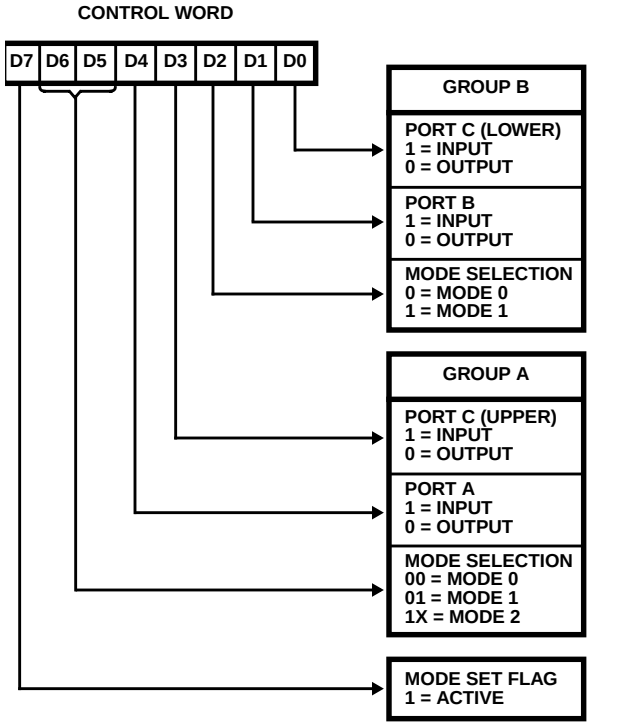


FIGURE 4. MODE DEFINITION FORMAT

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.

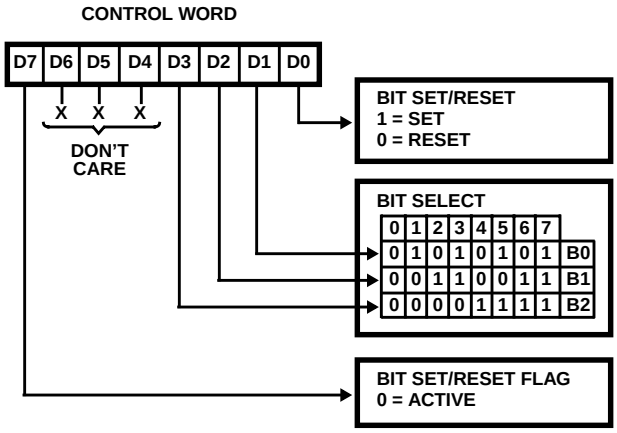


FIGURE 5. BIT SET/RESET FORMAT

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.

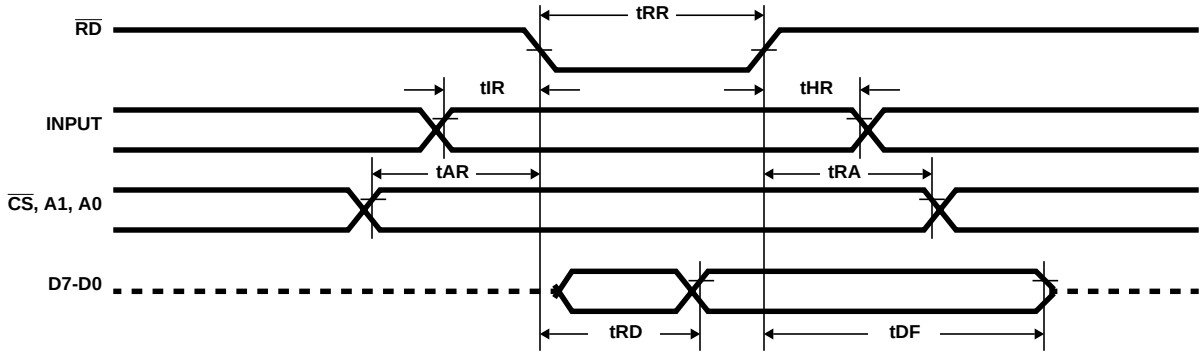
Mode 0 Basic Functional Definitions:

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

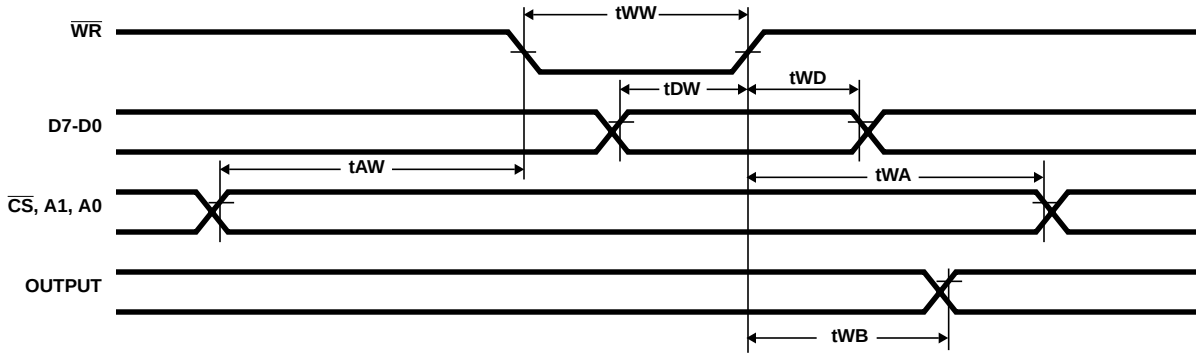
MODE 0 PORT DEFINITION

A		B		GROUP A			GROUP B	
D4	D3	D1	D0	PORT A	PORT C (Upper)	#	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

