AN ARGUS SPECIALIST PUBLICATION

ELECTRONICS DIGEST

Vol.4 No.2
£1.85

MICROPROCESSOR DATA FOR EXPERIMENTERS

CPUs

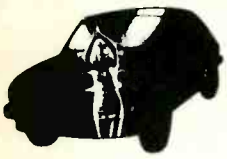
Interface ICs

RAMs

PROMs

From the Publishers of ETI & HE

BACKNUMBERS



February 1980
Passion Meter, Win Indicator, Short Circuit Special, Kit Review Special, Into Electronics Construction Part 1.

May 1980
MiniClocks, 5080 Preamp, Model Railway Track Cleaner, 5080 Loudspeakers, Loudspeaker Crossover Design, Radio Controlled Model Survey.

June 1980
Microbe Radio Control System, Egg Timer, Two Watt Amplifier, Fog Horn, Short Circuits, LEDs and LED Displays.

July 1980
Sound-Operated Flash Trigger, 18+18 Car Stereo Booster, Hazard Flasher, Electronics in Photography, Electronic Espionage, Piezo Electricity.

August 1980
EquiTone Car Equaliser, Pass-The-Loop Game, Gaztec Gas Detector, OP-Amp Checker, In-Car Entertainment Survey, Introducing Microprocessors.

September 1980
MicroMixer, Reaction Tester, Guitar Phaser, Development Timer, Teletext Explained, Into Digital Electronics Part 1.

October 1980
Kitchen Timer, Tug 'o' War Game, Light Dimmer, Freezer Alarm, Intruder Alarm, Temperature-Controlled Soldering Iron.

January 1981
Car Rev-Counter, Bench Amplifier, Sound-Into-Light Converter, Chuffer, Electronic Games reviewed.

February 1981
Heartbeat Monitor, High-Impedance Voltmeter, Medium Wave Radio, Two-Tone Train Horn, Audio Signal Generator.

March 1981
Public Address Amplifier, Windscreen Wiper Controller, Bicycle Speedometer, Photographic Timer, Microcassettes.

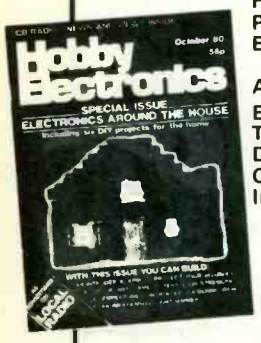
April 1981
Pre-Amplifier Part 1, Super Siren, Guitar Tremolo, Russian Roulette Game, Doorbell Monitor, Anatomy of a Space Shuttle.

May 1981
Electronic Organ, Voice-Operated Switch, Infra-Red Controller, Pre-Amplifier Part 2, Audio Millivoltmeter.

June 1981
Power Amplifier Part 1, Continuity Checker, Envelope Generator, Early Radio, Gadgets, Games and Kits Supplement.

July 1981
Burglar Alarm, Doorbuzzer, Treble Booster, Electronic Aids for the Disabled, Power Amplifier Part 2.

August 1981
Electronic Ignition, Thermometer, Electronic Organ (final part), RPM Meter, Bench Power Supply, Radio Control Survey, Into Electronic Components Part 1.



All of the 1980 issues, except January and April, are still available together with the remaining issues from 1981.

All backnumbers cost £1.50 each. For those of you who only want copies of articles, we do offer a photocopying service. Each copy costs £1.50 and information as to its title and publication date should be given. Ordering backnumbers and photocopies could hardly be easier, just fill in the coupon, cut it out and send it to the appropriate address.

HOBBY ELECTRONICS BACKNUMBER ORDER FORM TO:

Hobby Electronics
513, London Road,
Thornton Heath,
Surrey, CR4 6AR
England.

Please send me the following items:

NAME .....

ADDRESS .....

.....

.....

.....

Back issues .....at £1.50 each

I enclose £.....

Cheques and Postal Orders should be made payable to ASP Ltd.

HOBBY ELECTRONICS PHOTOCOPY ORDER FORM TO:

Hobby Electronics
145 Charing Cross Road,
London,
WC2H 0EE

Please send me the following items:

NAME .....

ADDRESS .....

.....

.....

Photocopies of .....in the

..... issue at £1.50 each

I enclose £.....

Cheques and Postal Orders should be made payable to ASP Ltd.

ELECTRONICS DIGEST

Volume 4 No. 2

GATEWAY TO ELECTRONICS

EDITORIAL OFFICE:
145 CHARING CROSS RD
LONDON WC2H 0EE

Edited and produced by
Helen P. Armstrong

Technical Editor
A. S. Armstrong

Idea by Dave Bradshaw

Managing Editor Ron Harris

Advertisement Manager
David Kitchener

Assistant Advertisement
Manager Joanne James

Managing Director
T. J. Connell

PUBLISHED BY: Argus Specialist Publications Ltd., 145 Charing Cross Road, London WC2H 0EE. DISTRIBUTED BY: Argus Press Sales & Distribution Ltd., 12-18 Paul Street, London EC2A 4JS (British Isles). PRINTED BY: QB Limited, Colchester.
©Argus Specialist Publications Ltd 1983. All material is subject to worldwide copyright protection. All reasonable care is taken in the preparation of the magazine contents, but the publishers cannot be held legally responsible for errors. Where mistakes do occur, a correction will normally be published as soon as possible afterwards. All prices and data contained in advertisements are accepted by us in good faith as correct at time of going to press. Neither the advertisers nor the publishers can be held responsible, however, for any variations affecting price or availability which may occur after the publication has closed for press.
◄ Subscription rates upon application to ETI Subscriptions Department, 145 Charing Cross Road, London WC2H 0EE.

CONTENTS

Note: While every effort has been made by the manufacturers and by the editors to make the data contained in this edition as accurate as possible, no liability in respect of its use is accepted by the editors or the manufacturers.

INTRODUCTION

This is the second of two editions of **Electronics Digest** Data specials – the first, featuring discrete components, linear ICs, CMOS and TTL, appeared in the Spring '82 edition of ED. This volume is dedicated to microprocessors and their support ICs.

The aim in this Data Digest is to provide enough information to enable constructors to sort out which chips they want to use for a particular application. With this in mind we have chosen a selection of CPUs and support chips, mostly (but not exclusively) eight-bit, which are frequently used by amateur constructors and are obtainable from most electronics suppliers. These ICs are simple enough for most amateur designers for most common uses, but not so simple as to be very restricting.

Condensed data is given on a selection of ICs considered to be useful to the amateur constructor and the small industrial user, perhaps foraying into the field of microprocessors seriously for the first time.

The ICs chosen are firstly those most commonly used – the 6800 and 8080 have topped the tables for a time, but data on these has been omitted because there are now more suitable basic ICs in the family for new designs.

Brief information on the much more advanced devices has been included for interest, and to give a perspective on the field, even though it is unlikely that the amateur or newcomer will want to use these ICs at first.

In order to give a reasonable amount of information on the basic processors chosen, only standard variants are specifically referred to. However, most microprocessors are available in different speed variants, and often there are CMOS versions available for low power consumption applications. A good example is the Hitachi HD6300 series, which are CMOS versions of the 6800 series. CMOS processors, being more specialised, cost more than the standard versions.

The data given for each chip is by no means the complete manufacturer's data! In order to get a reasonable selection into the available space, firstly, information which would only be required when carrying out the detailed design of a system has been omitted. Secondly, points that are unusual have been given preference over more pedestrian information which is common to many chips. One example: every microprocessor can add and subtract, but few can multiply. Such special features could be the deciding factor between one chip and another: for example, the presence of a counter-timer and the ability to do reciprocals, to provide rapid, accurate readout of low frequencies.

So the selection of data is not identical for each IC covered. We have used, generally, the points which we think are of greatest interest.

Finally, more information has been provided on ICs in fairly common use, so that a humble PIA may have as much space devoted to it as a complex all-singing, all-dancing disk controller. In any event, once a choice has been made the manufacturer's data sheet will be needed to complete the design.

MICROPROCESSORS

Almost everyone has heard about the so-called 'chip'. The original 'chip' – the integrated circuit – has been around since long before microprocessors appeared, and in many ways the latter can be considered the logical extension of the former.

The microprocessor is different from other ICs, because its high level of integration and programmability open the door to electronic control systems and computers which are economical, yet sophisticated enough to find widespread application.

Among other things, this has led to fears that jobs will be lost to microprocessor-based control systems where previously automatic control equipment would have been prohibitively expensive.

So both in technical terms and in terms of general public interest, microprocessors are significantly different from other ICs.

Firstly, a microprocessor can carry out sequential instructions under program control. One type of processor, with memory and support ICs, can perform an enormous range of tasks. This can safely be mass produced in large enough quantities to make it much cheaper than any moderately complex collection of logic gates. Typically, the microprocessor can carry out logical (AND, OR, etc.) operations, arithmetic instructions (ADD, SUBTRACT, etc.). various shifts and moves, and, of great importance, Conditional Jumps.

A Conditional Jump is an instruction which will transfer an execution to another part of the program if a certain condition is/is not met. Simple examples are 'branch if zero', 'branch if not zero', 'branch if plus', 'branch if minus', 'branch if equal', or 'branch if not equal'.

This type of instruction is essential – without this, it could perform only limited sets of predetermined functions, and would be equivalent only to a set of combinational logic, or perhaps a set of cam operated switches driven by a clock motor!

To carry out its functions, the microprocessor has internal

registers. These are used to store data while it is being worked on. Two essential registers are the Accumulator, and the Program Counter.

The Accumulator is used in virtually all logic and arithmetic operations. Some processors have more than one register able to be used as the accumulator, and certain eight bit processors can do some sixteen bit operations.

The program counter stores the address of the next instruction to be executed. It is automatically incremented on fetching the previous instruction. In an eight bit processor, the PC is normally sixteen bits wide.

Other registers of which most simple processors have a selection are: Stack Pointer (SP), Index Registers (IX), an extra Accumulator (ACCB) and a Condition Code Register (CC).

In writing a program of any complexity, there are sequences of instructions, eg. to multiply two eight bit numbers, that are used several times. It's often more convenient (and comprehensible) to use subroutines which will carry out these sequences of instructions each time they are called.

The subroutine resides at an address clear of the main program, and in order to return to the same place in the main program, the PC and any other registers affected by the subroutine are stored in specific order on a Stack.

A Stack is an area of memory reserved by the programmer for such use, and as each register is added, the SP is counted onwards by one, to point to the address of the last register stored.

Once the subroutine is finished, the return to the main program is carried out by loading registers back from the stack in reverse order. It is possible to have one subroutine call another in, and have an orderly return through the various levels. The stack can grow quite large in this way.

Anyone who has written programs in BASIC will be familiar with the concept of arrays: they are tables of data to be processed. To provide a similar function on a microprocessor the IX may be used. The number in this may be output onto the address bus, and may be incremented or decremented to address data in sequential order.

The Condition Code register has its individual bits set to indicate conditions such as overflow, carry etc. It may also have bits set to indicate results of comparisons.

The Bus is a group of eight, sixteen or however many lines to carry data, addresses, etc., often in either direction. Some processors have multiplexed buses in which, for example, eight of the sixteen address lines are mutiplexed with the data, and one signal is provided to indicate when the lines are used for address or data.

Another important function of microprocessors is Interrupts. If a microprocessor-based system must respond to the operator pushing a switch, but must (of course) be carrying out other tasks meanwhile, the switch may be connected to an interrupt pin.

When the pin gets a signal, a subroutine called Interrupt Service Routine will be called to take appropriate action.

A microprocessor must have support ICs of various types. The most essential is the program memory, normally stored on a Programmable Read Only Memory (PROM). It may also need input/output (I/O) chips, for example to light up a display, or accept input from a keyboard.

Commonly used PROMs are erasable by UV light. Sunlight will not quickly erase them, as the dominant wavelengths are not short enough, but an opaque adhesive label over the window preserves data for a longer period. Typical storage times of ten years are quoted.

The information is stored as an electronic charge on an isolated piece of metalization on the chip, arranged so that the charge may bias On or Off a FET. The charge is placed on the gate by application of a high voltage pulse which causes electrons to tunnel through the insulating layer which they could not normally cross. UV excites the bound electrons in the insulation so that the charge can slowly leak away, as though through a very high-value resistor.

There are many processors available – from four to 32 bits – there is even a one-bit controller IC, the MC14500 which could be considered a microprocessor!

The main ones included here are members of the 8080 family and the 6800 family. Some sample comparisons can be made:

- The 8080 has a good selection of internal registers which can often be used in pairs for double byte operations.

- The support ICs for the 8080/8085 group are "mix-and-match" – for example, one type, the 8155/8156, has 22 I/O lines (eight plus eight plus six), a timer and 256 bytes of RAM included within it.

- By contrast, the 6802 is stronger on indexed addressing and operations directly on memory, where the 8085 is weak. Its 256 bytes of built-in RAM has a battery backup mode to retain the information while the power is off – the current consumption in this mode is, of course, very low.

- The 6802 does not have the 8085's good selection of internal registers, nor are its support chips so much the "mixed flavours" types. However, the nearest equivalent to the 8155, the 6821, is an I/O port which can have its individual pins programmed as input or output, while the 8155 can only be programmed in three blocks.

- The 6502 has much hardware compatibility with the 6802, but its instruction set is different and includes some BCD operations, making it more suitable for home computers.

- The TMS9995 is a sixteen bit processor, but is organised to eight bit memories. It has many advanced features, such as an on-chip timer, RAM, etc. It could be compared with the more advanced chips from the other ranges, such as the 8088, 6809.

Though for most microprocessor families there is a primary manufacturer, the originator of the family, often a number of other manufacturers produce equivalent chips, which may be a minor variation or enhancement of the original. In order to give an idea of what is available, a selection of different manufacturers' data has been used for the different processors within each family.

Now read on . . .

FAIRCHILD F6800 SERIES General Information

The 6800 was originally developed by Motorola at about the same time that Intel introduced the 8080. Unlike other processors of its day, the 6800 had a single supply rail. Its level of popularity was similar to that of the 8080. Developments of the 6800 are currently popular and useful, and the family is now covered by many manufacturers.

The Fairchild F6800 microprocessor family is a set of 8-bit MOS devices that offers a complete and constantly growing selection of microprocessors having a powerful instruction set. As shown in figure 5-1, the F6800 family now includes seven different CPUs (described in table 5-1), supported by such circuits as synchronous and asynchronous controllers for data communications, timers, a direct memory access controller, CRT controllers, RAMs, ROMs, and EPROMs (described in table 5-2).

Table 5-1 F6800 Microprocessor Family CPUs

Device No.	No. of Pins	Power Supply	External Addressing	Data Length (Bits)	Clock	No. of Basic Instructions	Bytes (RAM)	Bytes (ROM)	No. of I/O Lines	Other I/O	Timer
F6800	40	+5 V	64K	8	No	—	—	—	—	—	
F6801	40	+5 V	64K	8	Yes	82	128	2K	31	Serial	16-Bit
F6802	40	+5 V	64K	8	Yes	72	128	—	—	—	—
F6803	40	+5 V	64K	8	Yes	82	128	—	13	Serial	16-Bit
F6808	40	+5 V	64K	8	Yes	72	—	—	—	—	—
F6809	40	+5 V	64K	8	Yes	59	—	—	—	—	—
F6882	40	+5 V	64K	8	Yes	72	128	—	—	—	—

Table 5-2 F6800 Peripheral Devices

Type	Number	Name	Comment
General-Purpose	F6820	Peripheral Interface Adapter	Twenty I/O Lines
General-Purpose	F6821	Peripheral Interface Adapter	Twenty I/O Lines
General-Purpose	F6840	Programmable Timer Module	Three-to 16-Bit Timers
General-Purpose	F68488	General-Purpose Interface Adapter	IEEE-488 Bus Controller
Special Function	F6844	Direct Memory Access Controller	Three I/O Channels
Special Function	F6845	CRT Controller	Available in Interlace or Non-Interlace
Special Function	F6846	ROM, I/O, Timer	2K x 8 ROM, Parallel I/O, Timer
Special Function	F6847	Video Display Generator	Low-Cost Video Controller
Data Communications	F6850	Asynchronous Communications Interface Adapter	
Data Communications	F6852	Synchronous Serial Data Adapter	
Data Communications	F6854	Advanced Data Link Controller	HDLC/SDLC
Data Communications	F6856/ F3846	Synchronous Communications Protocol Controller	HDLC/SDLC/BTSYNC
Data Communications	F68456/ F38456	Multi-Protocol Communications Controller	HDLC/SDLC/BISYNC/ASYNC
Memory	F6810	128 x 8-Bit Static RAM	

Data courtesy of Fairchild Semiconductors Ltd.

This is our interpretation of Fairchild data on devices in their F6800 family and is believed to be accurate and reliable. However, neither Fairchild nor ourselves can assume responsibility for its use or for use of any circuitry described. For more current, detailed information, please contact one of their franchised distributors, not Fairchild themselves.

-1 F6800 Family Organization

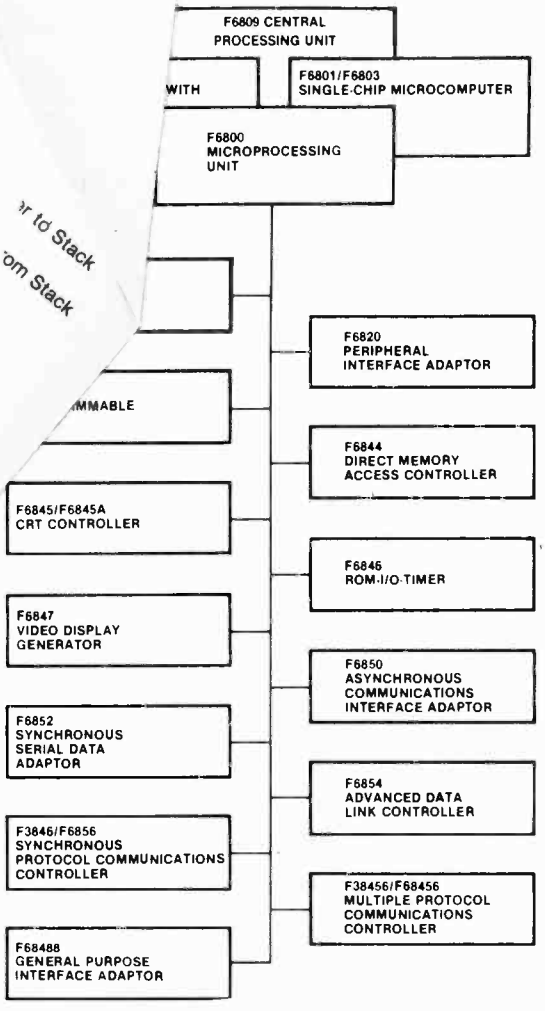


Figure 5-2 F6800/F6802/F6808/F6882 Programming Model

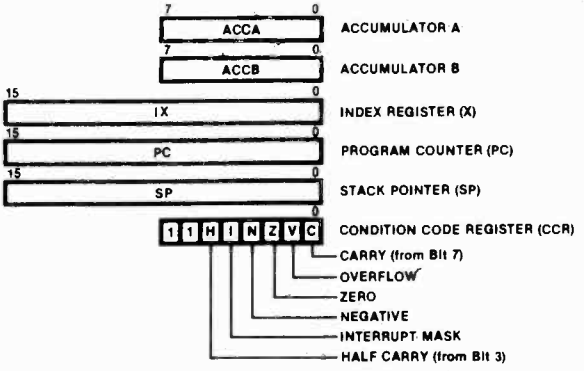
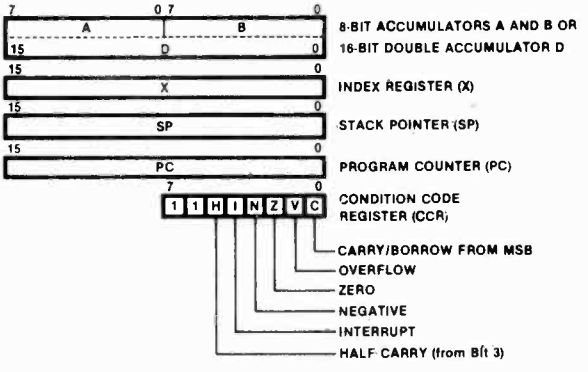


Figure 5-3 F6801/F6803 Programming Model



Instruction Set

Because a single instruction set is inadequate for the number and flexibility of devices in the F6800 family, it has been necessary to develop three such sets, each serving a portion of the family.

The basic instruction set, comprising 72 instructions, is supported by the F6800, F6802, F6808, and F6882; figure 5-2 is the associated programming model. An expanded instruction set, consisting of the basic set plus several additional instructions, is supported by the F6801 and F6803; figure 5-3 illustrates the associated programming model. The expanded instruction set is upward-compatible with the basic set (that is, programs written using either are interchangeable, provided that the additional instructions are not involved). Both the basic and expanded instruction sets are described in table 5-3.

The instruction set supported by the high-performance F6809 is similar in structure to the basic and expanded sets, but is not upward-compatible. It is greatly enhanced to take fullest advantage of the powerful F6809 architecture. Figure 5-4 illustrates the F6809 programming model and table 5-4 describes the instruction set.

Figure 5-4 F6809 Programming Model

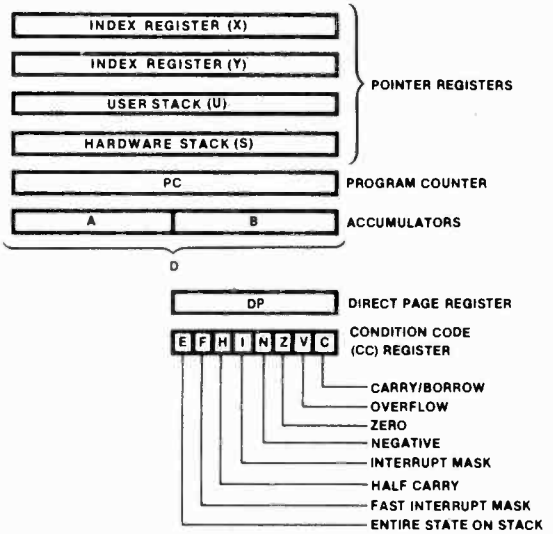


Table 5-3 Basic and Expanded Instruction Sets

Instruction	Description
ABA	Add Accumulators
*ABX	Add Accumulator B to Index Register
ADC	Add With Carry
ADD	Add
*ADDD	Add Double Accumulator to Memory; Leave Sum In Double Accumulator
AND	Logical AND
ASL	Arithmetic Shift Left
*ASLD	Double Accumulator Shift Left; Clear LSB; Shift MSB into C-Bit
ASR	Arithmetic Shift Right
BCC	Branch if Carry Clear
BCS	Branch if Carry Set
BEQ	Branch if Equal To Zero
BGE	Branch if Greater Than or Equal To Zero
BGT	Branch if Greater Than
BHI	Branch if Higher Than
*BHS	Branch if Higher Than or Same As
BIT	Bit Test
BLE	Branch if Less Than or Equal To
*BLO	Branch if Lower Than
BLS	Branch if Lower Than or Same As
BLT	Branch if Less Than Zero
BMI	Branch if Minus
BNE	Branch if Not Equal To Zero
BPL	Branch if Plus
BRA	Branch Always
*BRN	Branch Never
BSR	Branch to Subroutine
BVC	Branch if Overflow Clear
BVS	Branch if Overflow Set
CBA	Compare Accumulators
CLC	Clear Carry
CLI	Clear Interrupt Mask
CLR	Clear
CLV	Clear Overflow
CMP	Compare
COM	Complement
CPX	Compare Index Register
*CPX	Compare Index Register; Permits Use With Any Conditional Branch Instruction
DAA	Decimal Adjust
DEC	Decrement
DES	Decrement Stack Pointer
DEX	Decrement Index Register
EOR	Exclusive OR
INC	Increment
INS	Increment Stack Pointer
INX	Increment Index Register
JMP	Jump
JSR	Jump to Subroutine
*JSR	Additional Addressing Mode Direct
LDA	Load Accumulator
*LDD	Load Double Accumulator from Memory
LDS	Load Stack Pointer
LDX	Load Index Register
*LSL	Memory or Accumulator Shift Left; Clear LSB; Shift MSB into C-Bit
*LSLD	Double Accumulator Shift Left; Clear LSB; Shift MSB into C-Bit
LSR	Logical Shift Right
*LSRD	Double Accumulator Shift Right; Clear MSB; Shift LSB into C-Bit
*MUL	Multiply Accumulators; Leave Product in Double Accumulator
NEG	Negate
NOP	No Operation
ORA	Inclusive OR Ac
PSH	Push Data
*PSHX	Push Index Regist
PUL	Pull Data
*PULX	Pull Index Register f
ROL	Rotate Left
ROR	Rotate Right
RTI	Return from Interrupt
RTS	Return from Subroutine
SBA	Subtract Accumulators
SBC	Subtract With Carry
SEC	Set Carry
SEI	Set Interrupt Mask
SEV	Set Overflow
STA	Store Accumulator
*STD	Store Double Accumulator
STS	Store Stack Register
STX	Store Index Register
SUB	Subtract
*SUBD	Subtract Double Accumulator;
SWI	Software Interrupt
TAB	Transfer Accumulators
TAP	Transfer Accumulators to Condition Code Register
TBA	Transfer Accumulators
TPA	Transfer Condition Code Register to Accumulator
TST	Test
TSX	Transfer Stack Pointer to Index Register
TXS	Transfer Index Register to Stack Pointer
WAI	Wait for Interrupt

FAIRCHILD F6802/6882 CPU

The **6802** and **6808** are enhancements of the **6800**, and are probably always to be preferred to the 6800 for new designs. The most important differences are an internal clock generator on both chips, and internal RAM on the 6802.

Data courtesy of Fairchild Semiconductors Ltd.

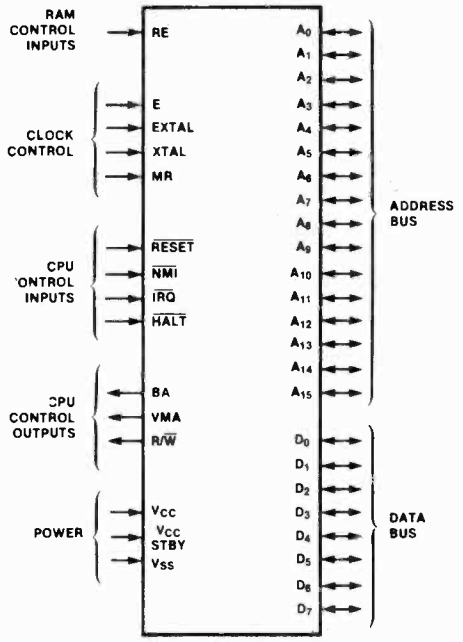
Description

The F6802/F6882 is a monolithic 8-bit microprocessor that contains all the registers and accumulators of the F6800, plus an internal clock oscillator and driver on the same chip. The F6802/F6882 also has 128 bytes of RAM on board, located at hex addresses $0000 to $007F, V_{CC} standby can be utilized on the F6802/F6882 to facilitate memory retention during a power-down situation; the first 8 bytes of RAM at hex addresses $0000 to $0007 can be retained on the F6882, and the first 32 bytes of RAM at hex addresses $0000 to $001F can be retained on the F6802. The F6808 is identical to the F6802 without on-board RAM.

The F6802/F6882 is completely software-compatible with the F6800 microprocessor and the entire F6800 family of parts. (*Figure 1* illustrates a typical application using an F6800 family device.)