Mode 0 (Basic Input)



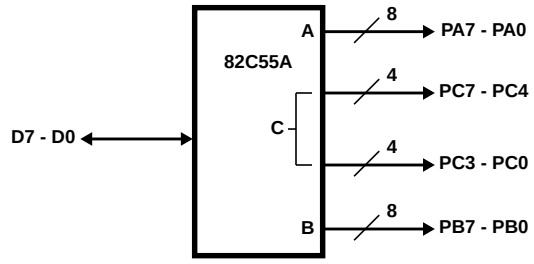
Mode 0 (Basic Output)



Mode 0 Configurations

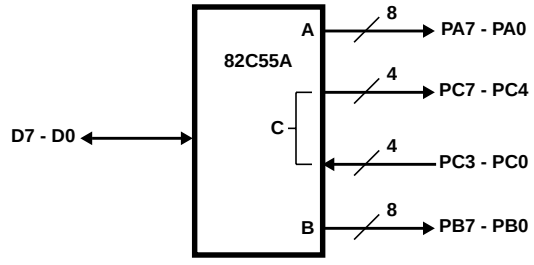
CONTROL WORD #0

D7	D6	D5	D4	D3	D2	D1	D0
1	0	0	0	0	0	0	0



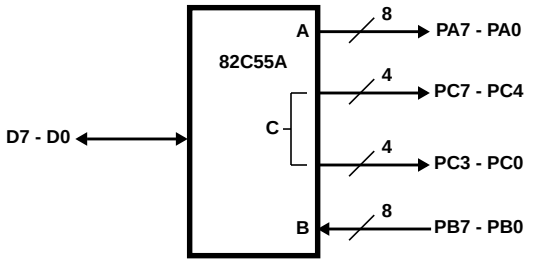
CONTROL WORD #1

D7	D6	D5	D4	D3	D2	D1	D0
1	0	0	0	0	0	0	1



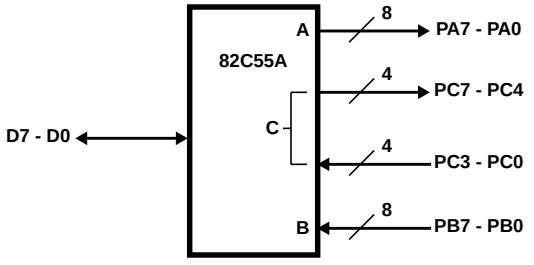
CONTROL WORD #2

D7	D6	D5	D4	D3	D2	D1	D0
1	0	0	0	0	0	1	0



CONTROL WORD #3

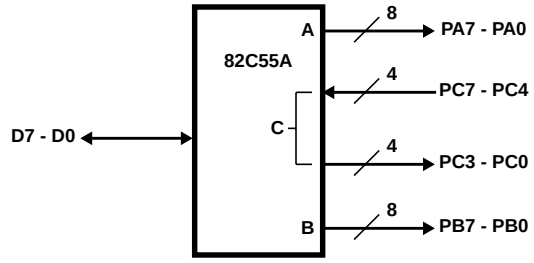
D7	D6	D5	D4	D3	D2	D1	D0
1	0	0	0	0	0	1	1



Mode 0 Configurations (Continued)