- **On-Chip Clock Circuit**
- **128 x 8-bit On-Chip RAM (Not Included on F6808)**
- **8 Bytes of RAM are Retainable on the F6882**
- **32 Bytes of RAM are Retainable on the F6802**
- **Software-Compatible with the F6800**
- **Standard TTL-Compatible Inputs and Outputs**
- **8-bit Bidirectional Data Bus**
- **16-bit Memory Addressing**
- **Interrupt Capability**
- **Three Speed Grades:**
  - **1.0 MHz F6802/F6882/F6808**
  - **1.5 MHz F68A02/F68A82/F68A08**
  - **2.0 MHz F68B02/F68B82/F68B08**

F6802/F6882/F6808 Signal Functions



Connection Diagram

40-Pin DIP

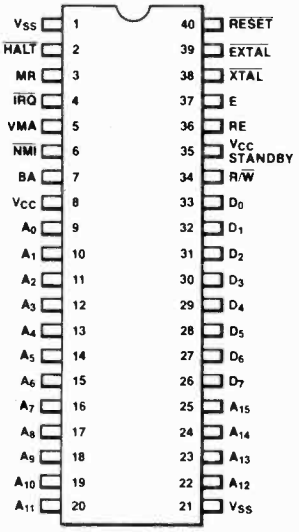
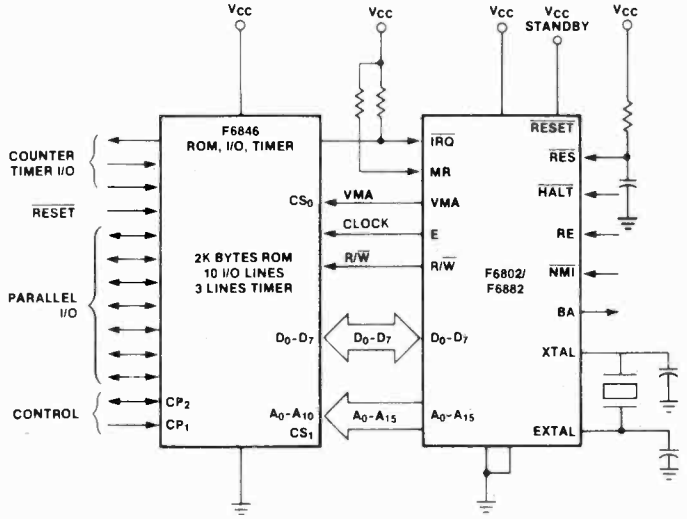


Fig. 1 Typical Microcomputer Block Diagram



F6802 CPU

Registers

A general block diagram of the F6802/F6882 is shown in *Figure 2*. The number and configuration of the registers are identical to the F6800, as shown, with a 128 x 8-bit RAM* added to the basic microprocessor. The first 8 bytes in the F6882 and the first 32 bytes in the F6802 may be operated in a low-power mode via a V_{CC} standby and can be retained during power-up and power-down conditions via the RE signal. The F6808 is identical to the F6802 except for on-board RAM. Since the F6808 does not have on-board RAM, pin 36 must be tied to ground, allowing the processor to utilize up to 64K bytes of external memory.

The microprocessing unit (MPU) has three 16-bit registers and three 8-bit registers available for use by the programmer

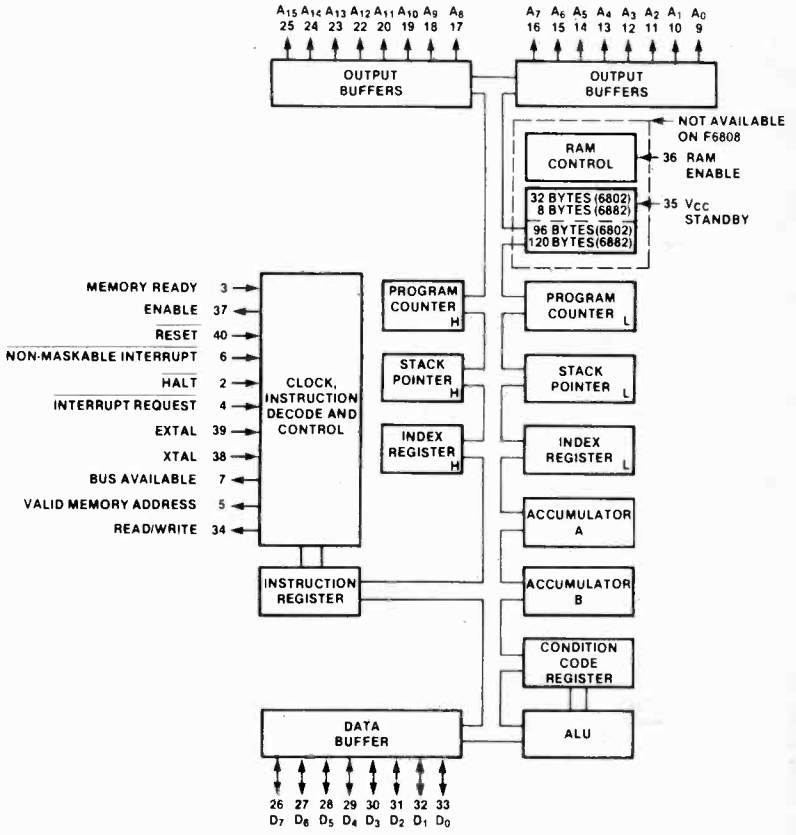
Program Counter
The program counter is a 2-byte (16-bit) register that points to the current program address.

Stack Pointer
The stack pointer is a 2-byte register that contains the address of the next available location in an external push-down/pop-up stack. This stack is normally a random access read/write memory that may have any location (address) that is convenient. In those applications that require storage of information in the stack when power is lost, the stack must be non-volatile.

Index Register
The index register is a 2-byte register that is used to store data or a 16-bit memory address for the indexed mode of memory addressing.

Accumulators
The two 8-bit accumulators are used to hold operands and results from an arithmetic logic unit (ALU).

F6802/F6882 Block Diagram



Condition Code Register (Status Word Register)
The condition code register indicates the results of an arithmetic logic unit operation: negative (N), zero (Z), overflow (V), carry from bit 7 (C), and half-carry from bit 3 (H). These bits of the condition code register are used as testable conditions for the conditional branch instructions. Bit 4 is the interrupt mask bit (I). The unused bits of the condition code register (bit 6 and bit 7) are binary ones (1).

$\overline{NMI}$ (Non-Maskable Interrupt), Pin 6
A low-going edge on this input requests that a non-maskable interrupt sequence be generated within the processor. As with the interrupt request ($\overline{IRQ}$) signal, the processor completes the current instruction being executed before it recognizes the $\overline{NMI}$ signal. The interrupt mask bit in the condition code register has no effect on $\overline{NMI}$.

The index register, program counter, accumulators, and condition code register are stored on the stack, as shown in *Figure 4*. At the end of the cycle, a 16-bit address will be loaded from memory locations $FFFC and $FFFD that points to a vectoring address. An address loaded from these locations causes the MPU to branch to a non-maskable interrupt routine in memory.

$\overline{IRQ}$ (Interrupt Request), Pin 4
This level-sensitive input requests that an interrupt sequence be generated within the machine. The processor waits until it completes the current instruction that is being executed before it recognizes the request. At that time, if the interrupt mask bit in the condition code register is not set, the machine begins an interrupt sequence. The index register, program counter, accumulators, and condition code register are stored on the stack

The MPU responds to the interrupt request by setting the interrupt mask bit high so that no further interrupts may occur. At the end of the cycle, a 16-bit address is loaded from memory locations $FFF8 and $FFF9 that point to a vectoring address. An address loaded from these locations causes the MPU to branch to an interrupt routine in memory.

The $\overline{HALT}$ line must be in the high state for interrupts to be serviced. Interrupts are latched internally while $\overline{HALT}$ is low.

$\overline{\text{HALT}}$ (Halt), Pin 2
When this input is in the low state, all activity in the machine is halted. This input is level-sensitive. In the halt mode, the machine stops at the end of an instruction. Bus Available is in a high state, and Valid Memory Address is in a low state. The address bus displays the address of the next instruction.

To ensure single-instruction operation, transition of the $\overline{\text{HALT}}$ line must occur t_{PCS} before the falling edge of E and the $\overline{\text{HALT}}$ line must go high for one clock cycle.

RE (RAM Enable), Pin 36
A TTL-compatible RAM enable input that controls the on-chip RAM. When placed in the high state, the on-chip memory is enabled to respond to the MPU controls. In the low state, the RAM is disabled. This pin may also be utilized to disable reading from and writing to the on-chip RAM during a power-down situation. The RE signal must be low three cycles before V_{CC} goes below 4.75 V during power-down as shown

V_{CC} STBY (Power Supply Standby), Pin 35
This pin supplies the dc voltage to the first 8 or 32 bytes of RAM as well as the RAM enable (RE) control logic. Thus, retention of data in this portion of the RAM on a power-up, power-down, or standby condition is guaranteed.

Extended Addressing
In extended addressing, the address contained in the second byte of the instruction is used as the higher 8 bits of the address of the operand. The third byte of the instruction is used as the lower 8 bits of the address for the operand. This is an absolute address in memory. These are 3-byte instructions.

Indexed Addressing
In indexed addressing, the address contained in the second byte of the instruction is added to the index register's lowest 8 bits in the MPU. The carry is then added to the higher order 8 bits of the index register. This result is then used to address memory. The modified address is held in a temporary address register so there is no change to the index register. These are 2-byte instructions.

Implied Addressing
In the implied addressing mode, the instructions give the address (i.e., stack pointer, index register, etc.). These are 1-byte instructions.

Relative Addressing
In relative addressing, the address contained in the second byte of the instruction is added to the program counter's lowest 8 bits plus two. The carry or borrow is then added to the higher 8 bits. This allows the user to address data within a range of – 125 to + 129 bytes of the present instruction. These are 2-byte instructions.

HEMMINGS ELECTRONICS LTD

Electronic Components & Microcomputers
DEPT ED, 16 BRAND STREET, HITCHIN, HERTS SG5 1JE
Telephone: (0462) 33031

Shop opening hours:
Mon. to Fri.
9am to 5.30pm
Sat. 9am to 5pm.
Wed. Closed

TERMS OF BUSINESS: Professional quality electronic components brand new and fully guaranteed. Mail order by return of post. Cash, Cheque, Postal Order or Banker's Draft with order payable to HEMMINGS ELECTRONICS LTD. Access or Barclaycard available using our 24-hour answerphone service. Monthly Credit Accounts available on request to government and educational authorities. Industrial customers wishing to open a credit account are required to furnish a bank reference and two trade references. Postage and packing add 45p to all orders under £10. All prices are exclusive of VAT. Please add 15% to total including p&p. No VAT on export orders or books.

COMPUTER ICs

6502	350p
6522	340p
6800	290p
68B00	450p
6802	345p
6809	845p
68B09	1350p
6809E	1295p
6810	120p
6821	180p
68B21	215p
6840	390p
68B40	580p
6844	1295p
6845	795p
6850	140p
6852	250p
6854	680p
6875	490p
8T26A	120p
8T2B	120p
8T95	90p
8T96	90p
8T97	90p
8T98	90p
8035L	340p
8039L	290p
8080A	360p
8085A	450p
8155	450p
8212	155p
8216	100p
8224	180p
8226	195p
8228	250p
8251	300p
8254	450p
8255	280p
8257	450p
8259	450p
8279	450p
75107	90p
7510B	90p
75110	88p
75112	160p
75182	95p
75450	85p
75451	50p
75452	50p
75453	72p
75461	40p
75491	70p
75492	70p
AY-3-1015D	300p
AY-5-1013A	300p
MC1408	295p
MC1488	55p
MC1489	55p
MC3459	265p
UPD7002	450p
Z80ACPU	350p

MEMORIES
Static RAM

2114L-200nS	90p
6116P3-150nS	390p
6116LP3-150nS	450p

Dynamic RAM

4116-200nS	85p
TMS4164-200nS	500p

Eprom

2708-450nS	220p
2716-450nS	250p
2532-450nS	350p
2732-450nS	320p
2764-450nS	495p

FLOPPY DISC CONTROLLERS

FD1791	1950p
UPD765A	1650p

CRT CONTROLLER

SFF96364	800p

ZENER DIODES
BZYBB Series
500mW E24
2V7 to 39V 8p
43V to 110V 12p
BZX61 Series
1.3W E24
2V7 to 39V 15p
43V to 82V 20p

BRIDGE RECTIFIERS

1A/100V	25p
1A/400V	30p
1A/800V	40p
2A/100V	40p
2A/400V	50p
2A/800V	70p
6A/400V	95p
10A400V	280p
35A400V	315p
BY164	52p

TRIACS

TIC206D	55p
TIC2226E	95p
T2800D	95p

DIACS

29-37V	25p

Z80AP10	300p
Z80ACTC	300p
Z80ADART	750p

PCB TRANSFERS
Make your own Printed Circuit Boards with Alfac Etch Resist PCB Transfers.
- Draw your artwork on 0.1" Grid
- Transfer to Copper Board using Carbon Paper
- Burnish the Alfac Transfers to the Board using a Spatula using Carbon marks to assist in accurate alignment
- Use Alfac Chemical Eraser to correct mistakes
- Etch in Ferric Chloride

EC900/1 0.1" Edge Connector
EC902/1 0.156" Edge Connector
EC908 0.063" Pads
EC910 0.094" Pads
EC911 0.189" Pads
EC940 0.016" Lines
EC941 0.031" Lines
EC942 0.039" Lines
EC943 0.049" Lines
EC944 0.061" Lines
EC945 0.079" Lines
EC946 0.100" Lines
EC947 0.124" Lines
EC950/1 0.031" 90 Bends
EC950/2 0.061" 90 Bends
EC951/1 0.031" 30, 45, 60 Bends
EC952/2 0.061" 30, 45, 60 Bends
EC960/1 TO-5 Transistor Pads
EC993/1 IC Pads
EC997/1 IC Pads with Tracks between Pads
5 Identical Sheets in Sealed Pack 195p
Individual Sheets 45p
Spatula AR4 for Burnishing 45p
Alfac Knife 180p
Spare Blades (pack of 10) 280p
Alfac Chemical Eraser 40p
Alfac Precision Grids Polyester Film, matt finish, 0.14mm thickness
20 lines/inch A4 100p
A3 195p
Double Sided Fibreglass Board 1/16" thickness
1 oz Copper 5" 4" 35p
5" 4" 60p
Dalo Etch Resist Pen 85p
Ferric Chloride Crystals, dissolve in ½ litre water 85p

LIQUID CRYSTAL DISPLAY MODULES
PCIM177 Frequency Counter. 5 digits, 0.35 FM, SW, HW, MHz. KHz Annunciators. Sample and Hold capability. Reset capability. 25 selectable IF Offsets. Prescaler available. Incandescent backlighting option. Supply voltage 5V. Operating current 4mm 1715p
PCM176 Digital Voltmeter 3½ digits, 0.5", +, – and decimal point. 200mV full scale input. True differential input. Guaranteed 'O' reading. Single 9V operation. Power consumption 20mW. Accuracy 0.15% ±1 Count. Temperature drift 80ppm/C. Low battery indicator. Incandescent backlighting 1950p
Both modules are supplied with a data sheet.

TEMPERATURE METER
A fully self-contained digital temperature meter, battery operated with an LCD display.
- Temperature range 0-99. 9 C
- Accuracy 0-40 C ±0.2 C
  40-70 C ±0.4 C
  70-99.9 C ±1.0 C
- Battery 9V alkaline
  Lifetime approx 1 year
- External temperature probe £19.95

CRYSTALS

32.768KHz	100p
1.0000MHz	320p
1.8432MHz	240p
2.0000MHz	225p
2.457MHz	225p
3.0000MHz	240p
3.5795MHz	120p
3.6864MHz	240p
4.000MHz	150p
4.1943MHz	190p
5.0688MHz	240p
6.000MHz	180p
6.880MHz	240p
8.000MHz	180p
10.000MHz	170p
16.000MHz	190p
18.432MHz	150p
19.6608MHz	240p
20.000MHz	200p
27.000MHz	170p
48.000MHz	170p

LITESOLD ARE OFFERING A COMPLETE SOLDERING KIT FOR ONLY

£14.55 inc.VAT & P.P.

Contains all you need to solder/de-solder any electronics project:
LITESOLD LC18H 240v high performance iron, made to professional standards in our own works, fitted with 3.2mm bit. 2 alternative bits, 1.6 and 2.4mm. Reel of 3 metres 18 swg flux-cored solder. Stainless steel tweezer. 3 soldering aids. Reel of 1.5 metres de-soldering braid. Packed in clear PVC presentation/storage wallet. Superb present — ideal for beginner or expert.
SK18H KIT SPECIAL PRICE **£14.55** inc. VAT & P.P. (normal resale value £18.45 inc.) LC18H IRON only **£6.00** (normally £7.39)
Spares, accessories and after-sales service available from us.
16-page colour catalogue — 60p. Send cheque/P.O. to LITESOLD or ring for Access/Barclaycard sales.

Access VISA

LITESOLD
LIGHT SOLDERING DEVELOPMENTS LTD.
Spencer Place, 97/99 Gloucester Road, Croydon CR0 2DN, Surrey. Tel: 01-689 0574.

MOTOROLA MC6803 CPU

The **6801,** a version of the 6800, is a mask programmed single chip MPU. It can be programmed to carry out, for example, simple control functions, all on its own. Being mask programmed this is of little use to home constructors or people needing an economical MPU in modest quantities.

The **6803** featured here is similar, except that it uses an external PROM, and so is ideal for this sort of use. The 6803 is an enhancement of the 6800 itself, and has 10 new instructions, including an eight bit multiply instruction.

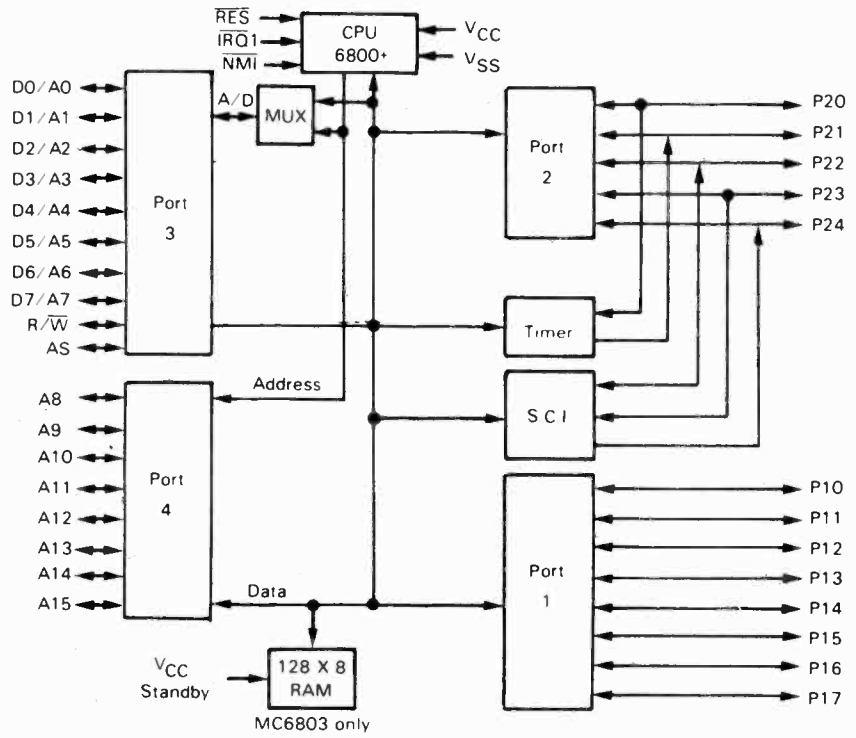
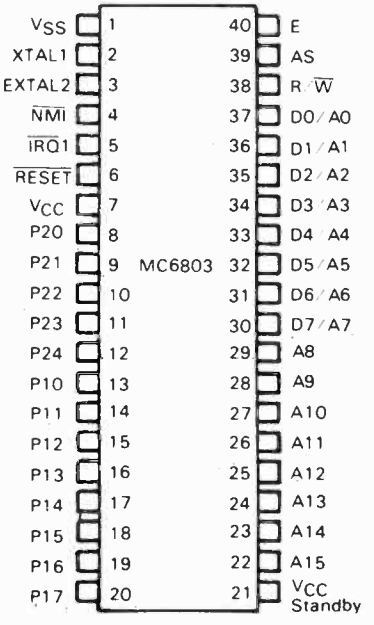


FIGURE 1 — BLOCK DIAGRAM

The MC6803 is an 8-bit microcomputer which employs a multiplexed address and data system, allowing expandability to 64K words. The MC6803 is object code compatible with the M6800 instruction set and includes improved execution times of key instructions. There are several new 16-bit and 8-bit instructions including an 8 by 8 multiply with 16-bit result. The MC6803 has 128 bytes of RAM, internal clock, SCI, parallel I/O, and three function 16-bit timer all on-board. The MC6803 requires only the addition of a ROM and an external crystal for MCU operation. The MC6803 internal clock's divide by four circuitry allows for use of the inexpensive 3.58 MHz color-burst crystal. The MC6803 MCU is fully TTL compatible and requires only one +5.0 volt power supply. An external RAM is needed with the MC6803 NR.

- Expanded M6800 Instruction Set
- Full Object Code Compatibility With M6800 MPU's
- Multiplexed Address and Data
- Compatible With Existing M6800 Peripherals
- 8 X 8 Multiply With 16-bit Result
- Up to 13 Parallel I/O Lines
- 128 Bytes On-Board RAM on MC6803
- On-Board RAM Retainable With V_{CC} Standby
- Serial Communications Interface On-Board
- 16-bit Timer On-Board
- Internal Clock/Divide by Four Circuitry
- Full TTL Compatibility
- Full Interrupt Capability

FIGURE 2 — PIN ASSIGNMENT



Some Signal Descriptions

V_{CC} Standby (MC6803)

This input pin should supply +5 volts ±5% to the standby RAM on the chip. The first 64 bytes of RAM will be maintained in the power down mode with 8 mA current max. The circuit of Figure 11 can be utilized to assure that V_{CC} Standby does not go below V_{SBB} during power down.

Address Strobe (AS)

This output is used to latch the 8 LSB's of address which are multiplexed with data. An 8-bit latch is utilized in conjunction with Address Strobe, as described in Figure 13. Address Strobe signals the time to latch the address so the lines can output data during the E pulse. Timing Figure 5. This signal is also used to disable the address from the multiplexed bus allowing a deselect time T_{ASD} before the data is enabled to the bus.

Read/$\overline{\text{Write}}$ (R/$\overline{\text{W}}$)

This TTL compatible output indicates to the peripherals and memory devices that the MPU is in a Read (high) or a Write (low) state. The normal state of this signal is Read (high).

Address/Data Bus (A0/D0-A7/D7)

Eight pins are used for the multiplexed address and data bus. These pins provide the lower order address lines plus the 8-bit bidirectional data bus.

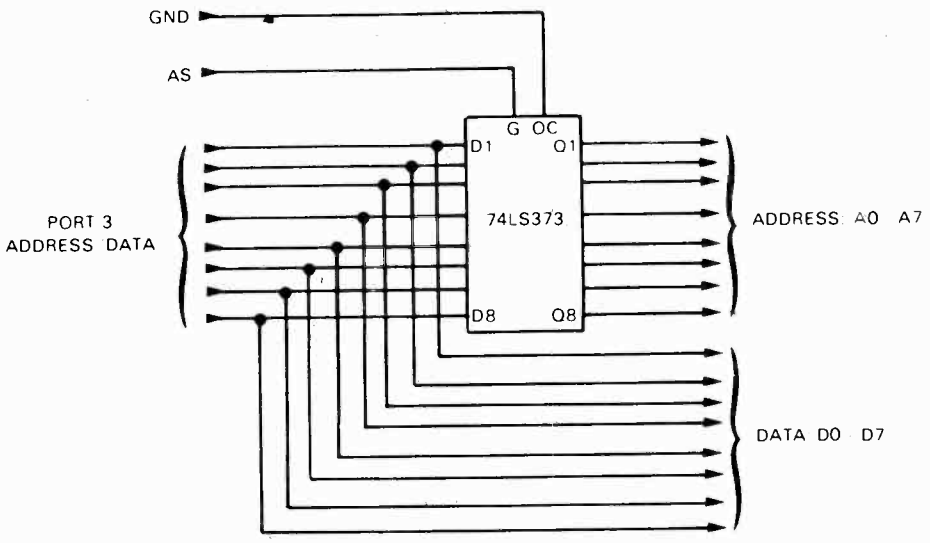
An external latch may be supplied by the user in order to supply the full 16-bit non-multiplexed address lines. The latch is controlled by the address strobe.

The SN74LS373 Transparent octal D-type latch can be used with MC6803/MC6803NR to latch the least significant address byte. Figure 13 shows how to connect the latch to MC6803.

Address Bus (A8-A15)

These eight lines output the higher order address lines allowing for the full 64K word expandability.

FIGURE 13 — LATCH CONNECTION



FUNCTION TABLE

OUTPUT CONTROL	ENABLE G	D	OUTPUT Q
L	H	H	H
L	H	L	L
L	L	X	Q_0
H	X	X	Z

Data courtesy of Motorola Ltd. © Motorola.

PROGRAMMABLE TIMER

The MC6803/MC6803NR contains an on-chip 16-bit programmable timer which may be used to perform measurements on an input waveform while independently generating an output waveform. Pulse widths for both input and output signals may vary from a few microseconds to many seconds. The timer hardware consists of:

- an 8-bit control and status register,
- a 16-bit free running counter,
- a 16-bit output compare register, and
- a 16-bit input capture register

A block diagram of the timer registers is shown in Figure 16.

Free Running Counter ($0009:000A)

The key element in the programmable timer is a 16-bit free-running counter which is incremented by the MPU clock. The counter value may be read by the MPU software at any time. The counter is cleared to zero on $\overline{\text{RESET}}$ and may be considered a read-only register with one exception. Any MPU write to the counter's address ($09) will always result in a preset vlaue of $FFF8 being loaded into the counter regardless of the value involved in the write. This preset feature is intended for testing operation of the part, but may be of value in some applications. However, this will also adversely affect operation of the SCI.

Output Compare Register ($000B:000C)

The Output Compare Register is a 16-bit read/write register which is used to control an output waveform. The contents of this register are constantly compared with the current value of the free running counter. When a match is found, a flag is set (OCF) in the Timer Control and Status Register (TCSR) and the current value of the Output Level bit (OLVL) in the TCSR is clocked to the output level register. Providing the Data Direction Register for Port 2 Bit 1 contains a "1" (output), the output level register value will appear on the pin for Port 2 Bit 1. The values in the Output Compare Register and Output level bit may then be changed to control the output level on the next compare value. The Output Compare Register is set to $FFFF during $\overline{\text{RESET}}$. The Compare function is inhibited for one cycle following a write to the high byte of the Output Compare Register to insure a valid 16-bit value is in the register before a compare is made.

Input Capture Register ($000D:000E)

The Input Capture Register is a 16-bit read-only register used to store the current value of the free running counter when the proper transition of an external input signal occurs. The input transition change required to trigger the counter transfer is controlled by the Input Edge bit (IEDG) in the TCSR. The Data Direction Register bit for Port 2 Bit 0, should* be cleared (zero) in order to gate in the external input signal to the edge detect unit in the timer.

*With Port 2 Bit 0 configured as an output and set to "1", the external input will still be seen by the edge detect unit.

Timer Control and Status Register (TCSR) ($0008)

The Timer Control and Status Register consists of an 8-bit register of which all 8 bits are readable but only the low order 5 bits may be written. The upper three bits contain read-only timer status information and indicate that:

- a proper transition has taken place of the input pin with a subsequent transfer of the current counter value to the input capture register, (bit 7).
- a match has been found between the value in the free running counter and the output compare register, (bit 6).
- when the free running counter has overflowed from $FFFF to $0000 (bit 5).

Each of the flags may be output to the MC6803 internal Interrupt Request (IRQ2) with an individual Enable bit in the TCSR. If the I-bit in the MC6803/MC6803NR Condition Code register is clear, a priority vectored interrupt will occur corresponding to the flag bit(s) set. A description for each bit follows:

PERIPHERAL INTERFACE LINES

The MC6803/MC6803NR provides an 8-bit port and a 5-bit port for interfacing to peripheral devices. They are bidirectional in that each bit can be programmed as either an input or an output by writing to the associated bit in the port's Data Direction Register. A "1" in the corresponding Data Direction Register bit will cause that I/O line to be an output. A "0" in the corresponding Data Direction Register bit will cause the I/O line to be an input. There is one exception in Port 2. Bit "1" of Port 2 can be either an input line or the timer output, which precludes its use as an output for any other purpose. The two ports and their associated Data Direction Registers are addressed as follows:

Ports	Port Address	Data Direction Register Address
I/O Port 1	$0002	$0000
I/O Port 2	$0003	$0001

I/O Port 1 (P10-P17)

This is an 8-bit port whose individual bits may be defined as inputs or outputs by its data direction register. The 8 input buffers have 3-state capability allowing them to enter a high impedance state when the peripheral data lines are used as inputs. In order to be read properly, the voltage on the input lines must be greater than 2.0V for a logic 1 and less than .8V for a logic 0. As outputs, these lines are TTL compatible and may also be used as a source of up to 1mA at 1.15V to directly drive a Darlington base. After Reset, the I/O lines are configured as inputs.

I/O Port 2 (P20-P24)

This port has five lines whose individual bits may be defined as inputs or outputs by its associated data direction register. Bit "1" can be selected as an input or timer output. The 5 input buffers have three-state

capability allowing them to enter a high impedance state when used as inputs. In order to be read properly, the voltage on the input lines must be greater than 2.0V for a logic 1 and less than .8V for a logic 0. As outputs, this port has no internal pullup resistors but will drive TTL inputs directly. For driving CMOS inputs, an external pullup resistor (10K) is required. After Reset, the I/O lines are configured as inputs.

Port 2 also provides access to the Serial Communications Interface and the Timer. The Timer has two associated lines: Timer Input (P20) and Timer Output (P21). The Serial Communications Interface has three associated lines: Transmitter (P24), Receiver (P23) and Clock (P22). Both the Timer and SCI have associated control registers which allow for their selection and access on the I/O Port 2 lines.

SERIAL COMMUNICATIONS INTERFACE

The MC6803 contains a full-duplex asynchronous serial communications interface (SCI). Two serial data formats (standard mark/space (NRZ) or Bi-phase) are provided at several different data rates. The controller comprises a transmitter and a receiver which operate independently of each other but in the same data format and at the same data rate. Both transmitter and receiver communicate with the MPU via the data bus, and with the outside world via bits 2, 3, and 4 of Port 2.

Wake-Up Feature

In a typical multi-processor application, the software protocol will usually contain a destination address in the initial byte(s) of the message. In order to permit non-selected MPU's to ignore the remainder of the message, a wake-up feature is included whereby all further SCI Receive interrupts may be optionally inhibited until the data line goes idle. The "wake-up" bit is automatically reset by a string of ten consecutive 1's which indicates an idle data line. The software protocol must provide for the short idle period between consecutive messages and no idle period within messages.

Programmable Options

The following features of the MC6803 serial I/O section are programmable:

- format — standard mark/space (NRZ) or Bi-phase
- clock — external or internal
- baud rate — one of 4 per given MPU $\phi 2$ clock frequency, or external clock X8 input
- wake-up feature — enabled or disabled
- interrupt requests — enabled individually for transmitter and receiver data registers.
- clock output — internal clock enabled or disabled to Port 2 (bit 2)
- Port 2 (bits 3 and 4) — dedicated or not dedicated to serial I/O individually for transmitter and receiver.

Serial Communications Hardware

The serial communications hardware is controlled by 4 registers as shown in Figure 17. The registers include:

- an 8-bit control and status register
- a 4-bit write only rate and mode control register
- an 8-bit read-only receive data register and
- an 8-bit write-only transmit data register.
- In addition to the four registers, the serial I/O section utilizes bit 3 (serial input) and bit 4 (serial output) of Port 2. Bit 2 of Port 2 is utilized if either the internal-clock-out or external-clock-in options are selected.

ADDED INSTRUCTIONS

ABX	Adds the 8-bit unsigned accumulator B to the 16-bit X-Register taking into account the possible carry out of the low order byte of the X-Register.	IX ← IX + ACCB
ADDD	Adds the double precision ACCD* to the double precision value M:M + 1 and places the results in ACCD.	ACCD ← (ACCD) + (M:M + 1)
ASLD	Shifts all bits of ACCD one place to the left. Bit 0 is loaded with zero. The C bit is loaded from the most significant bit of ACCD.	ACCD ← 2x (ACCD)
LDD	Loads the contents of double precision memory location into the double accumulator A:B. The condition codes are set according to the data.	ACCD ← (M:M + 1)
LSRD	Shifts all bits of ACCD one place to the right. Bit 15 is loaded with zero. The C bit is loaded from least significant bit to ACCD.	ACCD ← (ACCD) ÷ 2
MUL	Multiplies the 8 bits in accumulator A with the 8 bits in accumulator B to obtain a 16-bit unsigned number in A:B. ACCA contains MSB of result.	ACCD ← ACCA * ACCB
PSHX	The contents of the index register is pushed onto the stack at the address contained in the stack pointer. The stack pointer is decremented by 2.	(IXL), SP ← (SP) − 1 (IXH), SP ← (SP) − 1
PULX	The index register is pulled from the stack beginning at the current address contained in the stack pointer + 1. The stack pointer is incremented by 2 in total.	SP ← (SP) + 1, IXH SP ← (SP) + 1, IXL
STD	Stores the contents of double accumulator A:B in memory. The contents of ACCD remain unchanged.	M:M + 1 ← (ACCD)
SUBD	Subtracts the contents of M:M + 1 from the contents of double accumulator AB and places the result in ACCD.	ACCAB ← (ACCD) − (M:M + 1)

*ACCD is the 16 bit register (A:B) formed by concatenating the A and B accumulators.

MOTOROLA MC6809 CPU

Data courtesy of Motorola Ltd. © Motorola.

The **6809** is a development of the **6800** series. Its instruction set is not completely compatable with the 6800, as the software design has been chosen to make the best of hardware features. The strengths of the chip are its addressing modes and its sixteen bit arithmetic functions, though it is an eight bit processor. It is comparable to the 8088 and Z80.

8-BIT MICROPROCESSING UNIT

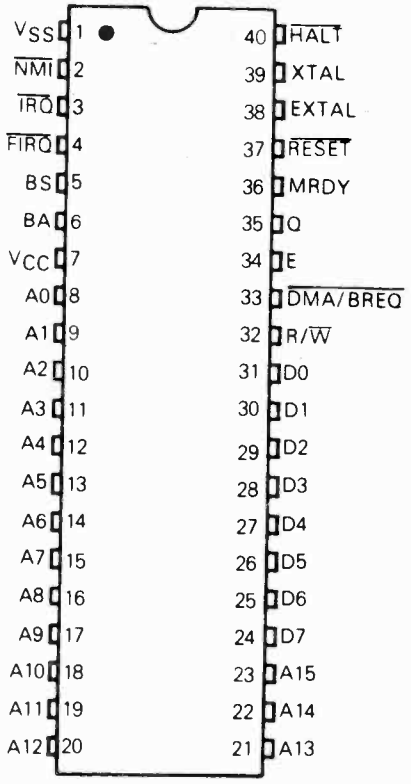
The MC6809 is a revolutionary high-performance 8-bit microprocessor which supports modern programming techniques such as position independence, reentrancy, and modular programming.

This third-generation addition to the M6800 Family has major architectural improvements which include additional registers, instructions, and addressing modes.

The basic instructions of any computer are greatly enhanced by the presence of powerful addressing modes. The MC6809 has the most complete set of addressing modes available on any 8-bit microprocessor today.

The MC6809 has hardware and software features which make it an ideal processor for higher level language execution or standard controller applications.

PIN ASSIGNMENTS



MC6800 COMPATIBLE

- Hardware – Interfaces with All M6800 Peripherals
- Software – Upward Source Code Compatible Instruction Set and Addressing Modes

ARCHITECTURAL FEATURES

- Two 16-Bit Index Registers
- Two 16-Bit Indexable Stack Pointers
- Two 8-Bit Accumulators can be Concatenated to Form One 16-Bit Accumulator
- Direct Page Register Allows Direct Addressing Throughout Memory

HARDWARE FEATURES

- On-Chip Oscillator (Crystal Frequency = 4 × E)
- $\overline{\text{DMA/BREQ}}$ Allows DMA Operation on Memory Refresh
- Fast Interrupt Request Input Stacks Only Condition Code Register and Program Counter
- MRDY Input Extends Data Access Times for Use with Slow Memory
- Interrupt Acknowledge Output Allows Vectoring by Devices
- Sync Acknowledge Output Allows for Synchronization to External Event
- Single Bus-Cycle $\overline{\text{RESET}}$
- Single 5-Volt Supply Operation
- $\overline{\text{NMI}}$ Inhibited After $\overline{\text{RESET}}$ Until After First Load of Stack Pointer
- Early Address Valid Allows Use with Slower Memories
- Early Write Data for Dynamic Memories

SOFTWARE FEATURES

- 10 Addressing Modes
  - 6800 Upward Compatible Addressing Modes
  - Direct Addressing Anywhere in Memory Map
  - Long Relative Branches
  - Program Counter Relative
  - True Indirect Addressing
  - Expanded Indexed Addressing:
    - 0-, 5-, 8-, or 16-Bit Constant Offsets
    - 8- or 16-Bit Accumulator Offsets
    - Auto Increment/Decrement by 1 or 2
- Improved Stack Manipulation
- 1464 Instructions with Unique Addressing Modes
- 8 × 8 Unsigned Multiply
- 16-Bit Arithmetic
- Transfer/Exchange All Registers
- Push/Pull Any Registers or Any Set of Registers
- Load Effective Address

FIGURE 2 — MC6809 EXPANDED BLOCK DIAGRAM

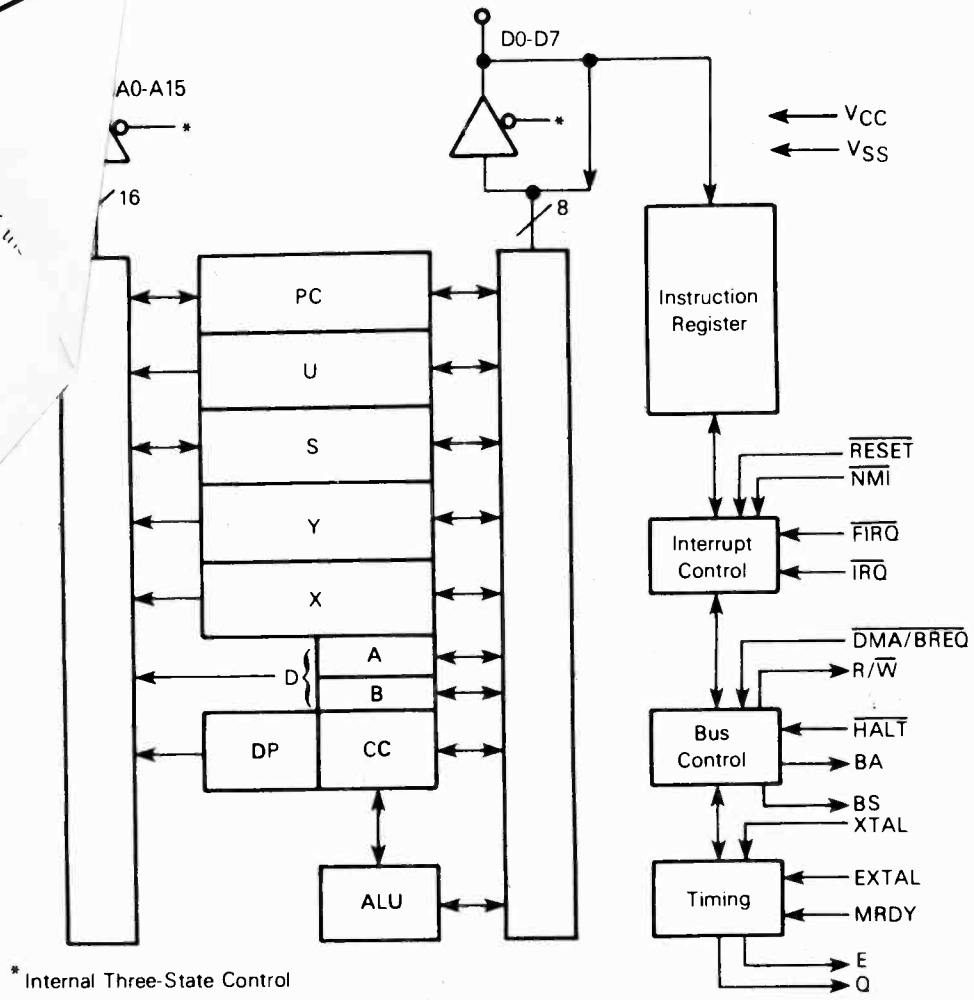
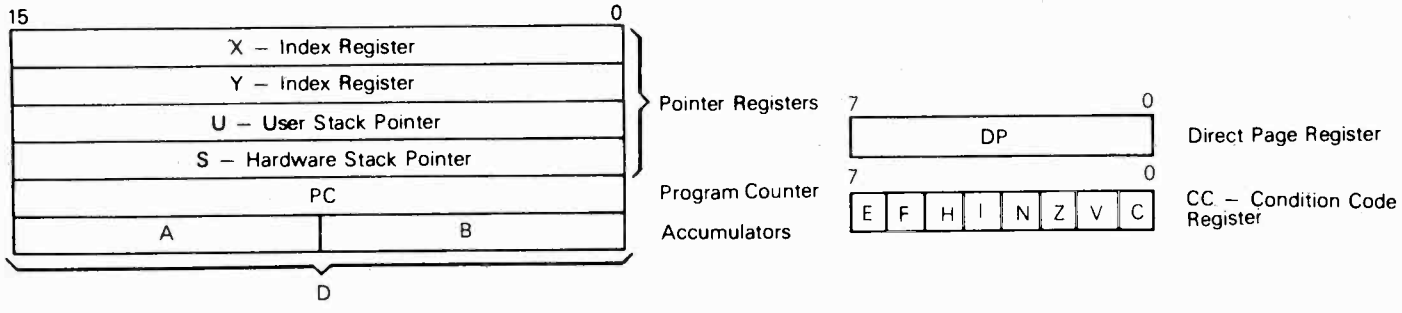


FIGURE 4 — PROGRAMMING MODEL OF THE MICROPROCESSING UNIT



PROGRAMMING MODEL

As shown in Figure 4, the MC6809 adds three registers to the set available in the MC6800. The added registers include a direct page register, the user stack pointer, and a second index register.

ACCUMULATORS (A, B, D)

The A and B registers are general purpose accumulators which are used for arithmetic calculations and manipulation of data.

Certain instructions concatenate the A and B registers to form a single 16-bit accumulator. This is referred to as the D register, and is formed with the A register as the most significant byte.

DIRECT PAGE REGISTER (DP)

The direct page register of the MC6809 serves to enhance the direct addressing mode. The content of this register appears at the higher address outputs (A8-A15) during direct addressing instruction execution. This allows the direct mode to be used at any place in memory, under program control. To ensure M6800 compatibility, all bits of this register are cleared during processor reset.

INDEX REGISTERS (X, Y)

The index registers are used in indexed mode of addressing. The 16-bit address in this register takes part in the calculation of effective addresses. This address may be used to point to data directly or may be modified by an optional constant or register offset. During some indexed modes, the contents of the index register are incremented or decremented to point to the next item of tabular type data. All four pointer registers (X, Y, U, S) may be used as index registers.

STACK POINTER (U,S)

The hardware stack pointer (S) is used automatically by the processor during subroutine calls and interrupts. The stack pointers of the MC6809 point to the top of the stack, in contrast to the MC6800 stack pointer, which pointed to the next free location on the stack. The user stack pointer (U) is controlled exclusively by the programmer. This allows arguments to be passed to and from subroutines with ease. Both stack pointers have the same indexed mode addressing capabilities as the X and Y registers, but also support **Push** and **Pull** instructions. This allows the MC6809 to be used efficiently as a stack processor, greatly enhancing its ability to support higher level languages and modular programming.

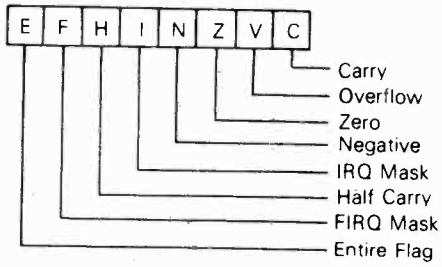
PROGRAM COUNTER

The program counter is used by the pro
the address of the next instruction to be exe
cessor. Relative addressing is provided allow
counter to be used like an index register in s

CONDITION CODE REGISTER

The condition code register defines the state
cessor at any given time. See Figure 5.

FIGURE 5 – CONDITION CODE REGISTER FORMAT



FAST-INTERRUPT REQUEST ($\overline{FIRQ}$)*

A low level on this input pin will initiate a fast interrupt sequence, provided its mask bit (F) in the CC is clear. This sequence has priority over the standard interrupt request ($\overline{IRQ}$), and is fast in the sense that it stacks only the contents of the condition code register and the program counter. The interrupt service routine should clear the source of the interrupt before doing an RTI.

ADDRESSING MODES

The basic instructions of any computer are greatly enhanced by the presence of powerful addressing modes. The MC6809 has the most complete set of addressing modes available on any microcomputer today. For example, the MC6809 has 59 basic instructions; however, it recognizes 1464 different variations of instructions and addressing modes. The addressing modes support modern programming techniques. The following addressing modes are available on the MC6809:

- Inherent (includes accumulator)
- Immediate
- Extended
  - Extended Indirect
- Direct
- Register
- Indexed
  - Zero-Offset
  - Constant Offset
  - Accumulator Offset
  - Auto Increment/Decrement
  - Indexed Indirect
- Relative
  - Short/Long Relative Branching
  - Program Counter Relative Addressing

INHERENT (INCLUDES ACCUMULATOR)

In this addressing mode, the opcode of the instruction contains all the address information necessary. Examples of inherent addressing are: ABX, DAA, SWI, ASRA, and CLRB.

IMMEDIATE ADDRESSING

In immediate addressing, the effective address of the data is the location immediately following the opcode (i.e., the data to be used in the instruction immediately following the opcode of the instruction). The MC6809 uses both 8- and 16-bit immediate values depending on the size of argument specified by the opcode. Examples of instructions with immediate addressing are:

```
LDA #$20
LDX #$F000
LDY #CAT
```

EXTENDED ADDRESSING

In extended addressing, the contents of the two bytes immediately following the opcode fully specify the 16-bit effective address used by the instruction.

Examples of extended addressing include:

```
LDA   CAT
STX   MOUSE
LDD   $2000
```

EXTENDED INDIRECT – As in the special case of indexed addressing (discussed below), one level of indirection may be added to extended addressing. In extended indirect, the two bytes following the postbyte of an indexed instruction contain the address of the data.

```
LDA   [CAT]
LDX   [$FFFE]
STU   [DOG]
```

DIRECT ADDRESSING

Direct addressing is similar to extended addressing except that only one byte of address follows the opcode. This byte specifies the lower eight bits of the address to be used. The upper eight bits of the address are supplied by the direct page register.

Of course, only 256 locations (one page) can be accessed without redefining the contents of the DP register.

Some examples of direct addressing are:

```
LDA   $30
SETDP $10 (assembler directive)
LDB   $1030
LDD   < CAT
```

REGISTER ADDRESSING

Some opcodes are followed by a byte that defines a register or set of registers to be used by the instruction. This is called a postbyte. Some examples of register addressing are:

TFR	X, Y	Transfers X into Y
EXG	A, B	Exchanges A with B
PSHS	A, B, X, Y	Push Y, X, B and A onto S
PULU	X, Y, D	Pull D, X, and Y from U

INDEXED ADDRESSING

In all indexed addressing, one of the pointer registers (X, Y, U, S, and sometimes PC) is used in a calculation of the effective address of the operand to be used by the instruction.

The postbyte of an indexed instruction specifies the basic type and variation of the addressing mode as well as the pointer register to be used.

ZERO-OFFSET INDEXED – In this mode, the selected pointer register contains the effective address of the data to be used by the instruction.

CONSTANT OFFSET INDEXED – In this mode, a twos-complement offset and the contents of one of the pointer registers are added to form the effective address of the operand. The pointer register's initial content is unchanged by the addition.

Three sizes of offsets are available:

5 bit (– 16 to + 15)
8 bit (– 128 to + 127)
16 bit (– 32768 to + 32767)

The twos complement 5-bit offset is included in the postbyte and, therefore, is most efficient in use of bytes and cycles.

ACCUMULATOR-OFFSET INDEXED – This mode is similar to constant offset indexed except that the twos-complement value in one of the accumulators (A, B, or D) and the contents of one of the pointer registers are added to form the effective address of the operand. The contents of both the accumulator and the pointer register are unchanged by the addition. The postbyte specifies which accumulator to use as an offset and no additional bytes are required.

The size of the increment/ decrement can be either one or two to allow for tables of either 8- or 16-bit data to be accessed and is selectable by the programmer.

AUTO INCREMENT/DECREMENT INDEXED – In the auto increment addressing mode, the pointer register contains the address of the operand. Then, after the pointer register is used it is incremented by one or two. This addressing mode is useful in stepping through tables, moving data, or for the creation of software stacks.

INDEXED INDIRECT – All of the indexing modes, with the exception of auto increment/decrement by one or a ± 4-bit offset, may have an additional level of indirection specified. In indirect addressing, the effective address is contained at the location specified by the contents of the index register plus any offset.

RELATIVE ADDRESSING

The byte(s) following the branch opcode is (are) treated as a signed offset which may be added to the program counter. If the branch condition is true, then the calculated address (PC + signed offset) is loaded into the program counter. Program execution continues at the new location as indicated by the PC; short (one byte offset) and long (two bytes offset) relative addressing modes are available. All of memory can be reached in long relative addressing as an effective address is interpreted modulo 2^{16}

PROGRAM COUNTER RELATIVE – The PC can be used as the pointer register with 8- or 16-bit signed offsets. As in relative addressing, the offset is added to the current PC to create the effective address. The effective address is then used as the address of the operand or data. Program counter relative addressing is used for writing position independent programs. Tables related to a particular routine will maintain the same relationship after the routine is moved, if referenced relative to the program counter.

INSTRUCTION SET

The instruction set of the MC6809E is similar to that of the MC6800 and is upward compatible at the source code level. The number of opcodes has been reduced from 72 to 59, but because of the expanded architecture and additional addressing modes, the number of available opcodes (with different addressing modes) has risen from 197 to 1464.

Some of the new instructions are described in detail below.

LEAX/LEAY/LEAU/LEAS

The LEA (load effective address) works by calculating the effective address used in an indexed instruction and stores that address value, rather than the data at that address, in a pointer register. This makes all the features of the internal addressing hardware available to the programmer.

MUL

Multiplies the unsigned binary numbers in the A and B accumulator and places the unsigned result into the 16-bit D accumulator. The unsigned multiply also allows multiple-precision multiplications.

LONG AND SHORT RELATIVE BRANCHES

The MC6809 has the capability of program counter relative branching throughout the entire memory map. In this mode, if the branch is to be taken, the 8- or 16-bit signed offset is added to the value of the program counter to be used as the effective address. This allows the program to branch anywhere in the 64K memory map. Position-independent code can be easily generated through the use of relative branching.

16-BIT OPERATION

The MC6809 has the capability of processing 16-bit data. These instructions include loads, stores, compares, adds, subtracts, transfers, exchanges, pushes, and pulls.

TABLE 4 — 8-BIT ACCUMULATOR AND MEMORY INSTRUCTIONS

Mnemonic(s)	Operation
ADCA, ADCB	Add memory to accumulator with carry
ADDA, ADDB	Add memory to accumulator
ANDA, ANDB	And memory with accumulator
ASL, ASLA, ASLB	Arithmetic shift of accumulator or memory left
ASR, ASRA, ASRB	Arithmetic shift of accumulator or memory right
BITA, BITB	Bit test memory with accumulator
CLR, CLRA, CLRB	Clear accumulator or memory location
CMPA, CMPB	Compare memory from accumulator
COM, COMA, COMB	Complement accumulator or memory location
DAA	Decimal adjust A accumulator
DEC, DECA, DECB	Decrement accumulator or memory location
EORA, EORB	Exclusive or memory with accumulator
EXG R1, R2	Exchange R1 with R2 (R1, R2 = A, B, CC, DP)
INC, INCA, INCB	Increment accumulator or memory location
LDA, LDB	Load accumulator from memory
LSL, LSLA, LSLB	Logical shift left accumulator or memory location
LSR, LSRA, LSRB	Logical shift right accumulator or memory location
MUL	Unsigned multiply (A × B → D)
NEG, NEGA, NEGB	Negate accumulator or memory
ORA, ORB	Or memory with accumulator
ROL, ROLA, ROLB	Rotate accumulator or memory left
ROR, RORA, RORB	Rotate accumulator or memory right
SBCA, SBCB	Subtract memory from accumulator with borrow
STA, STB	Store accumulator to memory
SUBA, SUBB	Subtract memory from accumulator
TST, TSTA, TSTB	Test accumulator or memory location
TFR R1, R2	Transfer R1 to R2 (R1, R2 = A, B, CC, DP)

NOTE: A, B, CC, or DP may be pushed to (pulled from) stack with either PSHS, PSHU (PULS, PULU) instructions.

TABLE 5 — 16-BIT ACCUMULATOR AND MEMORY INSTRUCTIONS

Mnemonic(s)	Operation
ADDD	Add memory to D accumulator
CMPD	Compare memory from D accumulator
EXG D, R	Exchange D with X, Y, S, U, or PC
LDD	Load D accumulator from memory
SEX	Sign Extend B accumulator into A accumulator
STD	Store D accumulator to memory
SUBD	Subtract memory from D accumulator
TFR D, R	Transfer D to X, Y, S, U, or PC
TFR R, D	Transfer X, Y, S, U, or PC to D

NOTE: D may be pushed (pulled) to stack with either PSHS, PSHU (PULS, PULU) instructions.

TABLE 6 — INDEX REGISTER/STACK POINTER INSTRUCTIONS

Instruction	Description
CMPS, CMPU	Compare memory from stack pointer
CMPX, CMPY	Compare memory from index register
EXG R1, R2	Exchange D, X, Y, X, U, or PC with D, X Y, S, U, or PC
LEAS, LEAU	Load effective address into stack pointer
LEAX, LEAY	Load effective address into index register
LDS, LDU	Load stack pointer from memory
LDX, LDY	Load index register from memory
PSHS	Push A, B, CC, DP, D, X, Y, U, or PC onto hardware stack
PSHU	Push A, B, CC, DP, D, X, Y, S, or PC onto user stack
PULS	Pull A, B, CC, DP, D, X, Y, U, or PC from hardware stack
PULU	Pull A, B, CC, DP, D, X, Y, S, or PC from hardware stack
STS, STU	Store stack pointer to memory
STX, STY	Store index register to memory
TFR R1, R2	Transfer D, X, Y, S, U or PC to D, X, Y, S, U, or PC
ABX	Add B accumulator to X (unsigned)

TABLE 7 — BRANCH INSTRUCTIONS

Instruction	Description
SIMPLE BRANCHES	
BEQ, LBEQ	Branch if equal
BNE, LBNE	Branch if not equal
BMI, LBMI	Branch if minus
BPL, LBPL	Branch if plus
BCS, LBCS	Branch if carry set
BCC, LBCC	Branch if carry clear
BVS, LBVS	Branch if overflow set
BVC LBVC	Branch if overflow clear
SIGNED BRANCHES	
BGT, LBGT	Branch if greater (signed)
BVS, LBVS	Branch if invalid 2s complement result
BGE, LBGE	Branch if greater than or equal (signed)
BEQ, LBEQ	Branch if equal
BNE, LBNE	Branch if not equal
BLE, LBLE	Branch if less than or equal (signed)
BVC, LBVC	Branch if valid 2s complement result
BLT, LBLT	Branch if less than (signed)
UNSIGNED BRANCHES	
BHI, LBHI	Branch if higher (unsigned)
BCC, LBCC	Branch if higher or same (unsigned)
BHS, LBHS	Branch if higher or same (unsigned)
BEQ, LBEQ	Branch if equal
BNE, LBNE	Branch if not equal
BLS, LBLS	Branch if lower or same (unsigned)
BCS, LBCS	Branch if lower (unsigned)
BLO, LBLO	Branch if lower (unsigned)
OTHER BRANCHES	
BSR, LBSR	branch to subroutine
BRA, LBRA	Branch always
BRN, LBRN	Branch never

TABLE 8 — MISCELLANEOUS INSTRUCTIONS

Instruction	Description
ANDCC	AND condition code register
CWAI	AND condition code register, then wait for interrupt
NOP	No operation
ORCC	OR condition code register
JMP	Jump
JSR	Jump to subroutine
RTI	Return from interrupt
RTS	Return from subroutine
SWI, SWI2, SWI3	Software interrupt (absolute indirect)
SYNC	Synchronize with interrupt line

FAIRCHILD F6810 128 × 8 STATIC RAM

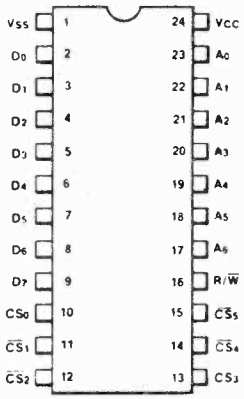
Description

The F6810 128 x 8-bit static RAM is a byte-organized memory designed for use in bus-organized systems. Fabricated with n-channel, silicon-gate technology, the device is available in three frequency ranges: 1.0MHz (F6810), 1.5 MHz (F68A 10), 2.0 MHz (F68B 10). The device, which operates from a single power supply, is compatible with TTL and DTL; it needs no clocks or refreshing because of its static operation.