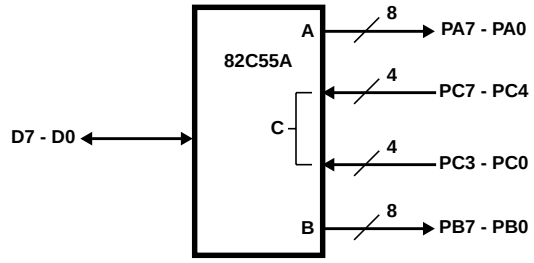
CONTROL WORD #4

D7	D6	D5	D4	D3	D2	D1	D0
1	0	0	0	1	0	0	0



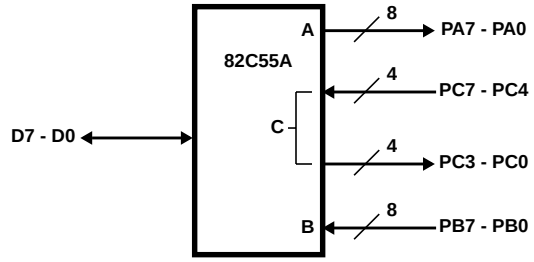
CONTROL WORD #5

D7	D6	D5	D4	D3	D2	D1	D0
1	0	0	0	1	0	0	1



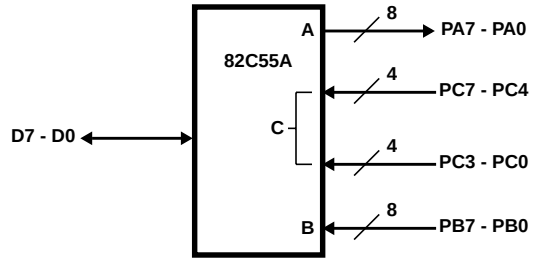
CONTROL WORD #6

D7	D6	D5	D4	D3	D2	D1	D0
1	0	0	0	1	0	1	0



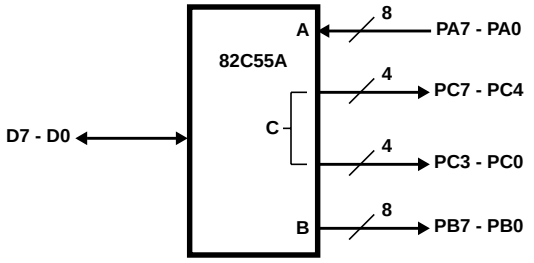
CONTROL WORD #7

D7	D6	D5	D4	D3	D2	D1	D0
1	0	0	0	1	0	1	1



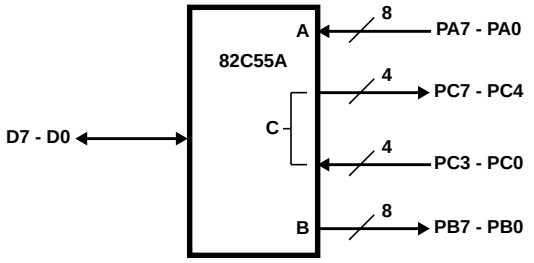
CONTROL WORD #8

D7	D6	D5	D4	D3	D2	D1	D0
1	0	0	1	0	0	0	0



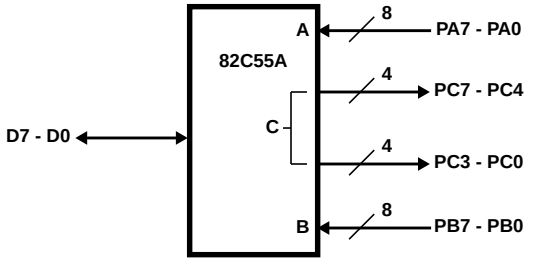
CONTROL WORD #9

D7	D6	D5	D4	D3	D2	D1	D0
1	0	0	1	0	0	0	1



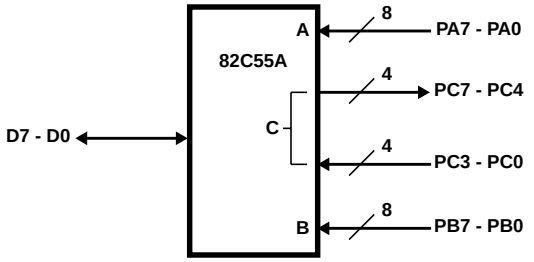
CONTROL WORD #10

D7	D6	D5	D4	D3	D2	D1	D0
1	0	0	1	0	0	1	0



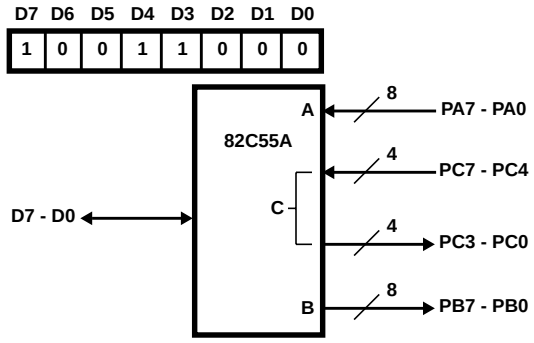
CONTROL WORD #11

D7	D6	D5	D4	D3	D2	D1	D0
1	0	0	1	0	0	1	1

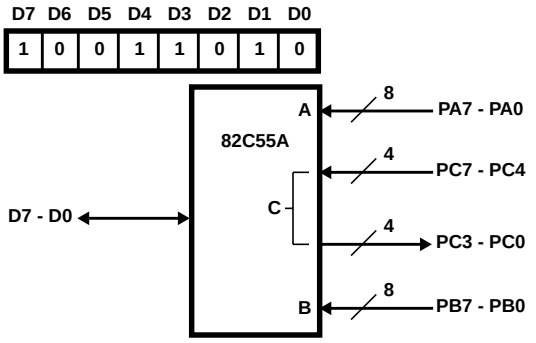


Mode 0 Configurations (Continued)