The memory is compatible with the F6800 microcomputer family, providing random storage in byte increments. Memory expansion is provided through multiple chip select inputs.

- **Organized as 128 Bytes of 8 Bits**
- **Static Operation**
- **Bidirectional 3-State Data Input/Output**
- **Six Chip Select Inputs (Four Active LOW, Two Active HIGH)**

Connection Diagram 24-Pin DIP



MOTOROLA MC6821 PIA

Data courtesy of Motorola Ltd. © Motorola.

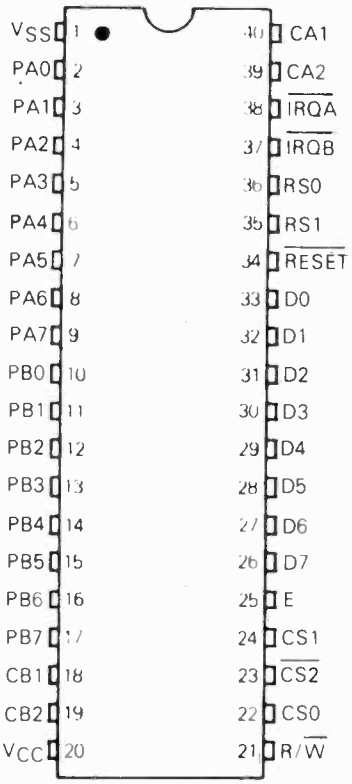
PERIPHERAL INTERFACE ADAPTER (PIA)

The MC6821 Peripheral Interface Adapter provides the universal means of interfacing peripheral equipment to the M6800 family of microprocessors. This device is capable of interfacing the MPU to peripherals through two 8-bit bidirectional peripheral data buses and four control lines. No external logic is required for interfacing to most peripheral devices.

The functional configuration of the PIA is programmed by the MPU during system initialization. Each of the peripheral data lines can be programmed to act as an input or output, and each of the four control/interrupt lines may be programmed for one of several control modes. This allows a high degree of flexibility in the overall operation of the interface.

- 8-Bit Bidirectional Data Bus for Communication with the MPU
- Two Bidirectional 8-Bit Buses for Interface to Peripherals
- Two Programmable Control Registers
- Two Programmable Data Direction Registers
- Four Individually-Controlled Interrupt Input Lines; Two Usable as Peripheral Control Outputs
- Handshake Control Logic for Input and Output Peripheral Operation
- High-Impedance Three-State and Direct Transistor Drive Peripheral Lines
- Program Controlled Interrupt and Interrupt Disable Capability
- CMOS Drive Capability on Side A Peripheral Lines
- Two TTL Drive Capability on All A and B Side Buffers
- TTL-Compatible
- Static Operation

PIN ASSIGNMENT



PIA INTERFACE SIGNALS FOR MPU

The PIA interfaces to the M6800 bus with an 8-bit bidirectional data bus, three chip select lines, two register select lines, two interrupt request lines, a read/write line, an enable line and a reset line. To ensure proper operation with the MC6800, MC6802, or MC6808 microprocessors, VMA should be used as an active part of the address decoding.

Bidirectional Data (D0-D7) – The bidirectional data lines (D0-D7) allow the transfer of data between the MPU and the PIA. The data bus output drivers are three-state devices that remain in the high-impedance (off) state except when the MPU performs a PIA read operation. The read/write line is in the read (high) state when the PIA is selected for a read operation.

Enable (E) – The enable pulse, E, is the only timing signal that is supplied to the PIA. Timing of all other signals is referenced to the leading and trailing edges of the E pulse.

Read/Write (R/$\overline{W}$) – This signal is generated by the MPU to control the direction of data transfers on the data bus. A low state on the PIA read/write line enables the input buffers and data is transferred from the MPU to the PIA on the E signal if the device has been selected. A high on the read/write line sets up the PIA for a transfer of data to the bus. The PIA output buffers are enabled when the proper address and the enable pulse E are present.

$\overline{RESET}$ – The active low $\overline{RESET}$ line is used to reset all register bits in the PIA to a logical zero (low). This line can be used as a power-on reset and as a master reset during system operation.

Chip Selects (CS0, CS1, and $\overline{CS2}$) – These three input signals are used to select the PIA. CS0 and CS1 must be high and $\overline{CS2}$ must be low for selection of the device.

Register Selects (RS0 and RS1) – The two register select lines are used to select the various registers inside the PIA. These two lines are used in conjunction with internal Control Registers to select a particular register that is to be written or read.

Interrupt Request ($\overline{IRQA}$ and $\overline{IRQB}$) – The active low Interrupt Request lines ($\overline{IRQA}$ and $\overline{IRQB}$) act to interrupt the MPU either directly or through interrupt priority circuitry. These lines are "open drain" (no load device on the chip). This permits all interrupt request lines to be tied together in a wire-OR configuration.

Each Interrupt Request line has two internal interrupt flag bits that can cause the Interrupt Request line to go low. Each flag bit is associated with a particular peripheral interrupt line. Also, four interrupt enable bits are provided in the PIA which may be used to inhibit a particular interrupt from a peripheral device.

Servicing an interrupt by the MPU may be accomplished by a software routine that, on a prioritized basis, sequentially reads and tests the two control registers in each PIA for interrupt flag bits that are set.

PIA PERIPHERAL INTERFACE LINES

The PIA provides two 8-bit bidirectional data buses and four interrupt/control lines for interfacing to peripheral devices.

Section A Peripheral Data (PA0-PA7) – Each of the peripheral data lines can be programmed to act as an input or output. This is accomplished by setting a "1" in the corresponding Data Direction Register bit for those lines which are to be outputs. A "0" in a bit of the Data Direction Register causes the corresponding peripheral data line to act as an input. During an MPU Read Peripheral Data Operation, the data on peripheral lines programmed to act as inputs appears directly on the corresponding MPU Data Bus lines. In the input mode, the internal pullup resistor on these lines represents a maximum of 1.5 standard TTL loads.

Section B Peripheral Data (PB0-PB7) – The peripheral data lines in the B Section of the PIA can be programmed to act as either inputs or outputs in a similar manner to PA0-PA7. They have three-state capability, allowing them to enter a high-impedance state when the peripheral data line is used as an input. In addition, data on the peripheral data lines PB0-PB7 will be read properly from those lines programmed as outputs even if the voltages are below 2.0 volts for a "high" or above 0.8 V for a "low". As outputs, these lines are compatible with standard TTL and may also be used as a source of up to 1 milliampere at 1.5 volts to directly drive the base of a transistor switch.

Interrupt Input (CA1 and CB1) – Peripheral input lines CA1 and CB1 are input only lines that set the interrupt flags of the control registers. The active transition for these signals is also programmed by the two control registers.

Peripheral Control (CA2) – The peripheral control line CA2 can be programmed to act as an interrupt input or as a peripheral control output. As an output, this line is compatible with standard TTL; as an input the internal pullup resistor on this line represents 1.5 standard TTL loads. The function of this signal line is programmed with Control Register A.

Peripheral Control (CB2) – Peripheral Control line CB2 may also be programmed to act as an interrupt input or peripheral control output. As an input, this line has high input impedance and is compatible with standard TTL. As an output it is compatible with standard TTL and may also be used as a source of up to 1 milliampere at 1.5 volts to directly drive the base of a transistor switch. This line is programmed by Control Register B.

MOTOROLA MC6840 PTM

PROGRAMMABLE TIMER MODULE (PTM)

The MC6840 is a programmable subsystem component of the M6800 family designed to provide variable system time intervals.

The MC6840 has three 16-bit binary counters, three corresponding control registers, and a status register. These counters are under software control and may be used to cause system interrupts and/or generate output signals. The MC6840 may be utilized for such tasks as frequency measurements, event counting, interval measuring, and similar tasks. The device may be used for square wave generation, gated delay signals, single pulses of controlled duration, and pulse width modulation as well as system interrupts.

- Operates from a Single 5 Volt Power Supply
- Fully TTL Compatible
- Single System Clock Required (Enable)
- Selectable Prescaler on Timer 3 Capable of 4 MHz for the MC6840, 6 MHz for the MC68A40 and 8 MHz for the MC68B40
- Programmable Interrupts ($\overline{IRQ}$) Output to MPU
- Readable Down Counter Indicates Counts to Go Until Time-Out
- Selectable Gating for Frequency or Pulse-Width Comparison
- $\overline{RESET}$ Input
- Three Asynchronous External Clock and Gate/Trigger Inputs Internally Synchronized
- Three Maskable Outputs

FIGURE 1 — PIN ASSIGNMENT

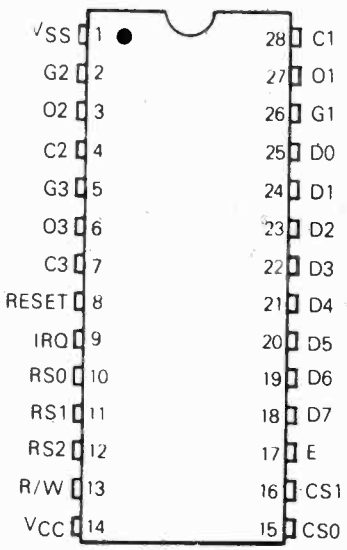
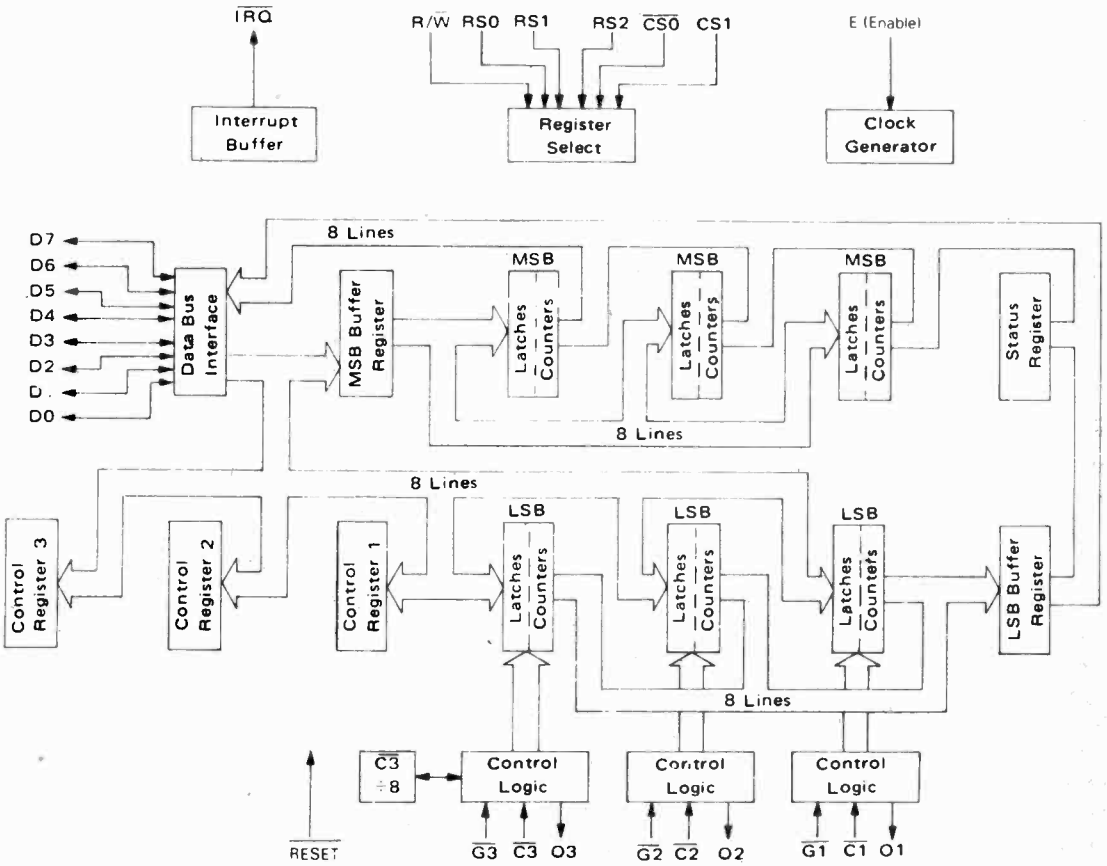


FIGURE 2 — BLOCK DIAGRAM



DEVICE OPERATION

The MC6840 is part of the M6800 microprocessor family and is fully bus compatible with M6800 systems. The three timers in the MC6840 operate independently and in several distinct modes to fit a wide variety of measurement and synthesis applications.

The MC6840 is an integrated set of three distinct counter/timers (Figure 1). It consists of three 16-bit data latches, three 16-bit counters (clocked independently), and the comparison and enable circuitry necessary to implement various measurement and synthesis functions. In addition, it contains interrupt drivers to alert the processor that a particular function has been completed.

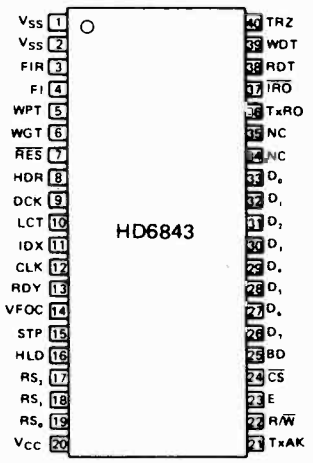
In a typical application, a timer will be loaded by first storing two bytes of data into an associated Counter Latch. This data is then transferred into the counter via a Counter Initialization cycle. If the counter is enabled, the counter decrements on each subsequent clock period which may be an external clock, or Enable (E) until one of several predetermined conditions causes it to halt or recycle. The timers are thus programmable, cyclic in nature, controllable by external inputs or the MPU program, and accessible by the MPU at any time.

Data courtesy of Motorola Ltd. © Motorola.

HITACHI HD6843 FDC

Data courtesy of Hitachi Electronic Components (UK) Ltd.

■ PIN ARRANGEMENT



(Top View)

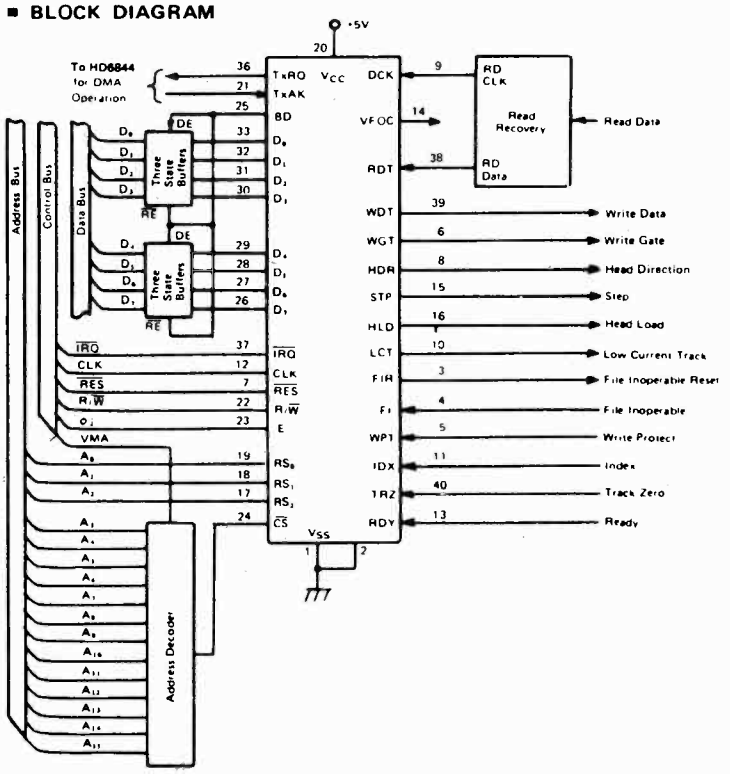
■ FEATURES

- Format compatible with IBM3740
- User Programmable read/write format
- Ten powerful macro-commands
- Macro End Interrupt allows parallel processing of MPU and FDC
- Controls multiple Floppies with external multiplexing
- Direct interface with HMCS6800
- Programmable seek and settling times enable operation with a wide range of Floppy drives
- Offers both Programmed Controlled I/O (PCIO) and DMA data transfer mode
- Free-Format read or write
- Single 5-volt power supply
- All registers directly accessible
- Compatible with MC6843

The HD6843 Floppy Disk Controller performs the complex MPU/Floppy interface function. The FDC was designed to optimize the balance between the "Hardware/Software" in order to achieve integration of all key functions and maintain flexibility.

The FDC can interface a wide range of drives with a minimum of external hardware. Multiple drives can be controlled with the addition of external multiplexing rather than additional FDC's.

■ BLOCK DIAGRAM



FAIRCHILD F6844 DMAC

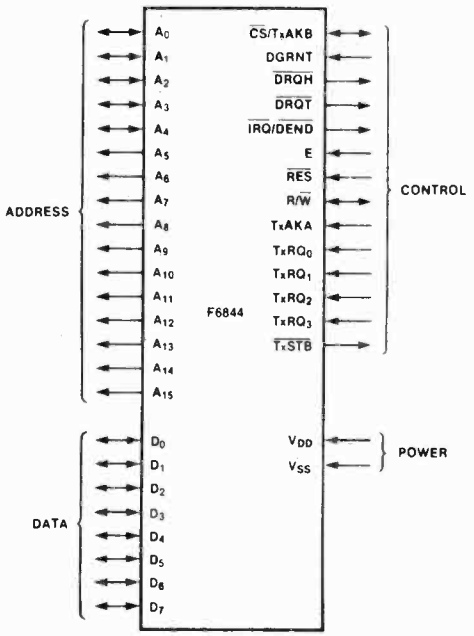
The F6844 Direct Memory Access Controller (DMAC) transfers data directly between memory and peripheral device controllers. In bus-organized systems, such as those based on the F6800 microprocessor, the DMAC, rather than the MPU, controls the address and data buses.

The DMAC bus interface includes select, read/write, interrupt, transfer request/grant, and bus interface logic to permit data transfer over an 8-bit bidirectional data bus. The F6844 functional configuration is programmed through the data bus. The internal structure provides for control and handling of four individual channels, each of which is separately configured. Programmable control registers provide control for the transfer location and length, individual channel control and transfer mode configuration, priority of servicing, data chaining, and interrupt control. Status and control lines serve the peripheral controllers.

The mode of transfer for each channel can be programmed as cycle-stealing or burst transfer.

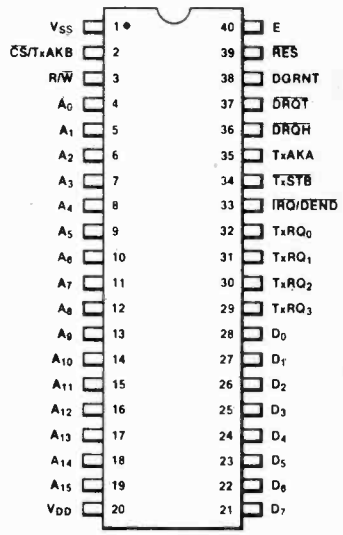
Typical applications include use with the F6856 Synchronous Protocol Communications Controller, the F6854 Advanced Data Link Controller, and the F68488 IEEE-488 Bus Controller.

- **Four DMA Channels, Each Having a 16-Bit Address Register and a 16-Bit Byte Count Register**
- **2M Byte/Sec Maximum Data Transfer Rate**
- **Selection of Fixed or Rotating Priority Service Control**
- **Separate Control Bits for Each Channel**
- **Data Chain Function**
- **Address Increment or Decrement Update**
- **Programmable Interrupts and DMA End to Peripheral Controllers**



F6844 Signal Functions

Connection Diagram
40-Pin DIP



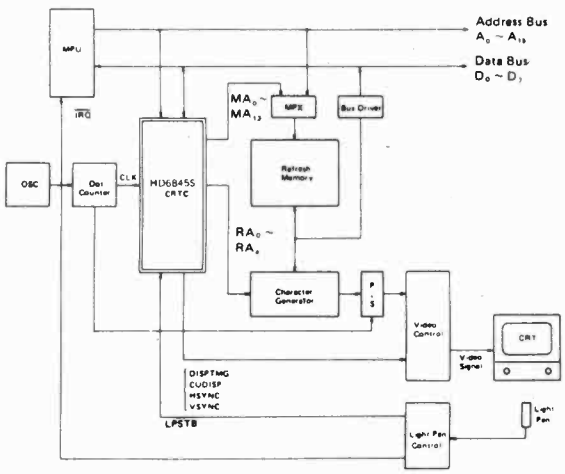
Data courtesy of Fairchild Semiconductors Ltd.

HITACHI HD6845 CRTC

Data courtesy of Hitachi Electronic Components (UK) Ltd.

The CRTC is a LSI controller which is designed to provide an interface for microcomputers to raster scan type CRT displays. The CRTC belongs to the HMCS6800 LSI Family and has full compatibility with MPU in both data lines and control lines. Its primary function is to generate timing signal which is necessary for raster scan type CRT display according to the specification programmed by MPU.

■ **SYSTEM BLOCK DIAGRAM**



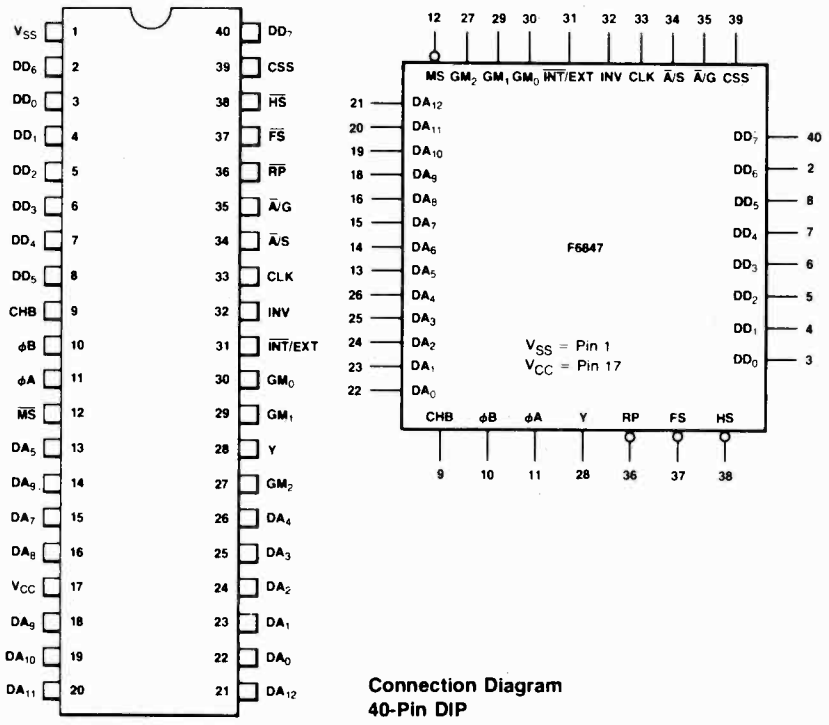
■ **FEATURES**

- Number of Displayed Characters on the Screen, Vertical Dot Format of One Character, Horizontal and Vertical Sync Signal, Display Timing Signal are Programmable
- 3.7 MHz High Speed Display Operation
- Line Buffer-less Refreshing
- 14-bit Refresh Memory Address Output (16k Words max. Access)
- Programmable Interlace/Non-interlace Scan Mode
- Programmable Cursor Height and its Blink
- Built-in Light Pen Detection Function
- Paging and Scrolling Capability

FAIRCHILD F6847 VDG

Data courtesy of Fairchild Semiconductors Ltd.

Logic Symbol



Connection Diagram
40-Pin DIP

The Fairchild F6847 Video Display Generator (VDG) provides a means of interfacing the Fairchild F6800 microprocessor family (or similar products) to a commercially available color or black-and-white television receiver. Applications of the VDG include video games, bioengineering displays, education, communications, and any instance in which graphics are required.

The VDG reads data from memory and produces a composite video signal that allows the generation of alphanumeric or graphic displays. The generated composite video may be up-modulated to either channel 3 or 4 by using a suitable rf modulator. The up-modulated signal is suitable for application to the antenna of a color TV. *Figure 1* illustrates a typical TV game application.

- **Generates Four Different Alphanumeric Display Modes and Eight Graphic Display Modes**
- **The Alphanumeric Modes Display 32 Characters per Line by 16 Lines.**
- **An Internal Multiplexer Allows the Use of Either the Internal ROM or an External Character Generator.**
- **One Display Mode Offers 8-Color 64 x 32 Density Graphics in an Alphanumeric Display Mode.**
- **One Display Mode Offers 4-Color 64 x 48 Density Graphics in an Alphanumeric Display Mode.**
- **All Alphanumeric Modes Have a Selectable Video Inverse.**
- **Generates Full Video Signal**
- **Generates R-Y and B-Y Signals for External Color Modulator**
- **Full Graphic Modes Offer 64 x 64, 128 x 64, 128 x 96, 128 x 192, or 256 x 192 Densities.**
- **Full Graphic Modes Allow 2-Color or 4-Color Data Structures.**
- **Full Graphic Modes Use One of Two 4-Color Sets or One of Two 2-Color Sets.**

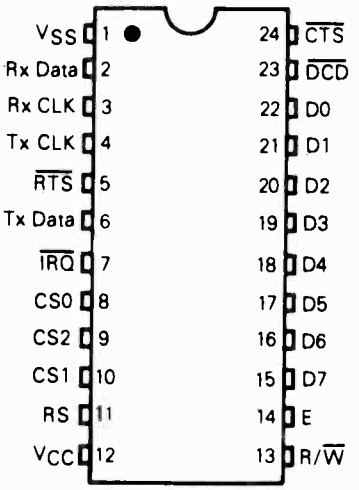
MOTOROLA MC6850 ACIA

Data courtesy of Motorola Ltd. © Motorola.

The MC6850 Asynchronous Communications Interface Adapter provides the data formatting and control to interface serial asynchronous data communications information to bus organized systems such as the MC6800 Microprocessing Unit.

The bus interface of the MC6850 includes select, enable, read/write, interrupt and bus interface logic to allow data transfer over an 8-bit bidirectional data bus. The parallel data of the bus system is serially transmitted and received by the asynchronous data interface, with proper formatting and error checking. The functional configuration of the ACIA is programmed via the data bus during system initialization. A programmable Control Register provides variable word lengths, clock division ratios, transmit control, receive control, and interrupt control. For peripheral or modem operation, three control lines are provided. These lines allow the ACIA to interface directly with the MC6860L 0-600 bps digital modem.

PIN ASSIGNMENT

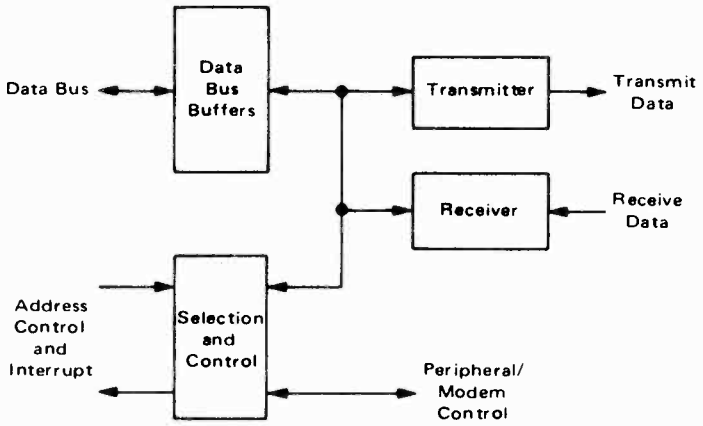


- 8- and 9-Bit Transmission
- Optional Even and Odd Parity
- Parity, Overrun and Framing Error Checking
- Programmable Control Register
- Optional ÷ 1, ÷ 16, and ÷ 64 Clock Modes
- Up to 1.0 Mbps Transmission
- False Start Bit Deletion
- Peripheral/Modem Control Functions
- Double Buffered
- One- or Two-Stop Bit Operation

TRANSMIT

A typical transmitting sequence consists of reading the ACIA Status Register either as a result of an interrupt or in the ACIA's turn in a polling sequence. A character may be written into the Transmit Data Register if the status read operation has indicated that the Transmit Data Register is empty. This character is transferred to a Shift Register where it is serialized and transmitted from the Transmit Data output preceded by a start bit and followed by one or two stop bits. Internal parity (odd or even) can be optionally added to the character and will occur between the last data bit and the first stop bit. After the first character is written in the Data Register, the Status Register can be read again to check for a Transmit Data Register Empty condition and current peripheral status. If the register is empty, another character can be loaded for transmission even though the first character is in the process of being transmitted (because of double buffering). The second character will be automatically transferred into the Shift Register when the first character transmission is completed. This sequence continues until all the characters have been transmitted.