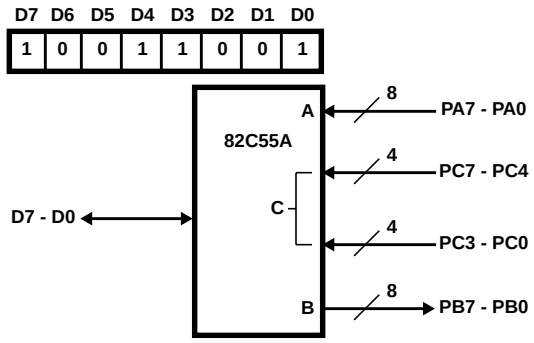
CONTROL WORD #12



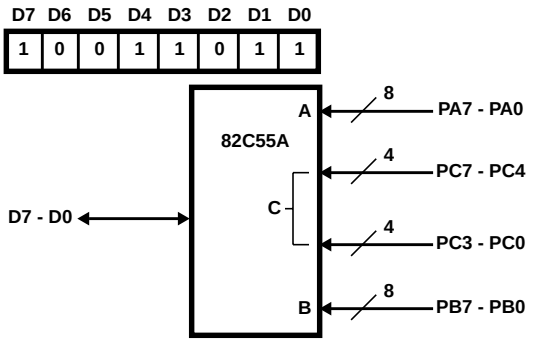
CONTROL WORD #14



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, and acknowledgment. IBF is set by $\overline{STB}$ input being low and is reset by the rising edge of the $\overline{RD}$ input.

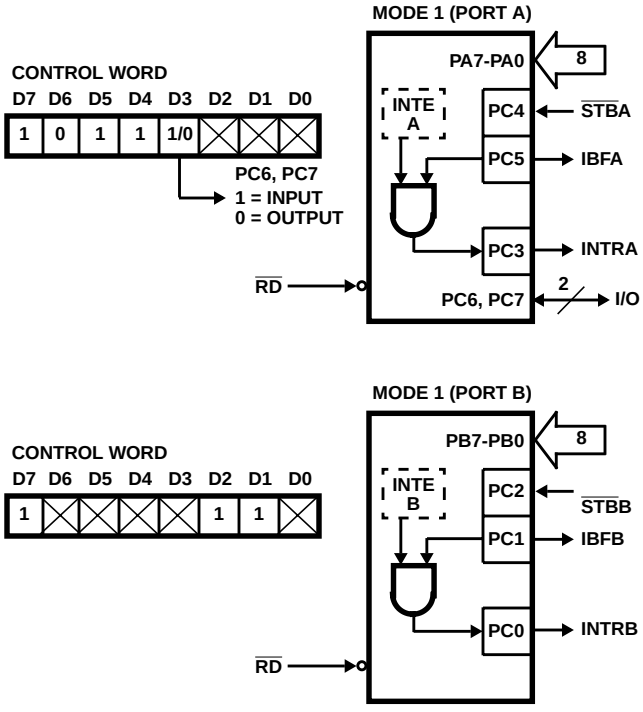


FIGURE 6. MODE 1 INPUT

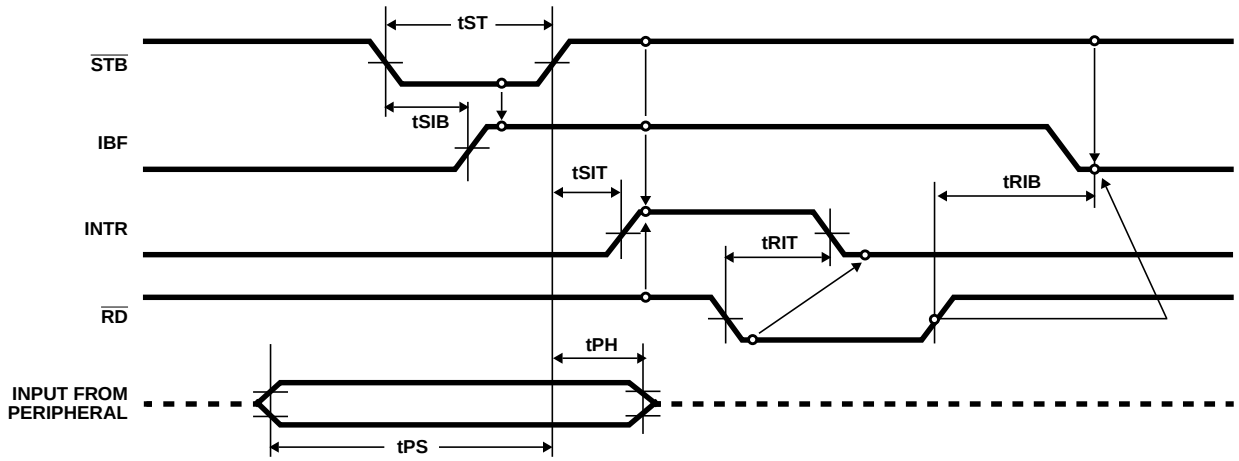


FIGURE 7. MODE 1 (STROBED INPUT)

INTR (Interrupt Request)

A “high” on this output can be used to interrupt the CPU when and input device is requesting service. INTR is set by the condition: $\overline{STB}$ is a “one”, IBF is a “one” and INTE is a “one”. It is reset by the falling edge of $\overline{RD}$. 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)

$\overline{OBF}$ - 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 $\overline{WR}$ input and reset by $\overline{ACK}$ input being low.