MC6850 ASYNCHRONOUS COMMUNICATIONS INTERFACE ADAPTER BLOCK DIAGRAM



RECEIVE

Data is received from a peripheral by means of the Receive Data input. A divide-by-one clock ratio is provided for an externally synchronized clock (to its data) while the divide-by-16 and 64 ratios are provided for internal synchronization. Bit synchronization in the divide-by-16 and 64 modes is initiated by the detection of 8 or 32 low samples on the receive line in the divide-by-16 and 64 modes respectively. False start bit deletion capability insures that a full half bit of a start bit has been received before the internal clock is synchronized to the bit time. As a character is being received, parity (odd or even) will be checked and the error indication will be available in the Status Register along with framing error, overrun error, and Receive Data Register full. In a typical receiving sequence, the Status Register is read to determine if a character has been received from a peripheral. If the Receiver Data Register is full, the character is placed on the 8-bit ACIA bus when a Read Data command is received from the MPU. When parity has been selected for a 7-bit word (7 bits plus parity), the receiver strips the parity bit (D7 = 0) so that data alone is transferred to the MPU. This feature reduces MPU programming. The Status Register can continue to be read to determine when another character is available in the Receive Data Register. The receiver is also double buffered so that a character can be read from the data register as another character is being received in the shift register. The above sequence continues until all characters have been received.

HITACHI HD6852 SSDA

Data courtesy of Hitachi Electronic Components (UK) Ltd.

The HD6852 Synchronous Serial Data Adapter provides a bi-directional serial interface for synchronous data information interchange. It contains interface logic for simultaneously transmitting and receiving standard synchronous communications characters in bus organized systems such as the HMCS6800 Microprocessor systems.

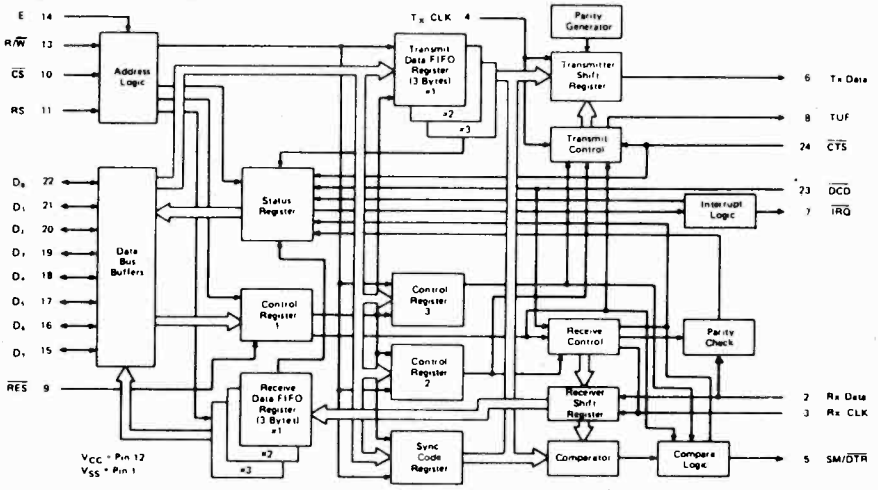
The bus interface of the HD6852 includes select, enable, read/write, interrupt, and bus interface logic to allow data transfer over an 8-bit bi-directional data bus. The parallel data of the bus system is serially transmitted and received by the synchronous data interface with synchronization, fill character insertion/deletion, and error checking. The functional configuration of the SSDA is programmed via the data bus during system initialization.

Programmable control registers provide control for variable word length, transmit control, receive control, synchronization control and interrupt control. Status, timing and control lines provide peripheral or modem control.

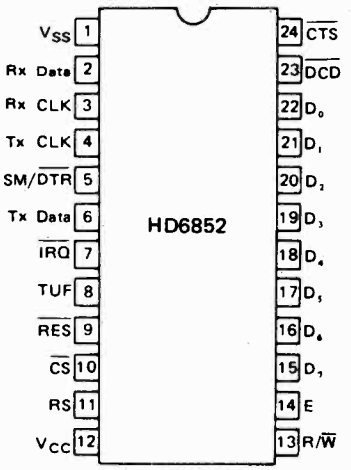
Typical applications include data communications terminals, floppy disk controllers, cassette or cartridge tape controllers and numerical control systems.

- **FEATURES**
- Programmable Interrupts from Transmitter, Receiver, and Error Detection Logic
- Character Synchronization on One or Two Sync Codes
- External Synchronization Available for Parallel-Serial Operation
- Programmable Sync Code Register
- Up to 600kbps Transmitter
- Peripheral/Modem Control Functions
- Three Bytes of FIFO Buffering on Both Transmit and Receive
- 6, 7, or 8 Bit Data Transmission
- Optional Even and Odd Parity
- Parity, Overrun, and Underflow Status
- Compatible with MC6852 and MC68A52

■ BLOCK DIAGRAM



■ PIN ARRANGEMENT



HITACHI HD46508 Analogue Interface

Data courtesy of Hitachi Electronic Components (UK) Ltd.

The HD46508 is a monolithic NMOS device with a 10-bit analog-to-digital converter, a programmable voltage comparator, a 16-channel analog multiplexer and HMCS6800 microprocessor family compatible interface.

Each of 16 analog inputs is either converted to a digital data by the analog-to-digital converter or compared with the specified value by the programmable comparator. The analog-to-digital converter uses successive approximation method as the conversion technique. It's intrinsic resolution is 10 bits but it can be 8 bits if the programmer so desires. The programmable voltage comparator compares the input voltage with the value specified by the programmer. The result (greater than, or smaller than) is reflected to the flag in the status register.

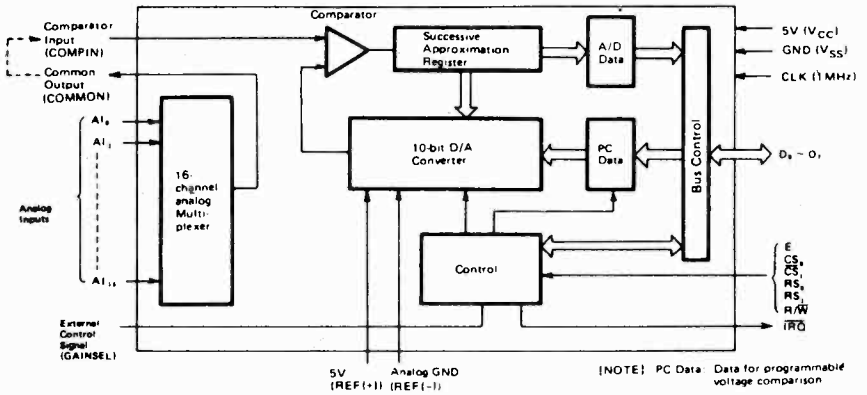
The device can expand its capability by controlling the external circuits such as sample holder, pre-amplifier and external multiplexer.

With these features, this device is ideally suited to applications such as process control, machine control and vehicle control.

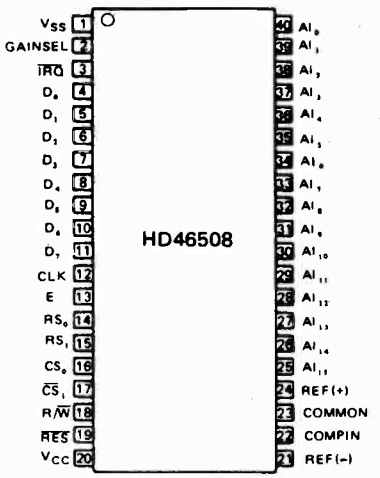
■ FEATURES

- 16-channel Analog multiplexer
- Programmable A/D Converter resolution (10-bit or 8-bit)
- Programmable Voltage comparison (PC)
- Conversion Time 100μs (A/D), 13μs(PC)
- External Sample and Hold Circuit Control
- Auto Range-switching Control of External Amplifier
- Waiting Function for the Settling Time of External Amplifier
- Interrupt Control (Only for A/D conversion)
- Single +5V Power Supply
- Compatible with HMCS6800 Bus (The connection with other Asynchronous Buses possible)

■ BLOCK DIAGRAM



■ PIN ARRANGEMENT



HITACHI HD146818 Real Time Clock

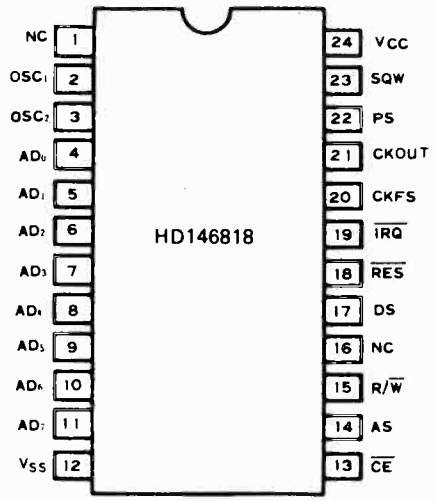
The HD146818 is a HMCS6800 peripheral CMOS device which combines three unique features: a complete time-of-day clock with alarm and one hundred calendar, a programmable periodic interrupt and square-wave generator, and 50 bytes of Low-power static RAM.

The Real-Time Clock plus RAM has two distinct uses. First, it is designed as battery powered CMOS part including all the common battery backed-up functions such as RAM, time, and calender. Secondly, the HD146818 may be used with a CMOS microprocessor to relieve the software of timekeeping work-load and to extend the available RAM of an MPU such as the HD6301.

■ **FEATURES**

- Time-of-Day Clock and Calendar
  - Counts Seconds, Minutes, and Hours of the Day
  - Counts Days of Week, Date, Month, and Year
- Binary or BCD Representation of Time, Calendar, and Alarm
- 12- or 24 Hour Clock with AM and PM in 12-Hour Mode
- Automatic End of Month Recognition
- Automatic Leap Year Compensation
- Interfaced with Software as 64 RAM Locations
  - 14 Bytes of Clock and Control Register
  - 50 Bytes of General Purpose RAM
- Three Interrupt are Separately Software Maskable and Testable
  - Time-of-Day Alarm, Once-per-Second to Once-per-Day
  - Periodic Rates from 30.5μs to 500ms
  - End-of-Clock Update Cycle
- Multiplexed Bus Interface Circuit of HD6801, HD6301 and 8085

■ **PIN ARRANGEMENT**



- Programmable Square-Wave Output Signal
- Three Time Base Input Options
  - 4.194304 MHz
  - 1.048576 MHz
  - 32.768 kHz
- Clock Output May be used as Microprocessor Clock Input
  - At Time Base Frequency ÷4 or ÷1

Data courtesy of Hitachi Electronic Components (UK) Ltd.

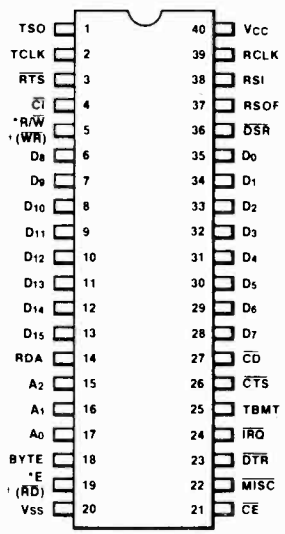
FAIRCHILD F3846/F6856 SPCC

Data courtesy of Fairchild Semiconductors Ltd.

Description

The F3846/F6856 Synchronous Protocol Communications Controller (SPCC) is a monolithic n-channel MOS LSI circuit designed to satisfy the major interface requirements of the bit-oriented protocol (BOP) and byte-control protocol (BCP). The SPCC converts parallel data from the CPU into a continuous serial data stream for transmission. Simultaneously, it converts received serial data to parallel data for the CPU. The SPCC is organized to interface with either an 8- or 16-bit bidirectional data bus, is fully TTL-compatible, and operates from a single +5 V supply.

Connection Diagram
40-Pin DIP



- **F6800 and 8080 Bus Compatible**
- **Data Rate From DC to 1M BPS**
- **Bit-Oriented Line Control Protocols: SDLC, ADCCP, HDLC**
  - **Automatic Detection and Generation of Special Control Sequences (e.g., Flag, Abort, Go-Ahead)**
  - **Zero Insertion and Deletion**
  - **Primary or Secondary Station Select**
  - **Global Address**
  - **Automatic Extended Address**
  - **One or Two Control Bytes**
  - **Data Character Length From Five to Eight Bits with 1- to 8-Bit Residual Last Character**
  - **CCITT-CRC Error Detection**
  - **Interrupt on End of Message**
  - **IBM Retail Store Loop Mode**
- **Byte Control Protocol: IBM BISYNC**
  - **Special Character Generation: DLE, SYNC**
  - **Special Character Detection: DLE, SYNC, SOH, STX, ITB, ETB, ETX**
  - **ASCII or EBCDIC**
  - **Non-Transparent Mode and Transparent Mode**
  - **8-Bit Character Length**
  - **Automatic Fill Character Insertion with Selectable Stripping**
  - **CCITT or CRC-16 Error Detection**
  - **Interrupt on End of Message**
- **Byte Control Protocols: DDCMP and Other Programmable SYNC Characters**
  - **5- to 8-Bit Character Length**
  - **Selectable CRC Error Detection**
  - **Automatic Fill Character Insertion with Selectable Stripping**
- **Directly Addressable Parameter Control Registers: Mode, SYNC/Address, Transmitter Control, and Receiver Control**
- **Separate Addressable Status and Data Registers for Receiver and Transmitter**
- **Modem Handshake Signals: RTS, CTS, DTR, DSR, and CD**
- **NRZ or NRZI (Zero-Complementing)**
- **Full- or Half-Duplex Operation**
- **Self-Test Loop Mode**
- **8- or 16-Bit Bidirectional 3-State Data Bus**

Rockwell R6500 Family/R6502

Data courtesy of Rockwell International Electronic Devices.

The **6502** is in some respect based on the **6800** series. Generally the 6800 support chips can be used, and the instruction set has similarites. However, the compatability does not extend to machine code.

It is architecturally less complex than the 6802, but can be considered an enhancement due to its comprehensive addressing modes. BCD addition and subtraction is also provided.

SYSTEM ABSTRACT

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

R6500 MICROPROCESSOR (CPU) CONCEPT

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

The family includes six microprocessors with on-board clock oscillators and drivers and four microprocessors driven by external clocks. The on-chip clock versions are aimed at high performance, low cost applications where single phase inputs, crystal or RC inputs provide the time base. The external clock versions are geared for multiprocessor system applications where maximum timing control is mandatory. All R6500 microprocessors are also available in a variety of packaging (ceramic and plastic), operating frequency (1 MHz, 2 MHz and 3 MHz) and temperature (commercial and industrial) versions.

FEATURES

- Single +5V supply
- N channel, silicon gate, depletion load technology
- Eight bit parallel processing
- 56 Instructions
- Decimal and binary arithmetic
- Thirteen addressing modes
- True indexing capability
- Programmable stack pointer
- Variable length stack
- Interrupt capability
- Non-maskable interrupt
- Use with any type of speed memory
- 8-bit Bidirectional Data Bus
- Addressable memory range of up to 64K bytes
- "Ready" input
- Direct Memory Access capability
- Bus compatible with M6800
- 1 MHz, 2 MHz, and 3 MHz versions
- Choice of external or on-chip clocks
- On-the-chip clock options
  - External single clock input
  - Crystal time base input
- Commercial and industrial temperature versions
- Pipeline architecture

MEMBERS OF THE R6500 MICROPROCESSOR (CPU) FAMILY

Microprocessors with Internal Two Phase Clock Generator

Model	Addressable Memory
R6502	64K Bytes
R6503	4K Bytes
R6504	8K Bytes
R6505	4K Bytes
R6506	4K Bytes
R6507	8K Bytes

Microprocessors with External Two Phase Clock Input

Model	Addressable Memory
R6512	64K Bytes
R6513	4K Bytes
R6514	8K Bytes
R6515	4K Bytes

Ordering Information

Order Number: R65XX _ _ _

- Temperature Range:
  - No suffix = 0°C to +70°C
  - E = -40°C to +85°C (Industrial)
- Package: C = Ceramic, P = Plastic
- Frequency Range:
  - No suffix = 1 MHz
  - A = 2 MHz
  - B = 3 MHz
- Model Designator:
  - XX = 02, 03, 04, . . . 15

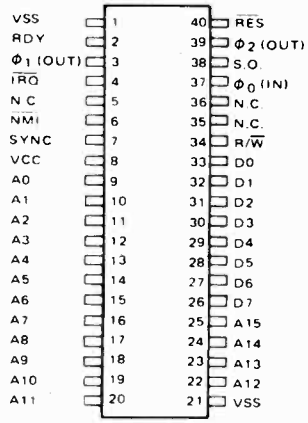
R6500/R6502 CPU

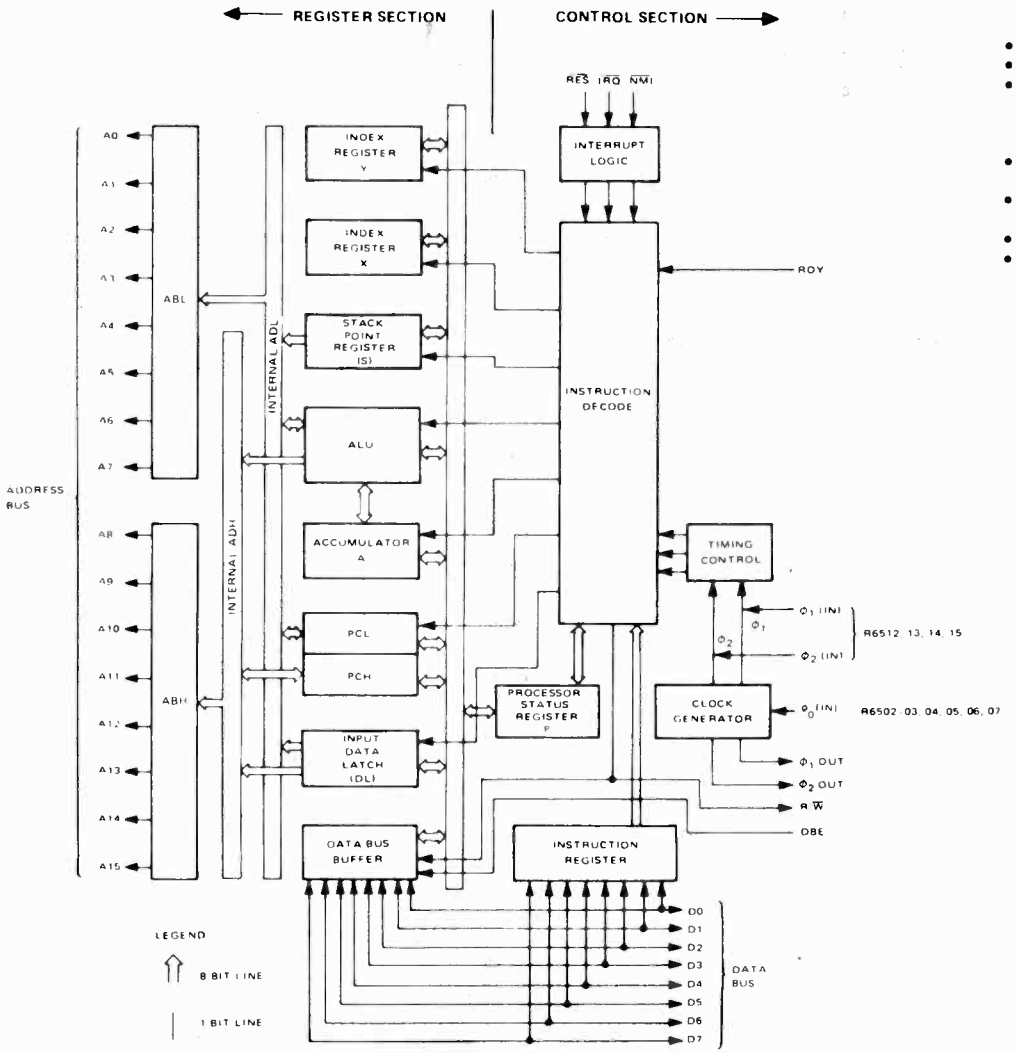
Thirteen addressing modes + true indexing = R6500 software power

The R6500 features 13 addressing modes. The first byte of each instruction is the operation code specifying both the instruction and the addressing mode. The addressing modes are summarized below.

- ACCUMULATOR ADDRESSING. A one byte instruction operating on the accumulator.
- IMMEDIATE ADDRESSING. The operand is in the second byte of the instruction.
- ABSULUTE ADDRESSING. The second and third bytes of the instruction specify the effective address in 65K bytes of addressable memory.
- ZERO PAGE ADDRESSING. Allows shorter code and execution times by assuming a zero-page address.
- INDEXED ZERO PAGE ADDRESSING (X or Y in dexing). Zero page addressing used with an index register.
- INDEXED ABSOLUTE ADDRESSING X or Y in dexing). Absolute addressing used with X or Y index registers.
- IMPLIED ADDRESSING. The register containing the operand is implicitly stated in the operation code.
- RELATIVE ADDRESSING. Used only with branch instructions. The second byte is an Offset added to the contents of the program counter.
- INDEXED INDIRECT ADDRESSING. Uses an indirect zero page address indexed by X to fetch the effective address.
- INDIRECT INDEXED ADDRESSING. Uses a zero page address to fetch the effective base address to be indexed by Y.
- ABSOLUTE INDIRECT. Used only with JMP the second and third bytes point to a two byte effective address.

R6502 – 40 Pin Package





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

R6500 Internal Architecture

Features of R6502

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

ADDRESSING MODES

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

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

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

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

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

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

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

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

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

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

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

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

INSTRUCTION SET – ALPHABETIC SEQUENCE

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

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

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

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

EOR "Exclusive-or" Memory with Accumulator

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

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

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

NOP No Operation

ORA "OR" Memory with Accumulator

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

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

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

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

R6500 Signal Description

Clocks (ϕ_1, ϕ_2)

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

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

Address Bus (A0-A15)

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

Data Bus (D0-D7)

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

Data Bus Enable (DBE)

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

Ready (RDY)

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

Interrupt Request ($\overline{IRQ}$)

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

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

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

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

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

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

Set Overflow Flag (S.O.)

A negative going edge on this input sets the overflow bit in the Status Code Register. This signal is sampled on the trailing edge of ϕ_1 and must be externally synchronized.

SYNC

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

Reset

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

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

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

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

ROCKWELL R6522 VIA

Data courtesy of Rockwell International Electronic Devices.

- Two 8-Bit Bidirectional I/O Ports
- Two 16-Bit Programmable Timer/Counters
- Serial Data Port
- Single +5V Power Supply
- TTL Compatible
- CMOS Compatible Peripheral Control Lines
- Expanded "Handshake" Capability Allows Positive Control of Data Transfers Between Processor and Peripheral Devices
- Latched Output and Input Registers
- 1 MHz and 2 MHz Operation

The R6522 Versatile Interface Adapter (VIA) is a very flexible I/O control device. In addition, this device contains a pair of very powerful 16-bit interval timers, a serial-to-parallel/parallel-to-serial shift register and input data latching on the peripheral ports. Expanded handshaking capability allows control of bi-drrectional data transfers between VIA's in multiple processor systems.

Control of peripheral devices is handled primarily through two 8-bit bi-directional ports. Each line can be programmed as either an input or an output. Several peripheral I/O lines can be controlled directly from the interval timers for generating programmable frequency square waves or for counting externally generated pulses. To facilitate control of the many powerful features of this chip, an interrupt flag register, an interrupt enable register and a pair of function control registers are provided.

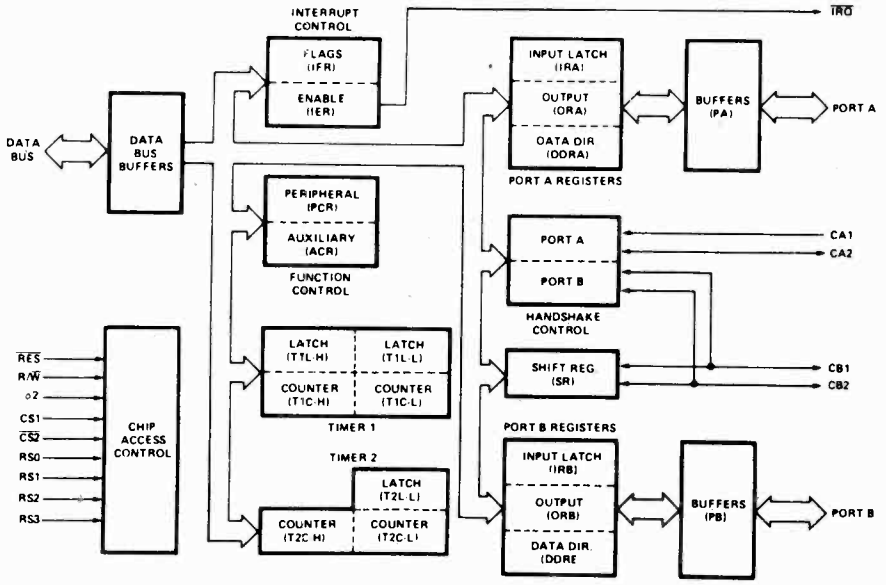


Figure 1. R6522 Block Diagram

ROCKWELL R6520 PIA

Data courtesy of Rockwell International Electronic Devices.

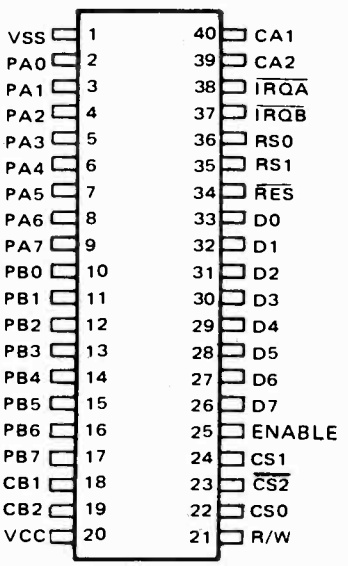
DESCRIPTION

The R6520 Peripheral Interface Adapter is designed to solve a broad range of peripheral control problems in the implementation of microcomputer systems. This device allows a very effective tradeoff between software and hardware by providing significant capability and flexibility in a low cost chip. When coupled with the power and speed of the R6500 family of microprocessors, the R6520 allows implementation of very complex systems at a minimum overall cost.

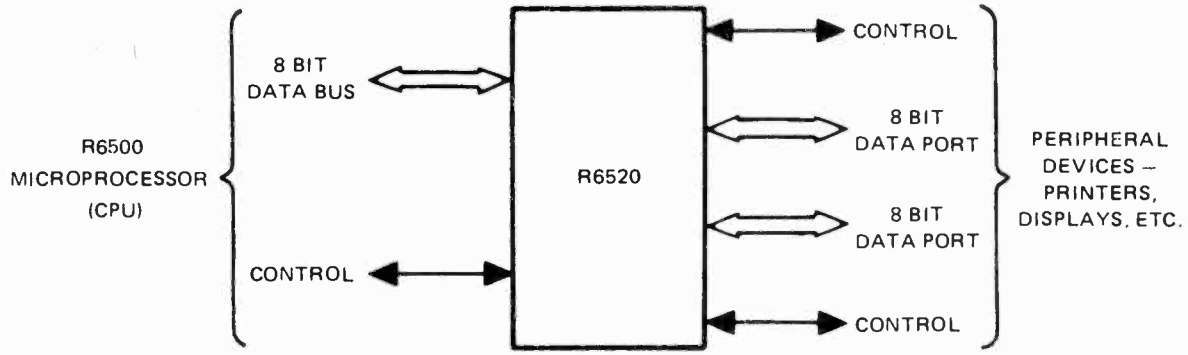
Control of peripheral devices is handled primarily through two 8-bit bidirectional ports. Each of these lines can be programmed to act as either an input or an output. In addition, four peripheral control/interrupt input lines are provided. These lines can be used to interrupt the processor or for "hand shaking" data between the processor and a peripheral device.

FEATURES

- High performance replacement for 6820 type peripheral adapter
- N channel, depletion load technology, single +5V supply
- Completely Static and TTL compatible
- CMOS compatible periphral control line
- Fully automatic "hand shake" allows positive control of data transfers between processor and peripheral devices
- Commercial, industrial and military temperature range versions



Pin Configuration



Basic R6520 Interface Diagram

ROCKWELL R6532 RIOT (RAM, I/O, Interval Timer)

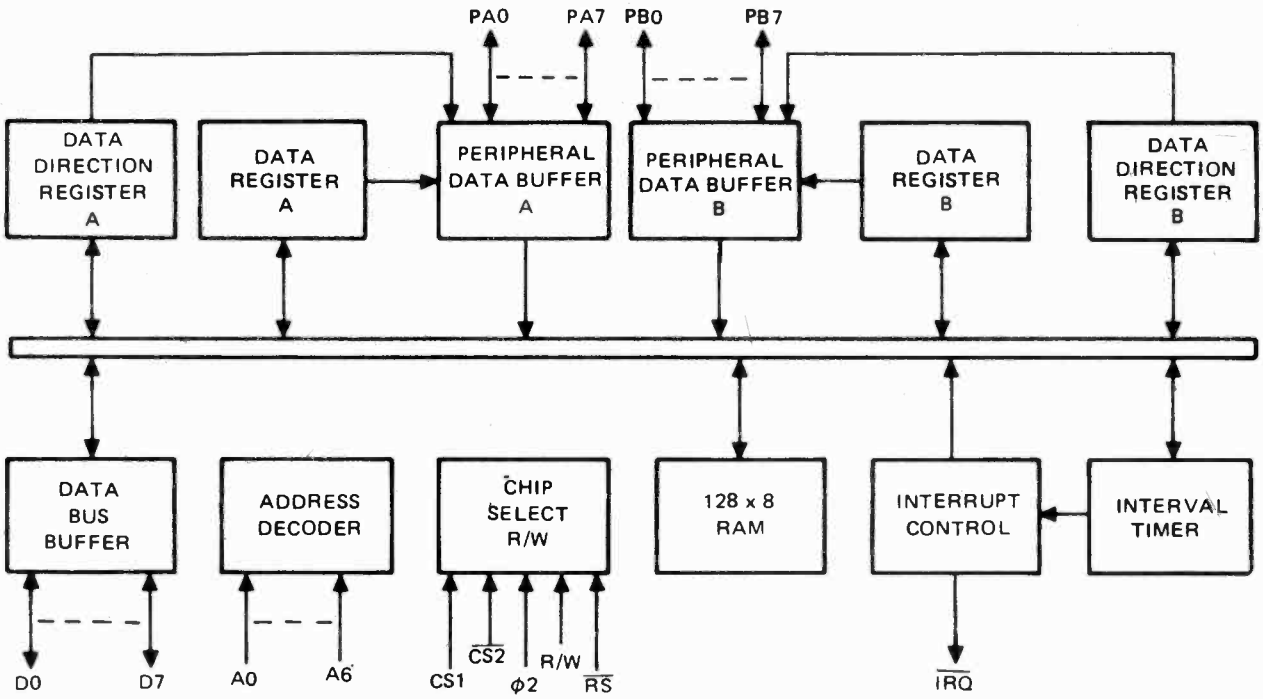
Data courtesy of Rockwell International Electronic Devices.

DESCRIPTION

The R6532 is designed to operate in conjunction with the R6500 Microprocessor Family. It is comprised of a 128 x 8 static RAM, two software-controlled, 8-bit bidirectional data ports allowing direct interfacing between the microcomputer and peripheral devices, a software programmable interval timer with interrupt, capable of timing in various intervals from 1 to 262,144 clock periods, and a programmable edge-detect circuit.

FEATURES

- 8 bit bidirectional Data Bus
- 128 x 8 static RAM
- Two 8 bit bidirectional data ports
- Programmable Interval Timer with Interrupt Capability
- TTL & CMOS compatible peripheral lines
- One port has Direct Transistor Drive Capability
- Programmable edge-sensitive interrupt input
- 6500/6800 bus compatible
- 1 MHz and 2 MHz parts available



R6532 Block Diagram

ROCKWELL R6592 Printer Controller

INTRODUCTION

The Rockwell R6592 is a single-chip printer controller for eight different EPSON* dot-matrix impact printers, models 210, 220, 240, 511L, 512, 522, 541L, and 542. The R6592 offers the flexibility to support any of these models with a minimum of circuitry. Generation of 96 standard ASCII upper and lower case characters and 6 special characters is provided. In addition, up to 10 ASCII control commands are accepted, depending upon the printer. Logic is included in the R6592 to print up to 26 columns on the 210, 220, and 240 models, and up to 40 columns on the 511L, 512, 522, 541L, and 542 models.

Input data may be selected to be in the RS-232 serial format with selectable baud rate from 50 to 7200 bits/second or the parallel format. External circuitry is required to convert RS-232 logic levels to R6592 interface logic levels. An external latch may be required for the R6592 to sample parallel data. If both selectable serial and parallel data interface capability is desired, two external multiplexers are required; one to combine four serial baud select lines and four parallel data interface lines into four R6592 input lines and the other to combine two serial data/control lines and two parallel control lines into two other R6592 input lines.

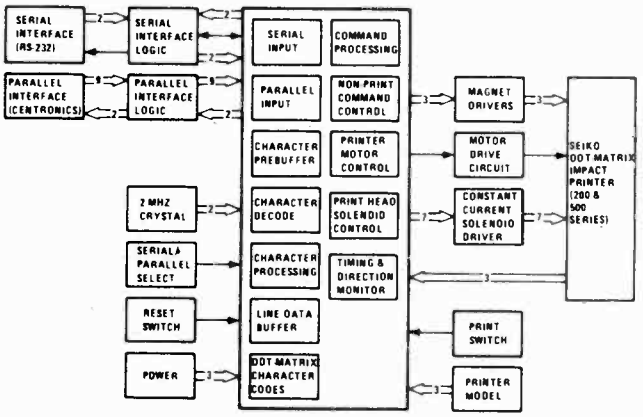
This data sheet summarizes the interface specifications of the R6592. Product Description 29650N56 describes the operation of the R6592 in detail.

*EPSON is a trade name of Shinshu Seiki Co., Ltd., a member of the Seiko Group. EPSON printers are distributed in the United States by C. Itoh Electronics, Inc. The R6592 meets the printer specifications listed in this data sheet.

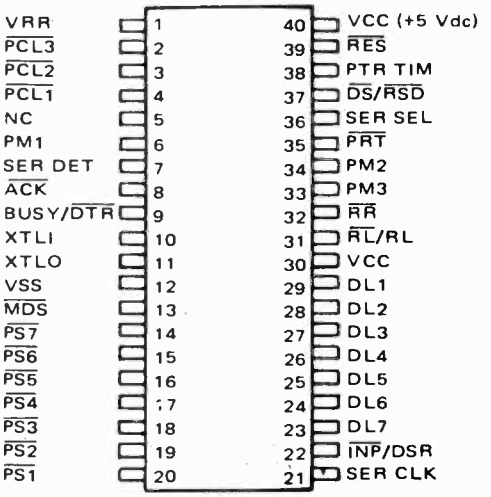
Data courtesy of Rockwell International Electronic Devices.

FEATURES

- Controls EPSON Dot-Matrix Impact Printers:
  - Model 210
  - Model 220
  - Model 240
  - Model 511L
  - Model 512
  - Model 522
  - Model 541L
  - Model 542
- Minimal Support Circuitry Required
- On-Chip 5 x 7 Dot-Matrix Character Generation
- 96 Standard Upper and Lower Case ASCII Characters (7 Bit Code)
- Six Special ASCII Characters (7 Bit Code)
- Up to 10 ASCII Commands Accepted (Printer Dependent)
- Selectable Serial or Parallel Input Data Operation
- Centronics Standard Parallel Interface
  - Seven Data Lines Plus Data Strobe and Input Drive Input
  - Busy and Acknowledge Output
- RS-232C Serial Interface
  - Baud Rate from 50 to 7200 Bits per Second
  - Received Data and Data Set Ready Input
  - Data Terminal Ready Output
- Single +5V ±10% power supply
- 40 pin plastic or ceramic DIP
- 1 MHz operation (2 MHz external crystal)



R6592 Interface Diagram



R6592 Pin Configuration

ROCKWELL R6545-1 CRTC

Data courtesy of Rockwell International Electronic Devices.

DESCRIPTION

The R6545-1 CRT Controller (CRTC) is designed to interface an 8-bit microprocessor to CRT raster scan video displays, and adds an advanced CRT controller to the established and expanding line of R6500 products.

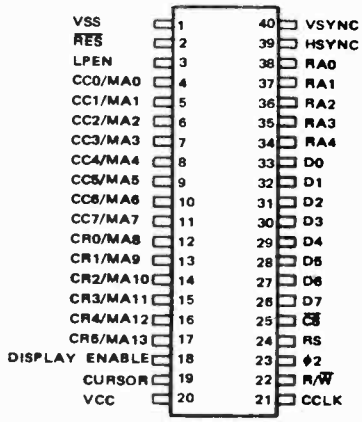
The R6545-1 provides refresh memory addresses and character generator row addresses which allow up to 16K characters with 32 scan lines per character to be addressed. A major advantage of the R6545-1 is that the refresh memory may be addressed in either straight binary or by row/column.

Other functions in the R6545-1 include an internal cursor register which generates a cursor output when its contents are equal to the current refresh address. Programmable cursor start and end registers allow a cursor of up to the full character scan in height to be placed on any scan lines of the character. Variable cursor display blink rates are provided. A light pen strobe input allows capture of the current refresh address in an internal light pen register. The refresh address lines are configured to provide direct dynamic memory refresh.

All timing for the video refresh memory signals is derived from the character clock input. Shift register, latch, and multiplex control signals (when needed) are provided by external high-speed timing. The mode control register allows non-interlaced video display modes at 50 or 60 Hz refresh rate. The internal status register may be used to monitor the R6545-1 operation. The $\overline{RES}$ input allows the CRTC-generated field rate to be dynamically-synchronized with line frequency jitter.

FEATURES

- Compatible with 8-bit microprocessors
- Up to 2.5 MHz character clock operation
- Refresh RAM may be configured in row/column or straight binary addressing
- Alphanumeric and limited graphics capability
- Up and down scrolling by page, line, or character
- Programmable Vertical Sync Width
- Fully programmable display (rows, columns, character matrix)
- Non-interlaced scan
- 50/60 Hz operation
- Fully programmable cursor
- Light pen register
- Addresses refresh RAM to 16K characters
- No external DMA required
- Internal status register
- 40-Pin ceramic or plastic DIP
- Pin-compatible with MC6845
- Single +5 ±5% Volt Power Supply



R6545-1 Pin Configuration

IT LIVES AGAIN!

From the past it came, growing daily, striking terror into the hearts of lesser publications, and spreading its influence across the country in its quest to infiltrate every town, every home, every mind.

Not a horror story, but a success story. And if electronics theory strikes terror into you, then you need the help of **Electronics — It's Easy**. Originally a long running series in Electronics Today International, **Electronics — It's Easy** was printed as a set of three books. They sold out. It was reprinted as a single volume. It sold out. Now this phenominally successful publication is available again, in its third reprint. **Electronics — It's Easy** is a comprehensive and simply written guide which explains the theory (and the practice) of electronics step by step. Every aspect of the subject is covered, starting with the basic principles and working through to the how and why of today's technology.

You can obtain your copy of **Electronics — It's Easy** by mail order using the coupon below. Make cheques or postal orders payable to ASP Ltd, alternatively you may pay by Access or Barclaycard.

Send to: Sales Office (Specials),
513 London Road, Thornton Heath
Surrey CR4 6AR

Please send me.....copies of Electronics — It's Easy. I have enclosed £..... (£4.95 each including p&p).

NAME .....

ADDRESS .....

.....

.....

Please debit my account
My Access/ Barclaycard No. is .....
Signature .....

INTEL 8085AH CPU

Data courtesy of the Intel Corporation (UK) Ltd.

The processor for which the **8080 family** is named is now obsolescent, owing to the multiple supply lines and the many other ICs needed to make the CPU work (eg system controllers, clock generator).

The family, however, goes on. There is the **8085**, which is in effect an 8080 with all the central functions on one chip. There is a high performance sixteen bit CPU, the **8086**, and a development of it having an eight bit data bus, but a sixteen bit internal architecture.

The family also has a well known and powerful derivative, the **Z80**. Its instruction set is a superset of the 8080, and it is one of the most powerful eight bit processors available.

Much more commonly used today in new designs where the **8080** would have been used is the **8085**. This has two more instructions than the 8080 (RIM and SIM, relating to serial data transfer), otherwise its instruction set is identical.

It is an enhancement of the 8080 in that it has (as is normal nowadays) a single +5 supply, and internal clock generator, needing only an external crystal or RC network and an internal system controller.

Few specific support ICs are available for use with the 8085; 8080 support ICs can be used with the aid of an **8212** or **74LS373** to de-multiplex the bus.

- **Single +5V Power Supply with 10% Voltage Margins**
- **3 MHz, 5 MHz and 6 MHz Selections Available**
- **20% Lower Power Consumption than 8085A for 3 MHz and 5 MHz**
- **1.3 μs Instruction Cycle (8085AH); 0.8 μs (8085AH-2); 0.67 μs (8085AH-1)**
- **100% Compatible with 8085A**
- **100% Software Compatible with 8080A**
- **On-Chip Clock Generator (with External Crystal, LC or RC Network)**
- **On-Chip System Controller; Advanced Cycle Status Information Available for Large System Control**
- **Four Vectored Interrupt Inputs (One is Non-Maskable) Plus an 8080A-Compatible Interrupt**
- **Serial In/Serial Out Port**
- **Decimal, Binary and Double Precision Arithmetic**
- **Direct Addressing Capability to 64K Bytes of Memory**
- **Available in EXPRESS**
  - **Standard Temperature Range**
  - **Extended Temperature Range**

The Intel® 8085AH is a complete 8 bit parallel Central Processing Unit (CPU) implemented in N-channel, depletion load, silicon gate technology (HMOS). Its instruction set is 100% software compatible with the 8080A microprocessor, and it is designed to improve the present 8080A's performance by higher system speed. Its high level of system integration allows a minimum system of three IC's [8085AH (CPU), 8156H (RAM/IO) and 8355/8755A (ROM/PROM/IO)] while maintaining total system expandability. The 8085AH-2 and 8085AH-1 are faster versions of the 8085AH.

The 8085AH incorporates all of the features that the 8224 (clock generator) and 8228 (system controller) provided for the 8080A, thereby offering a high level of system integration.

The 8085AH uses a multiplexed data bus. The address is split between the 8 bit address bus and the 8 bit data bus. The on-chip address latches of 8155H/8156H/8355/8755A memory products allow a direct interface with the 8085AH.

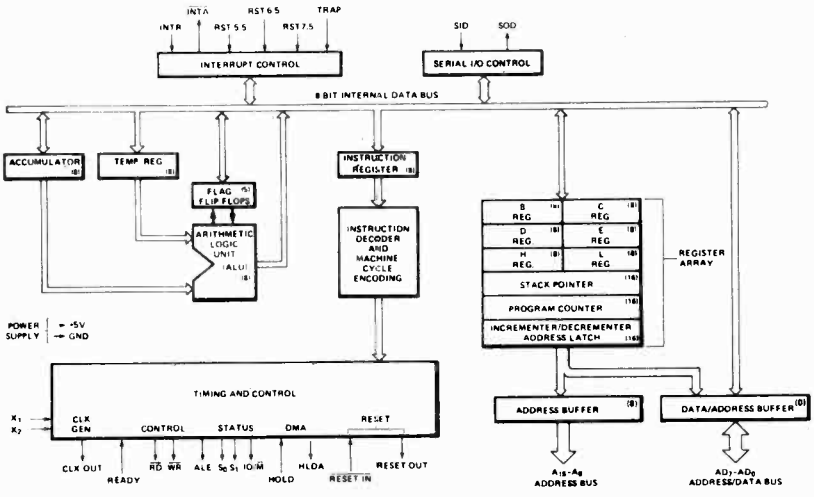


Figure 1. 8085AH CPU Functional Block Diagram

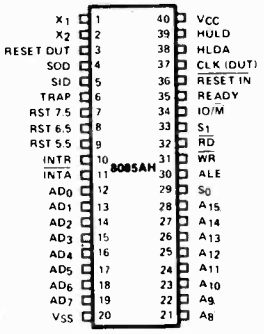


Figure 2. 8085AH Pin Configuration

Table 1. Pin Description

Symbol	Type	Name and Function
A_8–A_{15}	O	**Address Bus:** The most significant 8 bits of the memory address or the 8 bits of the I/O address, 3-stated during Hold and Halt modes and during RESET.
$AD_{0—7}$	I/O	**Multiplexed Address/Data Bus:** Lower 8 bits of the memory address (or I/O address) appear on the bus during the first clock cycle (T state) of a machine cycle. It then becomes the data bus during the second and third clock cycles.
ALE	O	**Address Latch Enable:** It occurs during the first clock state of a machine cycle and enables the address to get latched into the on-chip latch of peripherals. The falling edge of ALE is set to guarantee setup and hold times for the address information. The falling edge of ALE can also be used to strobe the status information. ALE is never 3-stated.
S_0, S_1, and IO/$\overline{M}$	O	**Machine Cycle Status:** IO/$\overline{M}$ S_1 S_0 Status 0 0 1 Memory write 0 1 0 Memory read 1 0 1 I/O write 1 1 0 I/O read 0 1 1 Opcode fetch 1 1 1 Interrupt Acknowledge * 0 0 Halt * X X Hold * X X Reset * = 3-state (high impedance) X = unspecified S_1 can be used as an advanced R/$\overline{W}$ status. IO/$\overline{M}$, S_0 and S_1 become valid at the beginning of a machine cycle and remain stable throughout the cycle. The falling edge of ALE may be used to latch the state of these lines.
$\overline{RD}$	O	**Read Control:** A low level on $\overline{RD}$ indicates the selected memory or I/O device is to be read and that the Data Bus is available for the data transfer, 3-stated during Hold and Halt modes and during RESET.
$\overline{WR}$	O	**Write Control:** A low level on $\overline{WR}$ indicates the data on the Data Bus is to be written into the selected memory or I/O location. Data is set up at the trailing edge of $\overline{WR}$. 3-stated during Hold and Halt modes and during RESET.

Symbol	Type	Name and Function
READY	I	**Ready:** If READY is high during a read or write cycle, it indicates that the memory or peripheral is ready to send or receive data. If READY is low, the cpu will wait an integral number of clock cycles for READY to go high before completing the read or write cycle. READY must conform to specified setup and hold times.
HOLD	I	**Hold:** Indicates that another master is requesting the use of the address and data buses. The cpu, upon receiving the hold request, will relinquish the use of the bus as soon as the completion of the current bus transfer. Internal processing can continue. The processor can regain the bus only after the HOLD is removed. When the HOLD is acknowledged, the Address, Data $\overline{RD}$, $\overline{WR}$, and IO/$\overline{M}$ lines are 3-stated.
HLDA	O	**Hold Acknowledge:** Indicates that the cpu has received the HOLD request and that it will relinquish the bus in the next clock cycle. HLDA goes low after the Hold request is removed. The cpu takes the bus one half clock cycle after HLDA goes low.
INTR	I	**Interrupt Request:** Is used as a general purpose interrupt. It is sampled only during the next to the last clock cycle of an instruction and during Hold and Halt states. If it is active, the Program Counter (PC) will be inhibited from incrementing and an $\overline{INTA}$ will be issued. During this cycle a RESTART or CALL instruction can be inserted to jump to the interrupt service routine. The INTR is enabled and disabled by software. It is disabled by Reset and immediately after an interrupt is accepted.
$\overline{INTA}$	O	**Interrupt Acknowledge:** Is used instead of (and has the same timing as) $\overline{RD}$ during the Instruction cycle after an INTR is accepted. It can be used to activate an 8259A Interrupt chip or some other interrupt port.
RST 5.5 RST 6.5 RST 7.5	I	**Restart Interrupts:** These three inputs have the same timing as INTR except they cause an internal RESTART to be automatically inserted. The priority of these interrupts is ordered as shown in Table 2. These interrupts have a higher priority than INTR. In addition, they may be individually masked out using the SIM instruction.

TRAP	I	**Trap:** Trap interrupt is a nonmaskable RESTART interrupt. It is recognized at the same time as INTR or RST 5.5-7.5. It is unaffected by any mask or Interrupt Enable. It has the highest priority of any interrupt. (See Table 2.)

RESET OUT	O	**Reset Out:** Reset Out indicates cpu is being reset. Can be used as a system reset. The signal is synchronized to the processor clock and lasts an integral number of clock periods.

Table 1. Pin Description (Continued)

Symbol	Type	Name and Function
$\overline{\text{RESET IN}}$	I	**Reset In:** Sets the Program Counter to zero and resets the Interrupt Enable and HLDA flip-flops. The data and address buses and the control lines are 3-stated during RESET and because of the asynchronous nature of RESET, the processor's internal registers and flags may be altered by RESET with unpredictable results. $\overline{\text{RESET IN}}$ is a Schmitt-triggered input, allowing connection to an R-C network for power-on RESET delay (see Figure 3). Upon power-up, $\overline{\text{RESET IN}}$ must remain low for at least 10 ms after minimum V_{CC} has been reached. For proper reset operation after the power-up duration, $\overline{\text{RESET IN}}$ should be kept low a minimum of three clock periods. The CPU is held in the reset condition as long as $\overline{\text{RESET IN}}$ is applied.

Symbol	Type	Name and Function
X_1, X_2	I	**X_1 and X_2:** Are connected to a crystal, LC, or RC network to drive the internal clock generator. X_1 can also be an external clock input from a logic gate. The input frequency is divided by 2 to give the processor's internal operating frequency.
CLK	O	**Clock:** Clock output for use as a system clock. The period of CLK is twice the X_1, X_2 input period.
SID	I	**Serial Input Data Line:** The data on this line is loaded into accumulator bit 7 whenever a RIM instruction is executed.
SOD	O	**Serial Output Data Line:** The output SOD is set or reset as specified by the SIM instruction.
V_{CC}		**Power:** +5 volt supply.
V_{SS}		**Ground:** Reference.

Table 2. Interrupt Priority, Restart Address, and Sensitivity

Name	Priority	Address Branched To (1) When Interrupt Occurs	Type Trigger
TRAP	1	24H	Rising edge AND high level until sampled.
RST 7.5	2	3CH	Rising edge (latched).
RST 6.5	3	34H	High level until sampled.
RST 5.5	4	2CH	High level until sampled.
INTR	5	See Note (2).	High level until sampled.

NOTES:
1. The processor pushes the PC on the stack before branching to the indicated address.
2. The address branched to depends on the instruction provided to the cpu when the interrupt is acknowledged.

FUNCTIONAL DESCRIPTION

The 8085AH is a complete 8-bit parallel central processor. It is designed with N-channel, depletion load, silicon gate technology (HMOS), and requires a single +5 volt supply. Its basic clock speed is 3 MHz (8085AH), 5 MHz (8085AH-2), or 6 MHz (8085AH-1), thus improving on the present 8080A's performance with higher system speed. Also it is designed to fit into a minimum system of three IC's: The CPU (8085AH), a RAM/IO (8156H), and a ROM or EPROM/IO chip (8355 or 8755A).

The 8085AH has twelve addressable 8-bit registers. Four of them can function only as two 16-bit register pairs. Six others can be used interchangeably as 8-bit registers or as 16-bit register pairs. The 8085AH register set is as follows:

Mnemonic	Register	Contents
ACC or A	Accumulator	8 bits
PC	Program Counter	16-bit address
BC,DE,HL	General-Purpose Registers; data pointer (HL)	8 bits x 6 or 16 bits x 3
SP	Stack Pointer	16-bit address
Flags or F	Flag Register	5 flags (8-bit space)

The 8085AH uses a multiplexed Data Bus. The address is split between the higher 8-bit Address Bus and the lower 8-bit Address/Data Bus. During the first T state (clock cycle) of a machine cycle the low order address is sent out on the Address/Data bus. These lower 8 bits may be latched externally by the Address Latch Enable signal (ALE). During the rest of the machine cycle the data bus is used for memory or I/O data.

The 8085AH provides $\overline{\text{RD}}$, $\overline{\text{WR}}$, S_0, S_1, and IO/$\overline{\text{M}}$ signals for bus control. An Interrupt Acknowledge signal ($\overline{\text{INTA}}$) is also provided. HOLD and all Interrupts are synchronized with the processor's internal clock. The 8085AH also provides Serial Input Data (SID) and Serial Output Data (SOD) lines for simple serial interface.

In addition to these features, the 8085AH has three maskable, vector interrupt pins, one nonmaskable TRAP interrupt, and a bus vectored interrupt, INTR.

INTERRUPT AND SERIAL I/O

The 8085AH has 5 interrupt inputs: INTR, RST 5.5, RST 6.5, RST 7.5, and TRAP. INTR is identical in function to the 8080A INT. Each of the three RESTART inputs, 5.5, 6.5, and 7.5, has a programmable mask. TRAP is also a RESTART interrupt but it is nonmaskable.

The three maskable interrupts cause the internal execution of RESTART (saving the program counter in the stack and branching to the RESTART address) if the interrupts are enabled and if the interrupt mask is not set. The nonmaskable TRAP causes the internal execution of a RESTART vector independent of the state of the interrupt enable or masks. (See Table 2.)

There are two different types of inputs in the restart interrupts. RST 5.5 and RST 6.5 are *high level-sensitive* like INTR (and INT on the 8080) and are recognized with the same timing as INTR. RST 7.5 is *rising edge-sensitive*.

For RST 7.5, only a pulse is required to set an internal flip-flop which generates the internal interrupt request (a normally high level signal with a low going pulse is recommended for highest system noise immunity). The RST 7.5 request flip-flop remains set until the request is serviced. Then it is reset automatically. This flip-flop may also be reset by using the SIM instruction or by issuing a $\overline{\text{RESET IN}}$ to the 8085AH. The RST 7.5 internal flip-flop will be set by a pulse on the RST 7.5 pin even when the RST 7.5 interrupt is masked out.

The status of the three RST interrupt masks can only be affected by the SIM instruction and $\overline{\text{RESET IN}}$. (See SIM, Chapter 5 of the MCS-80/85 User's Manual.)

The interrupts are arranged in a fixed priority that determines which interrupt is to be recognized if more than one is pending as follows: TRAP—highest priority, RST 7.5, RST 6.5, RST 5.5, INTR—lowest priority. This priority scheme does not take into account the priority of a routine that was started by a higher priority interrupt. RST 5.5 can interrupt an RST 7.5 routine if the interrupts are re-enabled before the end of the RST 7.5 routine.

The TRAP interrupt is useful for catastrophic events such as power failure or bus error. The TRAP input is recognized just as any other interrupt but has the highest priority. It is not affected by any flag or mask. The TRAP input is both *edge and level sensitive*. The TRAP input must go high and remain high until it is acknowledged. It will not be recognized again until it goes low, then high again. This avoids any false triggering due to noise or logic glitches. Figure 4 illustrates the TRAP interrupt request circuitry within the 8085AH. Note that the servicing of any interrupt (TRAP, RST 7.5, RST 6.5, RST 5.5, INTR) disables all future interrupts (except TRAPs) until an EI instruction is executed.

The TRAP interrupt is special in that it disables interrupts, but preserves the previous interrupt enable status. Performing the first RIM instruction following a TRAP interrupt allows you to determine whether interrupts were enabled or disabled prior to the TRAP. All subsequent RIM instructions provide current interrupt enable status. Performing a RIM instruction following INTR, or RST 5.5–7.5 will provide current Interrupt Enable status, revealing that Interrupts are disabled. See the description of the RIM instruction in the MCS-80/85 Family User's Manual.