$\overline{ACK}$ - 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 $\overline{ACK}$ is a “one”, OBF is a “one” and INTE is a “one”. It is reset by the falling edge of $\overline{WR}$.

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 $\overline{ACK}$ from the peripheral device and then latch data into the peripheral device on the rising edge of $\overline{OBF}$.

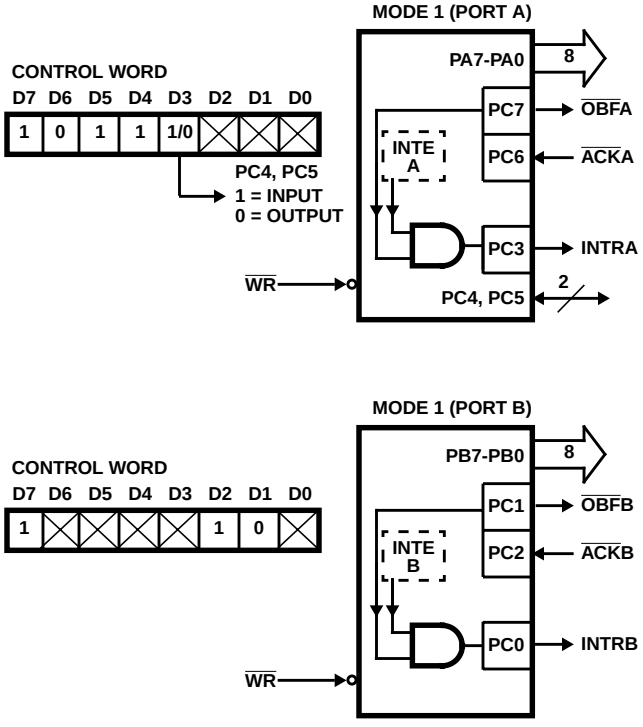


FIGURE 8. MODE 1 OUTPUT

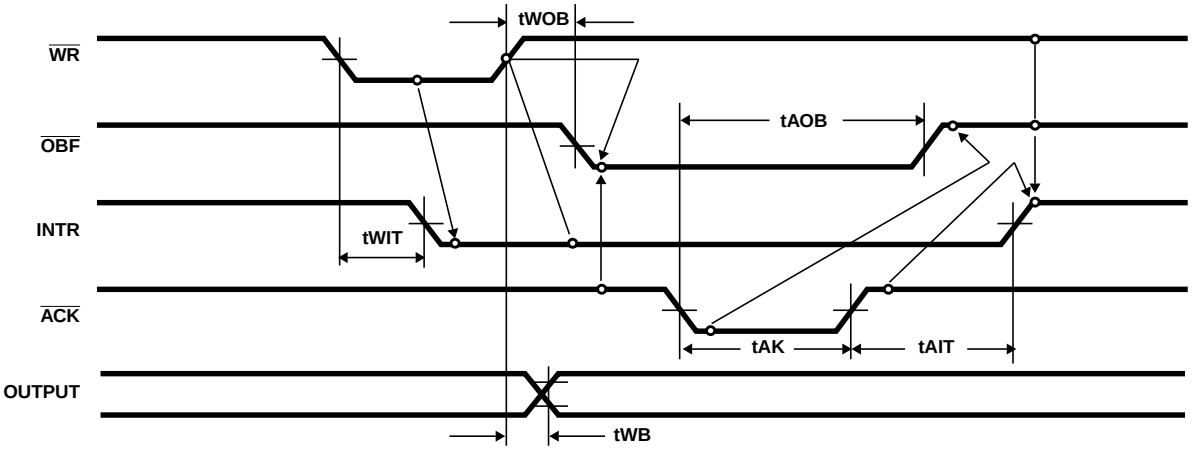
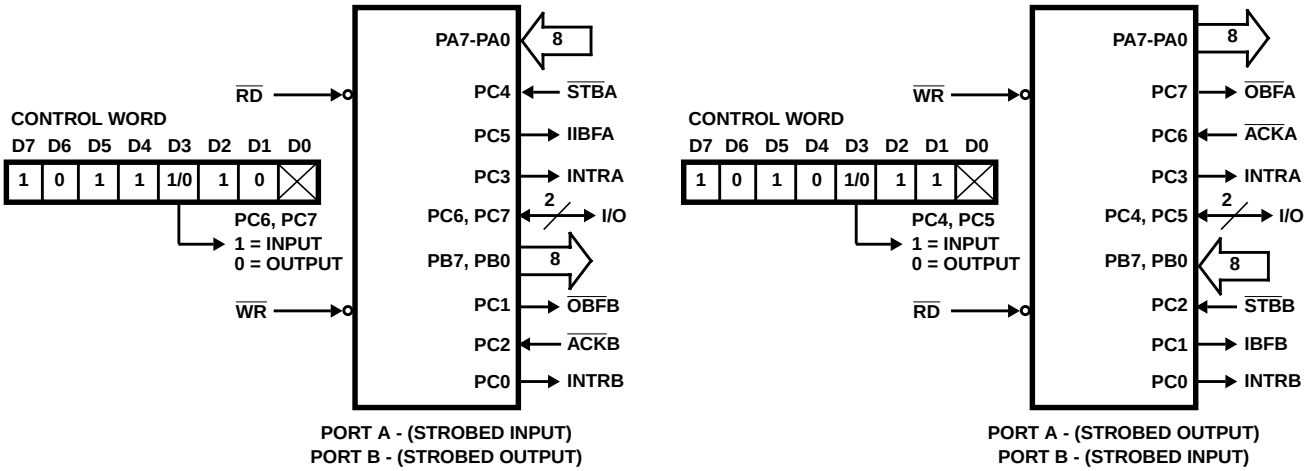