The serial I/O system is also controlled by the RIM and SIM instructions. SID is read by RIM, and SIM sets the SOD data.

DRIVING THE X_1 AND X_2 INPUTS

You may drive the clock inputs of the 8085AH, 8085AH-2, or 8085AH-1 with a crystal, an LC tuned circuit, an RC network, or an external clock source. The crystal frequency must be at least 1 MHz, and must be twice the desired internal clock frequency; hence, the 8085AH is operated with a 6 MHz crystal (for 3 MHz clock), the 8085AH-2 operated with a 10 MHz crystal (for 5 MHz clock), and the 8085AH-1 can be operated with a 12 MHz crystal (for 6 MHz clock).

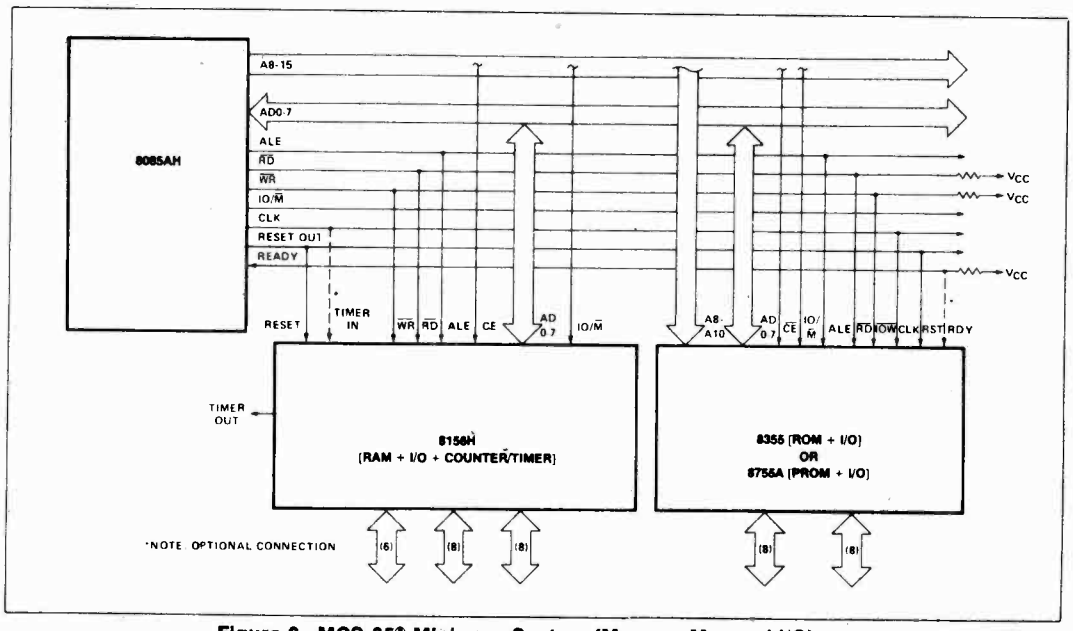


Figure 8. MCS-85® Minimum System (Memory Mapped I/O)

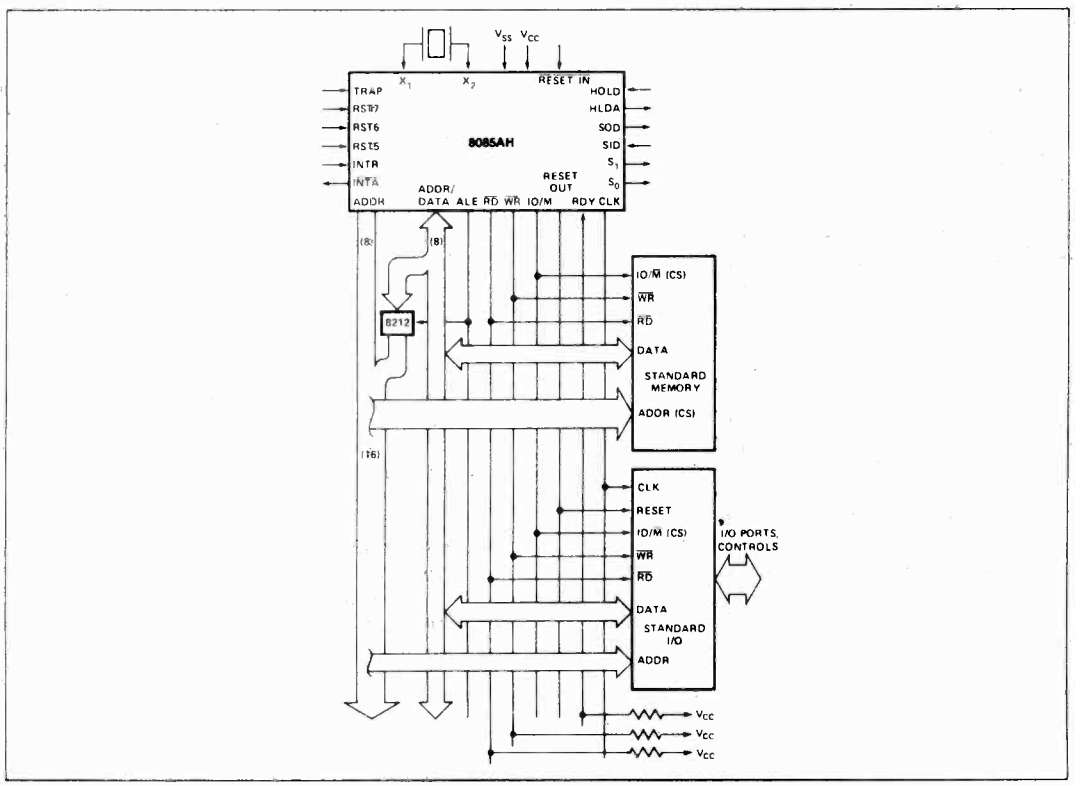


Figure 9. MCS-85® System (Using Standard Memories)

SYSTEM INTERFACE

The 8085AH family includes memory components, which are directly compatible to the 8085AH CPU. For example, a system consisting of the three chips, 8085AH, 8156H, and 8355 will have the following features:

- 2K Bytes ROM
- 256 Bytes RAM
- 1 Timer/Counter
- 4 8-bit I/O Ports
- 1 6-bit I/O Port
- 4 Interrupt Levels
- Serial In/Serial Out Ports

This minimum system, using the standard I/O technique is as shown in Figure 7.

In addition to standard I/O, the memory mapped I/O offers an efficient I/O addressing technique. With this technique, an area of memory address space is assigned for I/O address, thereby, using the memory address for I/O manipulation. Figure 8 shows the system configuration of Memory Mapped I/O using 8085AH.

The 8085AH CPU can also interface with the standard memory that does *not* have the multiplexed address/data bus. It will require a simple 8212 (8-bit latch) as shown in Figure 9.

GENERATING AN 8085AH WAIT STATE

If your system requirements are such that slow memories or peripheral devices are being used, the circuit shown in Figure 6 may be used to insert one WAIT state in each 8085AH machine cycle.

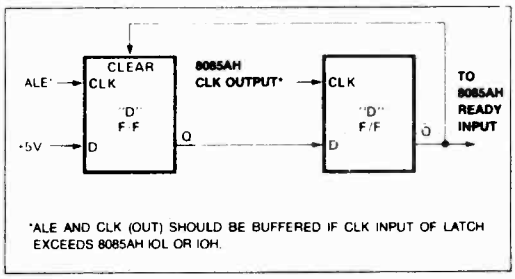


Figure 6. Generation of a Wait State for 8085AH CPU

Table 6. Instruction Set Summary

Mnemonic	Instruction Code D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Operations Description
MOVE, LOAD, AND STORE									
MOVr1 r2	0	1	D	D	D	S	S	S	Move register to register
MOV M,r	0	1	1	1	0	S	S	S	Move register to memory
MOV r,M	0	1	D	D	D	1	1	0	Move memory to register
MVI r	0	0	D	D	D	1	1	0	Move immediate register
MVI M	0	0	1	1	0	1	1	0	Move Immediate memory
LXI B	0	0	0	0	0	0	0	1	Load immediate register Pair B & C
LXI D	0	0	0	1	0	0	0	1	Load immediate register Pair D & E
LXI H	0	0	1	0	0	0	0	1	Load immediate register Pair H & L
STAX B	0	0	0	0	0	0	1	0	Store A indirect
STAX D	0	0	0	1	0	0	1	0	Store A indirect
LDAX B	0	0	0	0	1	0	1	0	Load A indirect
LDAX D	0	0	0	1	1	0	1	0	Load A indirect
STA	0	0	1	1	0	0	1	0	Store A direct
LDA	0	0	1	1	1	0	1	0	Load A direct
SHLD	0	0	1	0	0	0	1	0	Store H & L direct
LHLD	0	0	1	0	1	0	1	0	Load H & L direct
XCHG	1	1	1	0	1	0	1	1	Exchange D & E, H & L Registers
STACK OPS									
PUSH B	1	1	0	0	0	1	0	1	Push register Pair B & C on stack
PUSH D	1	1	0	1	0	1	0	1	Push register Pair D & E on stack
PUSH H	1	1	1	0	0	1	0	1	Push register Pair H & L on stack
PUSH PSW	1	1	1	1	0	1	0	1	Push A and Flags on stack
POP B	1	1	0	0	0	0	0	1	Pop register Pair B & C off stack
POP D	1	1	0	1	0	0	0	1	Pop register Pair D & E off stack
POP H	1	1	1	0	0	0	0	1	Pop register Pair H & L off stack
POP PSW	1	1	1	1	0	0	0	1	Pop A and Flags off stack
XTHL	1	1	1	0	0	0	1	1	Exchange top of stack, H & L
SPHL	1	1	1	1	1	0	0	1	H & L to stack pointer
LXI SP	0	0	1	1	0	0	0	1	Load immediate stack pointer
INX SP	0	0	1	1	0	0	1	1	Increment stack pointer
DCX SP	0	0	1	1	1	0	1	1	Decrement stack pointer
JUMP									
JMP	1	1	0	0	0	0	1	1	Jump unconditional
JC	1	1	0	1	1	0	1	0	Jump on carry
JNC	1	1	0	1	0	0	1	0	Jump on no carry
JZ	1	1	0	0	1	0	1	0	Jump on zero
JNZ	1	1	0	0	0	0	1	0	Jump on no zero
JP	1	1	1	1	0	0	1	0	Jump on positive
JM	1	1	1	1	1	0	1	0	Jump on minus
JPE	1	1	1	0	1	0	1	0	Jump on parity even
JPO	1	1	1	0	0	0	1	0	Jump on parity odd
PCHL	1	1	1	0	1	0	0	1	H & L to program counter
CALL									
CALL	1	1	0	0	1	1	0	1	Call unconditional
CC	1	1	0	1	1	1	0	0	Call on carry
CNC	1	1	0	1	0	1	0	0	Call on no carry
LOGICAL									
ANA r	1	0	1	0	0	S	S	S	And register with A
XRA r	1	0	1	0	1	S	S	S	Exclusive OR register with A
ORA r	1	0	1	1	0	S	S	S	OR register with A
CMP r	1	0	1	1	1	S	S	S	Compare register with A
ANA M	1	0	1	0	0	1	1	0	And memory with A
XRA M	1	0	1	0	1	1	1	0	Exclusive OR memory with A
ORA M	1	0	1	1	0	1	1	0	OR memory with A
CMP M	1	0	1	1	1	1	1	0	Compare memory with A
ANI	1	1	1	0	0	1	1	0	And immediate with A
XRI	1	1	1	0	1	1	1	0	Exclusive OR immediate with A
ORI	1	1	1	1	0	1	1	0	OR immediate with A
CPI	1	1	1	1	1	1	1	0	Compare immediate with A
ROTATE									
RLC	0	0	0	0	0	1	1	1	Rotate A left
RRC	0	0	0	0	1	1	1	1	Rotate A right
RAL	0	0	0	1	0	1	1	1	Rotate A left through carry
RAR	0	0	0	1	1	1	1	1	Rotate A right through carry

Mnemonic	Instruction Code D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	Operations Description
CZ	1	1	0	0	1	1	0	0	Call on zero
CNZ	1	1	0	0	0	1	0	0	Call on no zero
CP	1	1	1	1	0	1	0	0	Call on positive
CM	1	1	1	1	1	1	0	0	Call on minus
CPE	1	1	1	0	1	1	0	0	Call on parity even
CPO	1	1	1	0	0	1	0	0	Call on parity odd
RETURN									
RET	1	1	0	0	1	0	0	1	Return
RC	1	1	0	1	1	0	0	0	Return on carry
RNC	1	1	0	1	0	0	0	0	Return on no carry
RZ	1	1	0	0	1	0	0	0	Return on zero
RNZ	1	1	0	0	0	0	0	0	Return on no zero
RP	1	1	1	1	0	0	0	0	Return on positive
RM	1	1	1	1	1	0	0	0	Return on minus
RPE	1	1	1	0	1	0	0	0	Return on parity even
RPO	1	1	1	0	0	0	0	0	Return on parity odd
RESTART									
RST	1	1	A	A	A	1	1	1	Restart
INPUT/OUTPUT									
IN	1	1	0	1	1	0	1	1	Input
OUT	1	1	0	1	0	0	1	1	Output
INCREMENT AND DECREMENT									
INR r	0	0	D	D	D	1	0	0	Increment register
DCR r	0	0	D	D	D	1	0	1	Decrement register
INR M	0	0	1	1	0	1	0	0	Increment memory
DCR M	0	0	1	1	0	1	0	1	Decrement memory
INX B	0	0	0	0	0	0	1	1	Increment B & C registers
INX D	0	0	0	1	0	0	1	1	Increment D & E registers
INX H	0	0	1	0	0	0	1	1	Increment H & L registers
DCX B	0	0	0	0	1	0	1	1	Decrement B & C
DCX D	0	0	0	1	1	0	1	1	Decrement D & E
DCX H	0	0	1	0	1	0	1	1	Decrement H & L
ADD									
ADD r	1	0	0	0	0	S	S	S	Add register to A
ADC r	1	0	0	0	1	S	S	S	Add register to A with carry
ADD M	1	0	C	0	0	1	1	0	Add memory to A
ADC M	1	0	0	0	1	1	1	0	Add memory to A with carry
ADI	1	1	0	0	0	1	1	0	Add immediate to A
ACI	1	1	0	0	1	1	1	0	Add immediate to A with carry
DAD B	0	0	0	0	1	0	0	1	Add B & C to H & L
DAD D	0	0	0	1	1	0	0	1	Add D & E to H & L
DAD H	0	0	1	0	1	0	0	1	Add H & L to H & L
OAD SP	0	0	1	1	1	0	0	1	Add stack pointer to H & L
SUBTRACT									
SUB r	1	0	0	1	0	S	S	S	Subtract register from A
SBB r	1	0	0	1	1	S	S	S	Subtract register from A with borrow
SUB M	1	0	0	1	0	1	1	0	Subtract memory from A
SBB M	1	0	0	1	1	1	1	0	Subtract memory from A with borrow
SUI	1	1	0	1	0	1	1	0	Subtract immediate from A
SBI	1	1	0	1	1	1	1	0	Subtract immediate from A with borrow
SPECIALS									
CMA	0	0	1	0	1	1	1	1	Complement A
STC	0	0	1	1	0	1	1	1	Set carry
CMC	0	0	1	1	1	1	1	1	Complement carry
DAA	0	0	1	0	0	1	1	1	Decimal adjust A
CONTROL									
EI	1	1	1	1	1	0	1	1	Enable Interrupts
DI	1	1	1	1	0	0	1	1	Disable Interrupt
NOP	0	0	0	0	0	0	0	0	No-operation
HLT	0	1	1	1	0	1	1	0	Halt
NEW 8085A INSTRUCTIONS									
RIM	0	0	1	0	0	0	0	0	Read Interrupt Mask
SIM	0	0	1	1	0	0	0	0	Set Interrupt Mask

NOTES:

1. DDS or SSS: B 000, C 001, D 010, E011, H 100, L 101, Memory 110, A 111
2. Two possible cycle times (6/12) indicate instruction cycles dependent on condition flags.

*All mnemonics copyrighted ©Intel Corporation 1976.

MITSUBISHI M5L 8086S CPU

Data courtesy of Mitsubishi Electric (UK) Ltd.

The **8086** is a powerful sixteen bit microprocessor having a twenty line address bus with the lower sixteen bits multiplexed with the data. It has a pin labelled MN/MX, which switches the internal functions of the microprocessor to miniumum or maximum mode. The major difference is that in minimum mode the IC uses its own internal bus controller, but in maximum mode it uses a more comprehensive external bus controller, the 8288.

DESCRIPTION

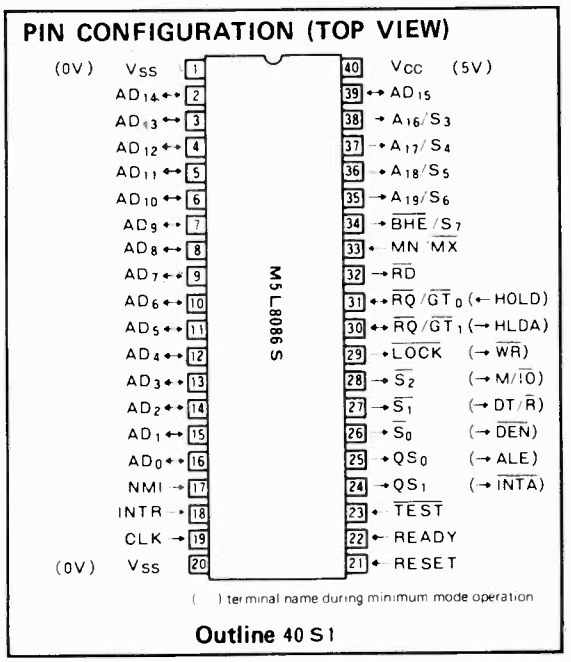
The M5L8086S is a 16-bit parallel microprocessor fabricated using high-speed N-channel silicon-gate ED-MOS techology. It requires a single 5V power supply and has a maximum basic clock rate of 5MHz.

The M5L8086S is upward compatible, both in hardware and software, with the M5L8080AP, S and M5L8085AP, S therefore it can replace either of these devices. It has higher performance because of additional and more powerful operation and addressing functions and instructions.

FEATURES

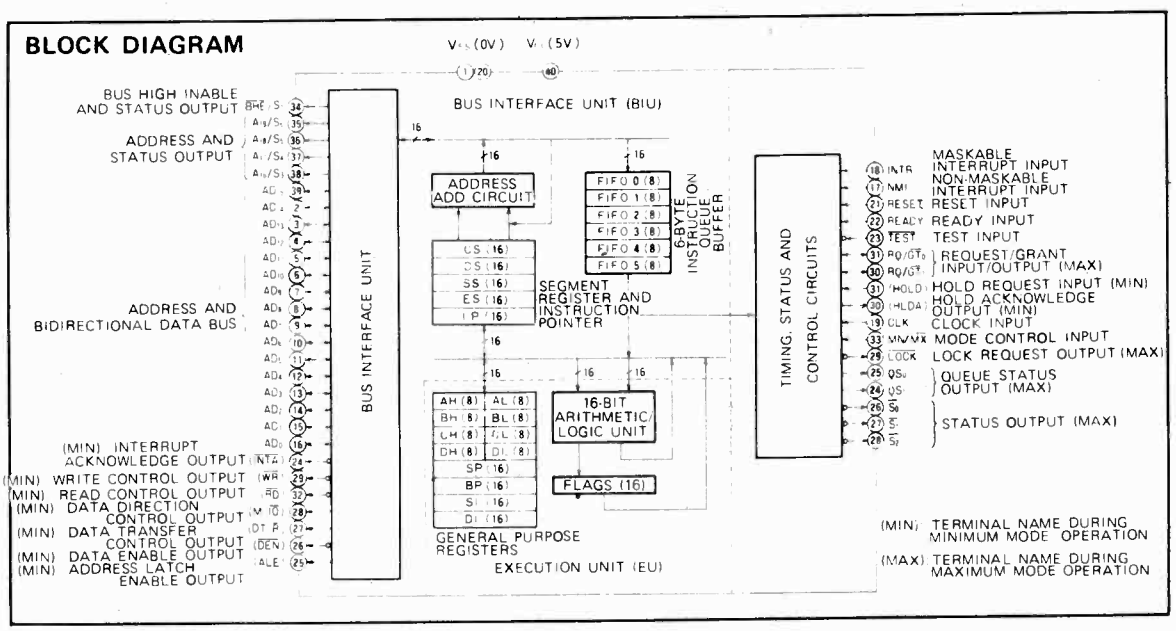
- Direct addressing: 1M byte
- Instruction set upward compatible with that of M5L–8080AP, S
- Enlarged powerful addressing: 24 modes
- On chip 16-bit registers: 14 registers
- Arithmetic operations include multiplication and division, signed or unsigned and 8-bit or 16-bit operands.
- Basic clock rate: 5MHz (max.)
- Multi-CPU functions
- Single 5V power supply
- Interchangeable with the Intel 8086 in pin configuration and electrical characteristics

PIN CONFIGURATION (TOP VIEW)



Outline 40 S1

BLOCK DIAGRAM



M5L 8087S CPU

FUNCTIONS

The M5L8086S has a minimum and maximum mode, which allows the composition to be selected to match the scale of the system in which it is used. The internal function consists of execution unit (EU) and bus interface unit (BIU). The BIU controls the 6-byte instruction queue, while generating addresses, and decodes instructions to be executed by the EU. Each unit operates asynchronously and can access the instruction queue.

The pipeline architecture increased the throughput of the system. The ability to select 8-bit bytes or 16-bit words by using terminals A_0 and $\overline{BHE}$, allows more efficient use of memory. This along with a large direct addressable memory (up to 1 M bytes) makes it practical to process large complicated programs. Two kinds of external interrupt input are provided. The INTR is a maskable interrupt input for the normal interrupt applications, while the NMI is a nonmaskable interrupt for the use of a higher priority interrupt such as power down. In addition to external interrupts, internal interrupts can be initiated by software with the overflow and so on.

PIN DESCRIPTIONS

Pins which have the same functions in minimum or maximum mode

Pin	Name	Input or Output	Functional description
AD_0–AD_{15}	Address and data bas	Input/output	AD_0~AD_{15} is used as both an address bus (A_0~A_{15}) and a data bus (D_0~D_{15}). Though time sharing it outputs addresses during T_1 state and outputs data during T_2, T_3, T_w, T_4 states
A_{19}/S_6 | A_{16}/S_3	Address and status	Output	The high-order 4 bits (A_{16}~A_{19}) and status (S_3~S_6) are output using time sharing techniques. The address bits are output during T_1 state and data are output during T_2, T_3, T_w, T_4 states. The status bits S_3 and S_4 determine which segment register is used in the bus cycle as follows. S_4 S_3 0 0 extra segment 0 1 stack segment 1 0 code segment or none 1 1 data segment while S_5 shows the interrupt enable flag and starts the beginning of a clock cycle. Status bit S_6 is always 0.
$\overline{BHE}/S_7$	Bus high enable and status	Output	Bus high enable ($\overline{BHE}$) and status are output using time sharing techniques. Bus high enable is output during T_1 state and status is output during T_2, T_3, T_w, T_4 states. $\overline{BHE}$ along with A_0 is used to select byte or word unit processing. The selection is as shown below. $\overline{BHE}$ A_0 0 0 word processing (16 bits) 0 1 high-order byte processing (8 bits) 1 0 low-order byte processing (8 bits) 1 1 undefined This pin goes to low-level during the first clock cycle of an interrupt acknowledge cycle. S_7 is a spare status bit.
$\overline{RD}$	Read control	Output	An active "L" signal indicates read timing from memory or an I/O port.
READY	Ready	Input	Signal indicating data transfer to or from memory and I/O device. When the READY signal is at low level the CPU waits for the signal to go high level. When the signal is at high level the CPU ends the read or write.
INTR	Maskable interrupt request	Input	This signal is sampled at the final clock cycle of each instruction for its level. Enable can be masked by software to inhibit interrupts. An interrupt vector of 256 types can be made using an M5L8259A.
$\overline{TEST}$	Test	Input	The CPU samples this pin while in the wait state. As the result of executing a WAIT instruction this pin is at high level. If the pin is still at high level when sampled the CPU continues to idle until it goes to low level and when that happens the CPU will resume operation.
NMI	Non-maskable interrupt request	Input	This signal is sampled during the final clock cycle of an instruction execution cycle. It is used for urgent interrupts such as power down. A type 2 interrupt is generated by this signal.
RESET	Reset	Input	This signal is used to initialized the CPU. When used it must be maintained at high level for 4 clock cycles to be effective.
CLK	Clock	Input	This signal is used for internal clocking. It is normally attached to the clock output of a M5L8284P or similar device.

Pin Description During Minimum Mode

Pin	Name	Input or output	Functional description
M/$\overline{IO}$	Data direction control	Output	This pin indicates whether the CPU is accessing memory or an I/O device at the time.
$\overline{WR}$	Write control	Output	This signal is used for timing when writing data to external memory or I/O device.
$\overline{INTA}$	Interrupt acknowledge	Output	This pin is used as the read strobe for the interrupt vector on the data bus during the interrupt acknowledge cycle.
ALE	Address latch enable	Output	This signal is the output strobe from the CPU for write address. This is output using time sharing techniques to an external latch.

DT $\overline{R}$	Data transfer control	Output	This signal indicates the direction of data transfer between the data bus buffer and an external device.
$\overline{DEN}$	Data enable	Output	This signal enables the external data bus buffer.
HOLD	Hold request	Input	When a hold request is received by the CPU it will enter the hold state and surrender control of the data bus at the end of the current instruction execution cycle.
HLDA	Hold acknowledge	Output	This signal shows that the CPU bus accepted a hold request from a peripheral device and that control of the data bus has been surrendered to the peripheral device.

Pin Description During Maximum Mode

Pin	Name	Input or Output	Function description
$\overline{S_0}$ $\overline{S_2}$, $\overline{S_1}$,	Status	Output	$\overline{S_2}$ $\overline{S_1}$ $\overline{S_0}$ 0 0 0 Interrupt acknowledge 0 0 1 Read I/O port 0 1 0 Write I/O port 0 1 1 Hold 1 0 0 Instruction fetch 1 0 1 Read memory 1 1 0 Write memory 1 1 1 Passive cycle
$\overline{RQ}/\overline{GT_0}$ $\overline{RQ}/\overline{GT_1}$	Request/Grant	Input/output	This pin is used by other local bus masters to input a hold request to the CPU and then used to output acknowledge. $\overline{RQ}/\overline{GT_0}$ has higher priority than $\overline{RQ}/\overline{GT_1}$.
$\overline{LOCK}$	Lock request	Output	This signal forbids the use of the system bus by any other system bus masters when the CPU is using the system bus.
QS_1, QS_0	Queue status	Output	The status signal is used for indicating queue operations. QS_0 QS_1 0 0 No operation 0 1 fetch first byte (operation code) of the instruction 1 0 clear the contents of queue 1 1 fetch the next byte of the instruction

INTEL 8088 CPU

Data courtesy of the Intel Corporation (UK) Ltd.

The **8088** is in effect an eight bit data bus version of the 8086. It is more economical in many ways, including that it uses ordinary eight bit wide memory.

iAPX 88/10
8-BIT HMOS MICROPROCESSOR
8088/8088-2

- **8-Bit Data Bus Interface**
- **16-Bit Internal Architecture**
- **Direct Addressing Capability to 1 Mbyte of Memory**
- **Direct Software Compatibility with iAPX 86/10 (8086 CPU)**
- **14-Word by 16-Bit Register Set with Symmetrical Operations**
- **24 Operand Addressing Modes**
- **Byte, Word, and Block Operations**
- **8-Bit and 16-Bit Signed and Unsigned Arithmetic in Binary or Decimal, Including Multiply and Divide**
- **Compatible with 8155-2, 8755A-2 and 8185-2 Multiplexed Peripherals**
- **Two Clock Rates:**
  5 MHz for 8088
  8 MHz for 8088-2
- **Available in EXPRESS**
  - Standard Temperature Range
  - Extended Temperature Range

The Intel® iAPX 88/10 is a new generation, high performance microprocessor implemented in N-channel, depletion load, silicon gate technology (HMOS), and packaged in a 40-pin CerDIP package. The processor has attributes of both 8- and 16-bit microprocessors. It is directly compatible with iAPX 86/10 software and 8080/8085 hardware and peripherals.