FIGURE 9. MODE 1 (STROBED OUTPUT)



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

$\overline{\text{OBF}}$ - (Output Buffer Full). The $\overline{\text{OBF}}$ output will go “low” to indicate that the CPU has written data out to port A.

$\overline{\text{ACK}}$ - (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 $\overline{\text{OBF}}$). Controlled by bit set/reset of PC4.

Input Operations

$\overline{\text{STB}}$ - (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.

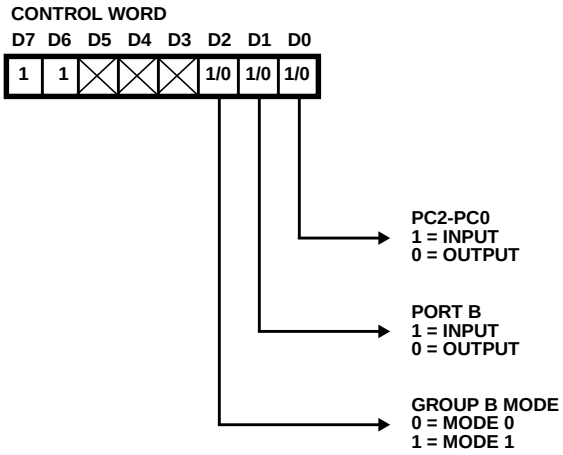


FIGURE 11. MODE CONTROL WORD

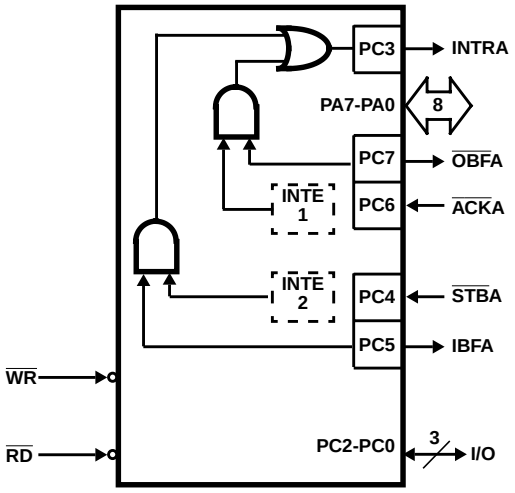
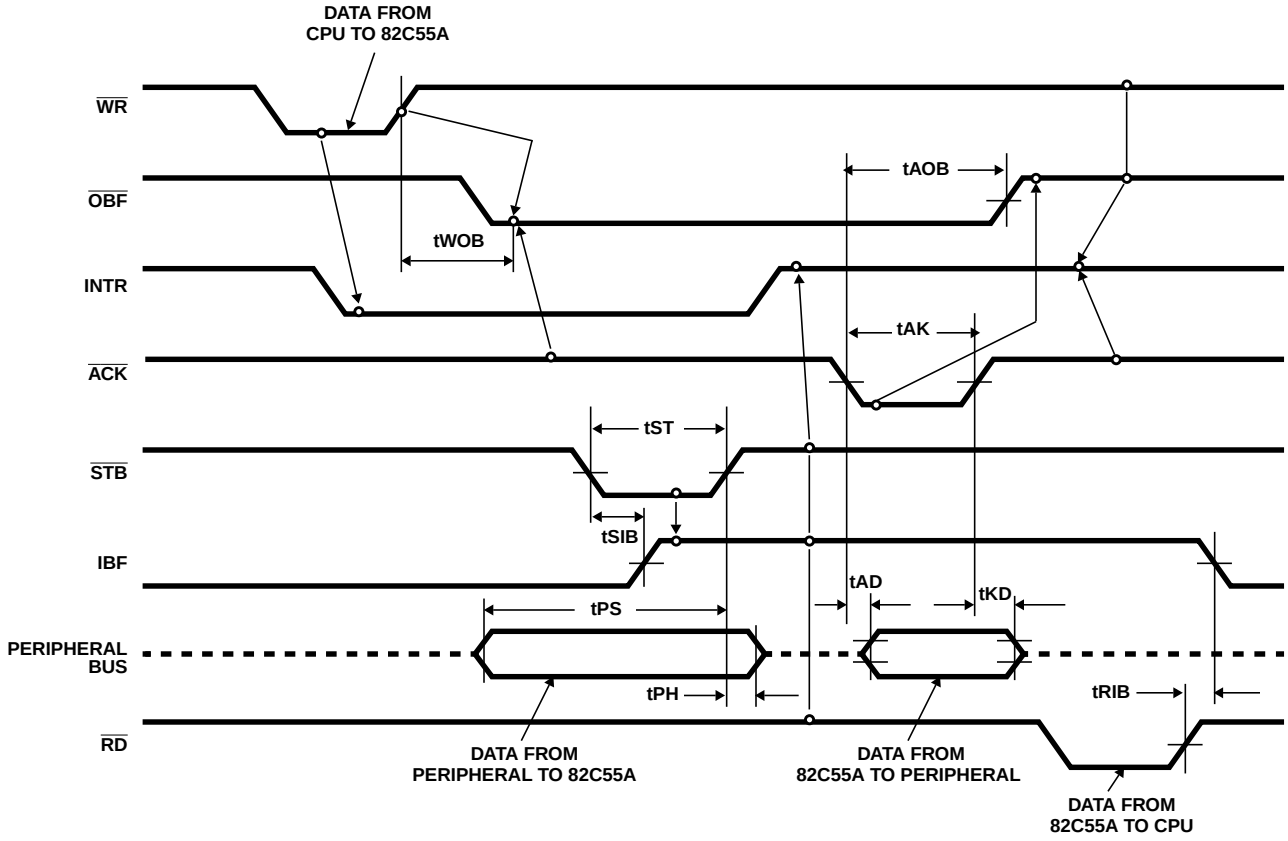


FIGURE 12. MODE 2



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

FIGURE 13. MODE 2 (BI-DIRECTIONAL)

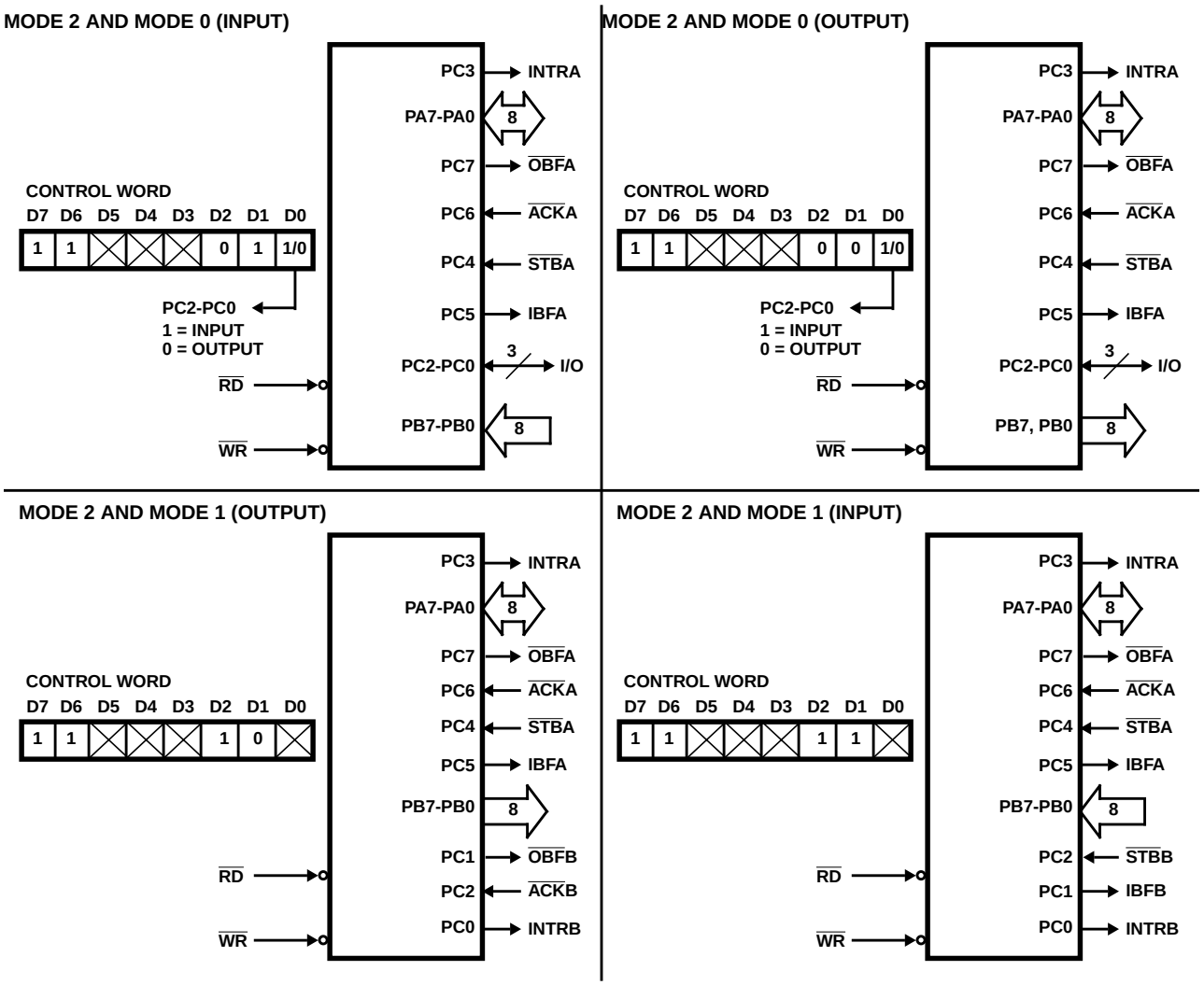


FIGURE 14. MODE 2 COMBINATIONS

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		Mode 0 or Mode 1 Only
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	$\overline{OBF}$B	I/O	
PC2	In	Out	$\overline{STB}$B	$\overline{ACK}$B	I/O	
PC3	In	Out	INTRA	INTRA	INTRA	
PC4	In	Out	$\overline{STB}$A	I/O	$\overline{STB}$A	
PC5	In	Out	IBFA	I/O	IBFA	
PC6	In	Out	I/O	$\overline{ACK}$A	$\overline{ACK}$A	
PC7	In	Out	I/O	$\overline{OBF}$A	$\overline{OBF}$A	

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 $\overline{ACK}$ and $\overline{STB}$ lines, will be placed on the data bus. In place of the $\overline{ACK}$ and $\overline{STB}$ 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 $\overline{OBF}$) can be written, or an interrupt enable flag can be either set or reset. Port C lines programmed as inputs, including $\overline{ACK}$ and $\overline{STB}$ 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 $\overline{ACK}$ and $\overline{STB}$ 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

D7	D6	D5	D4	D3	D2	D1	D0
I/O	I/O	IBFA	INTEA	INTRA	INTEB	IBFB	INTRB
GROUP A					GROUP B		

OUTPUT CONFIGURATION

D7	D6	D5	D4	D3	D2	D1	D0
OBFA	INTEA	I/O	I/O	INTRA	INTEB	OBFB	INTRB
GROUP A					GROUP B		

FIGURE 15. MODE 1 STATUS WORD FORMAT

D7	D6	D5	D4	D3	D2	D1	D0
OBFA	INTE1	IBFA	INTE2	INTRA	X	X	X
GROUP A					GROUP B		

(Defined by Mode 0 or Mode 1 Selection)

FIGURE 16. MODE 2 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	$\overline{ACK}$B (Output Mode 1) or $\overline{STB}$B (Input Mode 1)
INTE A2	PC4	$\overline{STB}$A (Input Mode 1 or Mode 2)
INTE A1	PC6	$\overline{ACK}$A (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.

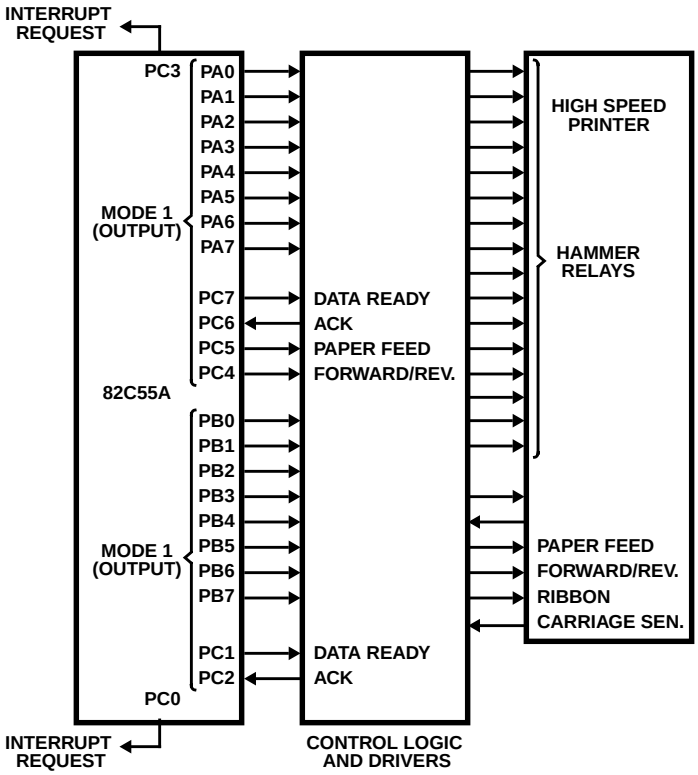


FIGURE 18. PRINTER INTERFACE