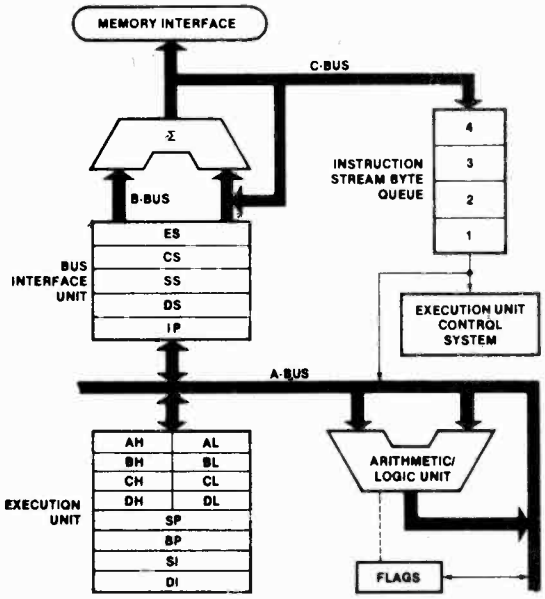


Figure 1. iAPX 88/10 CPU Functional Block Diagram

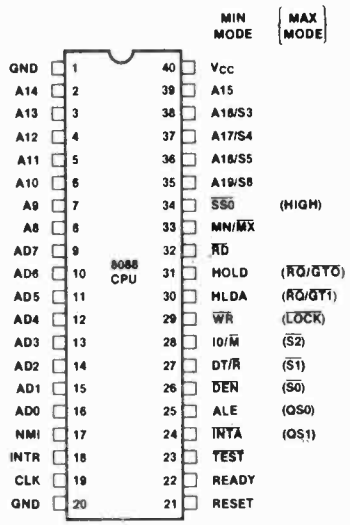


Figure 2. iAPX 88/10 Pin Configuration

Table 1. Pin Description

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

Symbol	Pin No.	Type	Name and Function
AD7–AD0	9-16	I/O	**Address Data Bus:** These lines constitute the time multiplexed memory/IO address (T1) and data (T2, T3, Tw, and T4) bus. These lines are active HIGH and float to 3-state OFF during interrupt acknowledge and local bus "hold acknowledge".
A15–A8	2-8, 39	O	**Address Bus:** These lines provide address bits 8 through 15 for the entire bus cycle (T1–T4). These lines do not have to be latched by ALE to remain valid. A15–A8 are active HIGH and float to 3-state OFF during interrupt acknowledge and local bus "hold acknowledge".
A19/S6, A18/S5, A17/S4, A16/S3	34-38	O	**Address/Status:** During T1, these are the four most significant address lines for memory operations. During I/O operations, these lines are LOW. During memory and I/O operations, status information is available on these lines during T2, T3, Tw, and T4. S6 is always low. The status of the interrupt enable flag bit (S5) is updated at the beginning of each clock cycle. S4 and S3 are encoded as shown. This information indicates which segment register is presently being used for data accessing. These lines float to 3-state OFF during local bus "hold acknowledge".
$\overline{RD}$	32	O	**Read:** Read strobe indicates that the processor is performing a memory or I/O read cycle, depending on the state of the IO/$\overline{M}$ pin or S2. This signal is used to read devices which reside on the 8088 local bus. $\overline{RD}$ is active LOW during T2, T3 and Tw of any read cycle, and is guaranteed to remain HIGH in T2 until the 8088 local bus has floated. This signal floats to 3-state OFF in "hold acknowledge".
READY	22	I	**READY:** is the acknowledgement from the addressed memory or I/O device that it will complete the data transfer. The RDY signal from memory or I/O is synchronized by the 8284 clock generator to form READY. This signal is active HIGH. The 8088 READY input is not synchronized. Correct operation is not guaranteed if the set up and hold times are not met.

S4	S3	CHARACTERISTICS
0 (LOW)	0	Alternate Data
0	1	Stack
1 (HIGH)	0	Code or None
1	1	Data
S6 is 0 (LOW)		

INTR	18	I	**Interrupt Request:** is a level triggered input which is sampled during the last clock cycle of each instruction to determine if the processor should enter into an interrupt acknowledge operation. A subroutine is vectored to via an interrupt vector lookup table located in system memory. It can be internally masked by software resetting the interrupt enable bit. INTR is internally synchronized. This signal is active HIGH.
$\overline{TEST}$	23	I	**$\overline{TEST}$:** input is examined by the "wait for test" instruction. If the $\overline{TEST}$ input is LOW, execution continues, otherwise the processor waits in an "idle" state. This input is synchronized internally during each clock cycle on the leading edge of CLK.
NMI	17	I	**Non-Maskable Interrupt:** is an edge triggered input which causes a type 2 interrupt. A subroutine is vectored to via an interrupt vector lookup table located in system memory. NMI is not maskable internally by software. A transition from a LOW to HIGH initiates the interrupt at the end of the current instruction. This input is internally synchronized.

Symbol	Pin No.	Type	Name and Function
RESET	21	I	**RESET:** causes the processor to immediately terminate its present activity. The signal must be active HIGH for at least four clock cycles. It restarts execution, as described in the instruction set description, when RESET returns LOW. RESET is internally synchronized.
CLK	19	I	**Clock:** provides the basic timing for the processor and bus controller. It is asymmetric with a 33% duty cycle to provide optimized internal timing.
V_{CC}	40		**V_{CC}:** is the +5V ±10% power supply pin.
GND	1, 20		**GND:** are the ground pins.
MN/$\overline{MX}$	33	I	**Minimum/Maximum:** indicates what mode the processor is to operate in. The two modes are discussed in the following sections.

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

IO/$\overline{M}$	28	O	**Status Line:** is an inverted maximum mode $\overline{S2}$. It is used to distinguish a memory access from an I/O access. IO/$\overline{M}$ becomes valid in the T4 preceding a bus cycle and remains valid until the final T4 of the cycle (I/O=HIGH, M=LOW). IO/$\overline{M}$ floats to 3-state OFF in local bus "hold acknowledge".
$\overline{WR}$	29	O	**Write:** strobe indicates that the processor is performing a write memory or write I/O cycle, depending on the state of the IO/$\overline{M}$ signal. WR is active for T2, T3, and Tw of any write cycle. It is active LOW, and floats to 3-state OFF in local bus "hold acknowledge".
$\overline{INTA}$	24	O	**INTA:** is used as a read strobe for interrupt acknowledge cycles. It is active LOW during T2, T3, and Tw of each interrupt acknowledge cycle.
ALE	25	O	**Address Latch Enable:** is provided by the processor to latch the address into the 8282/8283 address latch. It is a HIGH pulse active during clock low of T1 of any bus cycle. Note that ALE is never floated.
DT/$\overline{R}$	27	O	**Data Transmit/Receive:** is needed in a minimum system that desires to use an 8286/8287 data bus transceiver. It is used to control the direction of data flow through the transceiver. Logically, DT/$\overline{R}$ is equivalent to $\overline{S1}$ in the maximum mode, and its timing is the same as for IO/$\overline{M}$ (T=HIGH, R=LOW). This signal floats to 3-state OFF in local "hold acknowledge".
$\overline{DEN}$	26	O	**Data Enable:** is provided as an output enable for the 8286/8287 in a minimum system which uses the transceiver. $\overline{DEN}$ is active LOW during each memory and I/O access, and for $\overline{INTA}$ cycles. For a read or $\overline{INTA}$ cycle, it is active from the middle of T2 until the middle of T4, while for a write cycle, it is active from the beginning of T2 until the middle of T4. $\overline{DEN}$ floats to 3-state OFF during local bus "hold acknowledge".
HOLD, HLDA	30,31	I, O	**HOLD:** indicates that another master is requesting a local bus "hold". To be acknowledged, HOLD must be active HIGH. The processor receiving the "hold" request will issue HLDA (HIGH) as an acknowledgement, in the middle of a T4 or TI clock cycle. Simultaneous with the issuance of HLDA the processor will float the local bus and control lines. After HOLD is detected as being LOW, the processor lowers HLDA, and when the processor needs to run another cycle, it will again drive the local bus and control lines. Hold is not an asynchronous input. External synchronization should be provided if the system cannot otherwise guarantee the set up time.

$\overline{SSO}$	34	O	**Status line:** is logically equivalent to $\overline{S0}$ in the maximum mode. The combination of $\overline{SSO}$, IO/$\overline{M}$ and DT/$\overline{R}$ allows the system to completely decode the current bus cycle status.

IO/$\overline{M}$	DT/$\overline{R}$	$\overline{SSO}$	CHARACTERISTICS
1 (HIGH)	0	0	Interrupt Acknowledge
1	0	1	Read I/O port
1	1	0	Write I/O port
1	1	1	Halt
0 (LOW)	0	0	Code access
0	0	1	Read memory
0	1	0	Write memory
0	1	1	Passive

The following pin function descriptions are for the 8088, 8228 system in maximum mode (i.e., MN/MX = GND.) Only the pin functions which are unique to maximum mode are described; all other pin functions are as described above.

Symbol	Pin No.	Type	Name and Function
$\overline{S2}$, $\overline{S1}$, $\overline{S0}$	26-28	O	**Status:** is active during clock high of T4, T1, and T2, and is returned to the passive state (1,1,1) during T3 or during Tw when READY is HIGH. This status is used by the 8288 bus controller to generate all memory and I/O access control signals. Any change by $\overline{S2}$, $\overline{S1}$, or $\overline{S0}$ during T4 is used to indicate the beginning of a bus cycle, and the return to the passive state in T3 or Tw is used to indicate the end of a bus cycle. These signals float to 3-state OFF during "hold acknowledge". During the first clock cycle after RESET becomes active, these signals are active HIGH. After this first clock, they float to 3-state OFF.
$\overline{RQ}/\overline{GT0}$, $\overline{RQ}/\overline{GT1}$	30, 31	I/O	**Request/Grant:** pins are used by other local bus masters to force the processor to release the local bus at the end of the processor's current bus cycle. Each pin is bidirectional with $\overline{RQ}/\overline{GT0}$ having higher priority than $\overline{RQ}/\overline{GT1}$. $\overline{RQ}/\overline{GT}$ has an internal pull-up resistor, so may be left unconnected. The request/grant sequence is as follows (See Figure 8): 1. A pulse of one CLK wide from another local bus master indicates a local bus request ("hold") to the 8088 (pulse 1). 2. During a T4 or TI clock cycle, a pulse one clock wide from the 8088 to the requesting master (pulse 2), indicates that the 8088 has allowed the local bus to float and that it will enter the "hold acknowledge" state at the next CLK. The CPU's bus interface unit is disconnected logically from the local bus during "hold acknowledge". The same rules as for HOLD/HOLDA apply as for when the bus is released. 3. A pulse one CLK wide from the requesting master indicates to the 8088 (pulse 3) that the "hold" request is about to end and that the 8088 can reclaim the local bus at the next CLK. The CPU then enters T4. Each master-master exchange of the local bus is a sequence of three pulses. There must be one idle CLK cycle after each bus exchange. Pulses are active LOW. If the request is made while the CPU is performing a memory cycle, it will release the local bus during T4 of the cycle when all the following conditions are met: 1. Request occurs on or before T2. 2. Current cycle is not the low bit of a word. 3. Current cycle is not the first acknowledge of an interrupt acknowledge sequence. 4. A locked instruction is not currently executing. If the local bus is idle when the request is made the two possible events will follow: 1. Local bus will be released during the next clock. 2. A memory cycle will start within 3 clocks. Now the four rules for a currently active memory cycle apply with condition number 1 already satisfied.

$\overline{S2}$	$\overline{S1}$	$\overline{S0}$	CHARACTERISTICS
0 (LOW)	0	0	Interrupt Acknowledge
0	0	1	Read I/O port
0	1	0	Write I/O port
0	1	1	Halt
1 (HIGH)	0	0	Code access
1	0	1	Read memory
1	1	0	Write memory
1	1	1	Passive

Symbol	Pin No.	Type	Name and Function
$\overline{LOCK}$	29	O	**$\overline{LOCK}$:** indicates that other system bus masters are not to gain control of the system bus while $\overline{LOCK}$ is active (LOW). The $\overline{LOCK}$ signal is activated by the "LOCK" prefix instruction and remains active until the completion of the next instruction. This signal is active LOW, and floats to 3-state off in "hold acknowledge".

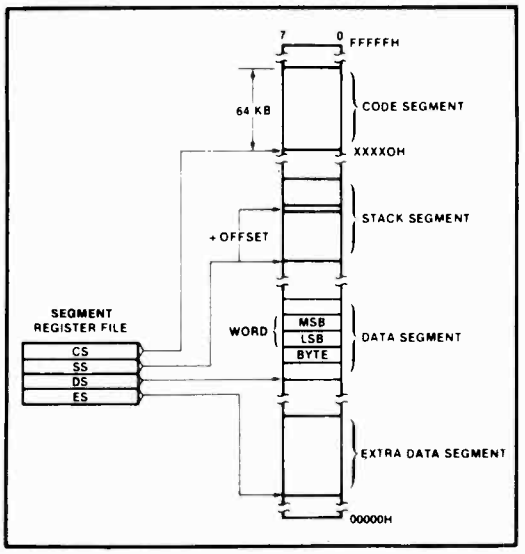
QS1, QS0	24, 25	O	**Queue Status:** provide status to allow external tracking of the internal 8088 instruction queue. The queue status is valid during the CLK cycle after which the queue operation is performed. QS1 / QS0 / CHARACTERISTICS 0 (LOW) / 0 / No operation 0 / 1 / First byte of opcode from queue 1 (HIGH) / 0 / Empty the queue 1 / 1 / Subsequent byte from queue
—	34	O	Pin 34 is always high in the maximum mode.

FUNCTIONAL DESCRIPTION

Memory Organization

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

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



Minimum and Maximum Modes

The requirements for supporting minimum and maximum 8088 systems are sufficiently different that they cannot be done efficiently with 40 uniquely defined pins. Consequently, the 8088 is equipped with a strap pin (MN/$\overline{MX}$) which defines the system configuration. The definition of a certain subset of the pins changes, dependent on the condition of the strap pin. When the MN/$\overline{MX}$ pin is strapped to GND, the 8088 defines pins 24 through 31 and 34 in maximum mode. When the MN/$\overline{MX}$ pin is strapped to V_{CC}, the 8088 generates bus control signals itself on pins 24 through 31 and 34.

The minimum mode 8088 can be used with either a multiplexed or demultiplexed bus. The multiplexed bus configuration is compatible with the MCS-85™ multiplexed bus peripherals (8155, 8156, 8355, 8755A, and 8185). This configuration (See Figure 5) provides the user with a minimum chip count system. This architecture provides the 8088 processing power in a highly integrated form.

The demultiplexed mode requires one latch (for 64K addressability) or two latches (for a full megabyte of addressing). A third latch can be used for buffering if the address bus loading requires it. An 8286 or 8287 transceiver can also be used if data bus buffering is required. (See Figure 6.) The 8088 provides $\overline{DEN}$ and DT/$\overline{R}$ to control the transceiver, and ALE to latch the addresses. This configuration of the minimum mode provides the standard demultiplexed bus structure with heavy bus buffering and relaxed bus timing requirements.

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

The 8088 Compared to the 8086

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

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

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

INTEL 8155H/8156H Programmable Peripheral Interface

Data courtesy of the Intel Corporation (UK) Ltd.

- **Single +5V Power Supply with 10% Voltage Margins**
- **30% Lower Power Consumption than the 8155 and 8156**
- **100% Compatible with 8155 and 8156**
- **256 Word x 8 Bits**
- **Completely Static Operation**
- **Internal Address Latch**
- **2 Programmable 8-Bit I/O Ports**
- **1 Programmable 6-Bit I/O Port**
- **Programmable 14-Bit Binary Counter/ Timer**
- **Compatible with 8085AH, 8085A and 8088 CPU**
- **Multiplexed Address and Data Bus**

The Intel® 8155H and 8156H are RAM and I/O chips implemented in N-Channel, depletion load, silicon gate technology (HMOS), to be used in the 8085AH and 8088 microprocessor systems. The RAM portion is designed with 2048 static cells organized as 256 x 8. They have a maximum access time of 400 ns to permit use with no wait states in 8085AH CPU. The 8155H-2 and 8156H-2 have maximum access times of 330 ns for use with the 8085AH-2 and the 5 MHz 8088 CPU.

The I/O portion consists of three general purpose I/O ports. One of the three ports can be programmed to be status pins, thus allowing the other two ports to operate in handshake mode.

A 14-bit programmable counter/timer is also included on chip to provide either a square wave or terminal count pulse for the CPU system depending on timer mode.

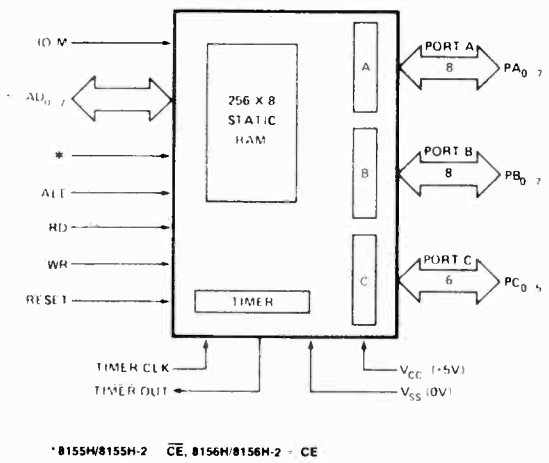


Figure 1. Block Diagram

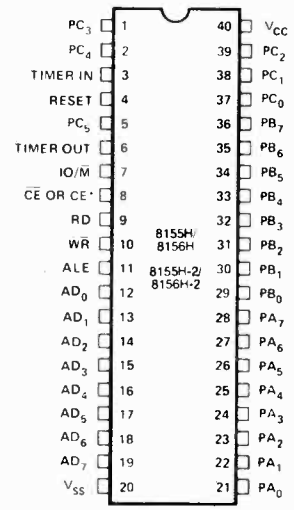


Figure 2. Pin Configuration

INPUT/OUTPUT SECTION

The I/O section of the 8155H/8156H consists of five registers: (See Figure 7.)

- **Command/Status Register (C/S)** — Both registers are assigned the address XXXXX000. The C/S address serves the dual purpose.

  When the C/S registers are selected during WRITE operation, a command is written into the command register. The contents of this register are *not* accessible through the pins.

  When the C/S (XXXXX000) is selected during a READ operation, the status information of the I/O ports and the timer becomes available on the AD_{0-7} lines.

- **PA Register** — This register can be programmed to be either input or output ports depending on the status of the contents of the C/S Register. Also depending on the command, this port can operate in either the basic mode or the strobed mode (See timing diagram). The I/O pins assigned in relation to this register are PA_{0-7}. The address of this register is XXXXX001.

- **PB Register** — This register functions the same as PA Register. The I/O pins assigned are PB_{0-7}. The address of this register is XXXXX010.

- **PC Register** — This register has the address XXXXX011 and contains only 6 bits. The 6 bits can be programmed to be either input ports, output ports or as control

signals for PA and PB by properly programming the AD_2 and AD_3 bits of the C/S register.

When PC_{0-5} is used as a control port, 3 bits are assigned for Port A and 3 for Port B. The first bit is an interrupt that the 8155H sends out. The second is an output signal indicating whether the buffer is full or empty, and the third is an input pin to accept a strobe for the strobed input mode. (See Table 2.)

Note in the diagram that when the I/O ports are programmed to be output ports, the contents of the output ports can still be read by a READ operation when appropriately addressed.

The outputs of the 8155H/8156H are "glitch-free" meaning that you can write a "1" to a bit position that was previously "1" and the level at the output pin will not change.

TIMER SECTION

The timer is a 14-bit down-counter that counts the TIMER IN pulses and provides either a square wave or pulse when terminal count (TC) is reached.

Note that while the counter is counting, you may load a new count and mode into the count length registers. Before the new count and mode will be used by the counter, you must issue a START command to the counter. This applies even though you may only want to change the count and use the previous mode.

The counter in the 8155H is not initialized to any particular mode or count when hardware RESET occurs, but RESET does *stop* the counting. Therefore, counting cannot begin following RESET until a START command is issued via the C/S register.

Please note that the timer circuit on the 8155H/8156H chip is designed to be a square-wave timer, not an event counter. To achieve this, it counts down by twos twice in completing one cycle. Thus, its registers do not contain values directly representing the number of TIMER IN pulses received. You cannot load an initial value of 1 into the count register and cause the timer to operate, as its terminal count value is 10 (binary) or 2 (decimal). (For the detection of single pulses, it is suggested that one of the hardware interrupt pins on the 8085AH be used.) After the timer has started counting down, the values residing in the count registers can be used to calculate the actual number of TIMER IN pulses required to complete the timer cycle if desired. To obtain the remaining count, perform the following operations in order:

1. Stop the count
2. Read in the 16-bit value from the count length registers
3. Reset the upper two mode bits
4. Reset the carry and rotate right one position all 16 bits through carry
5. If carry is set, add 1/2 of the full original count (1/2 full count − 1 if full count is odd).

INTEL 8185 256 x 8 STATIC RAM

Data courtesy of the Intel Corporation (UK) Ltd.

- **Multiplexed Address and Data Bus**
- **Directly Compatible with 8085A and iAPX 88 Microprocessors**
- **Low Operating Power Dissipation**
- **Low Standby Power Dissipation**
- **Single +5V Supply**
- **High Density 18-Pin Package**

The Intel® 8185 is an 8192-bit static random access memory (RAM) organized as 1024 words by 8-bits using N-channel Silicon-Gate MOS technology. The multiplexed address and data bus allows the 8185 to interface directly to the 8085A and iAPX 88 microprocessors to provide a maximum level of system integration.

The low standby power dissipation minimizes system power requirements when the 8185 is disabled.

The 8185-2 is a high-speed selected version of the 8185 that is compatible with the 5 MHz 8085A-2 and the 5 MHz iAPX 88.

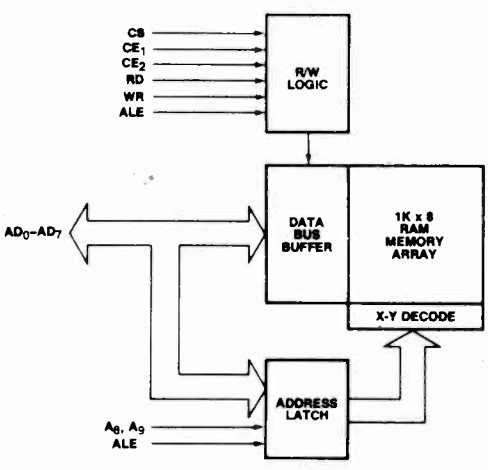
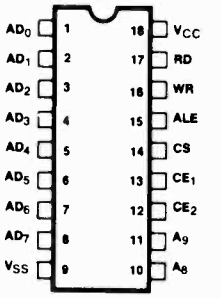


Figure 1. Block Diagram



AD_0–AD_7	ADDRESS/DATA LINES
A_8, A_9	ADDRESS LINES
CS	CHIP SELECT
CE_1	CHIP ENABLE (IO/M)
CE_2	CHIP ENABLE
ALE	ADDRESS LATCH ENABLE
WR	WRITE ENABLE

INTEL 8254 Interval Timer/Counter.

Data courtesy of the Intel Corporation (UK) Ltd.

- **Compatible with Most Microprocessors Including 8080A, 8085A, iAPX 88 and iAPX 86**
- **Handles Inputs from DC to 8 MHz (10 MHz for 8254-2)**
- **Six Programmable Counter Modes**
- **Three Independent 16-bit Counters**
- **Binary or BCD Counting**
- **Status Read-Back Command**

The Intel® 8254 is a counter/timer device designed to solve the common timing control problems in microcomputer system design. It provides three independent 16-bit counters, each capable of handling clock inputs up to 10 MHz. All modes are software programmable. The 8254 is a superset of the 8253.

The 8254 uses HMOS technology and comes in a 24-pin plastic or CERDIP package.

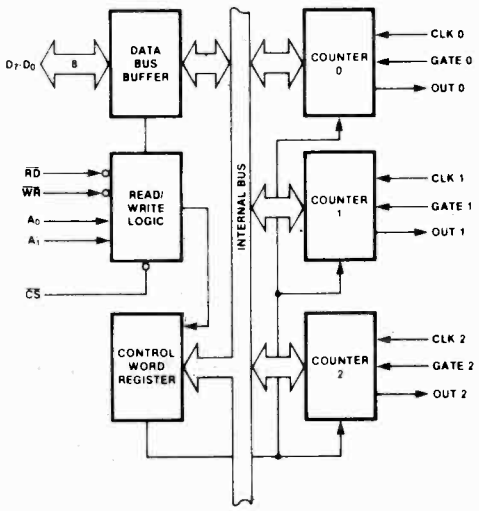


Figure 1. 8254 Block Diagram

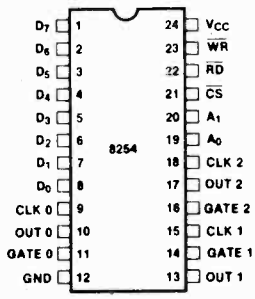


Figure 2. Pin Configuration

The 8254 is a programmable interval timer/counter designed for use with Intel 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 8254 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 8254 to match his requirements and programs one of the counters for the desired delay. After the desired delay, the 8254 will interrupt the CPU. Software overhead is minimal and variable length delays can easily be accommodated.

Some of the other counter/timer functions common to microcomputers which can be implemented with the 8254 are:

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

INTEL 8255A I/O PORT

Data courtesy of the Intel Corporation (UK) Ltd.

- MCS-85™ Compatible 8255A-5
- 24 Programmable I/O Pins
- Completely TTL Compatible
- Fully Compatible with Intel® Microprocessor Families
- Direct Bit Set/Reset Capability Easing Control Application Interface
- Reduces System Package Count
- Improved DC Driving Capability
- Improved Timing Characteristics

The Intel® 8255A is a general purpose programmable I/O device designed for use with Intel® microprocessors. It has 24 I/O pins which may be individually programmed in 2 groups of 12 and used in 3 major modes of operation. In the first mode (MODE 0), each group of 12 I/O pins may be programmed in sets of 4 to be input or output. In MODE 1, the second mode, each group may be programmed to have 8 lines of input or output. Of the remaining 4 pins, 3 are used for handshaking and interrupt control signals. The third mode of operation (MODE 2) is a bidirectional bus mode which uses 8 lines for a bidirectional bus, and 5 lines, borrowing one from the other group, for handshaking.

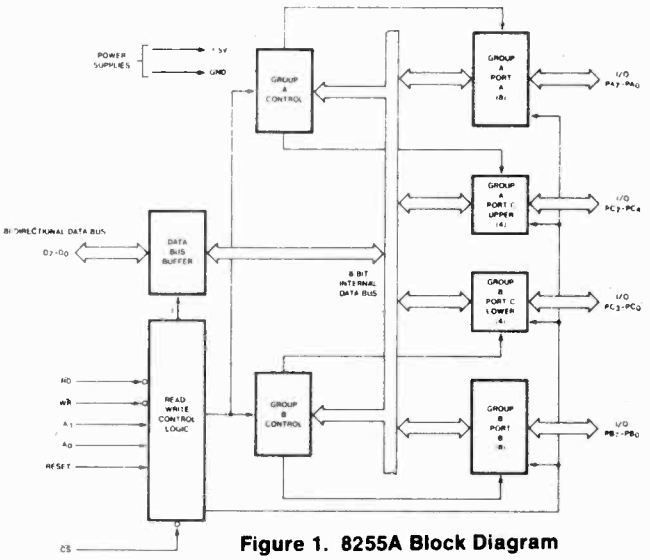


Figure 1. 8255A Block Diagram

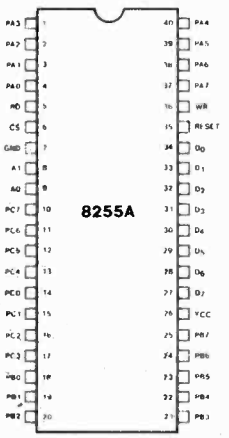


Figure 2. Pin Configuration

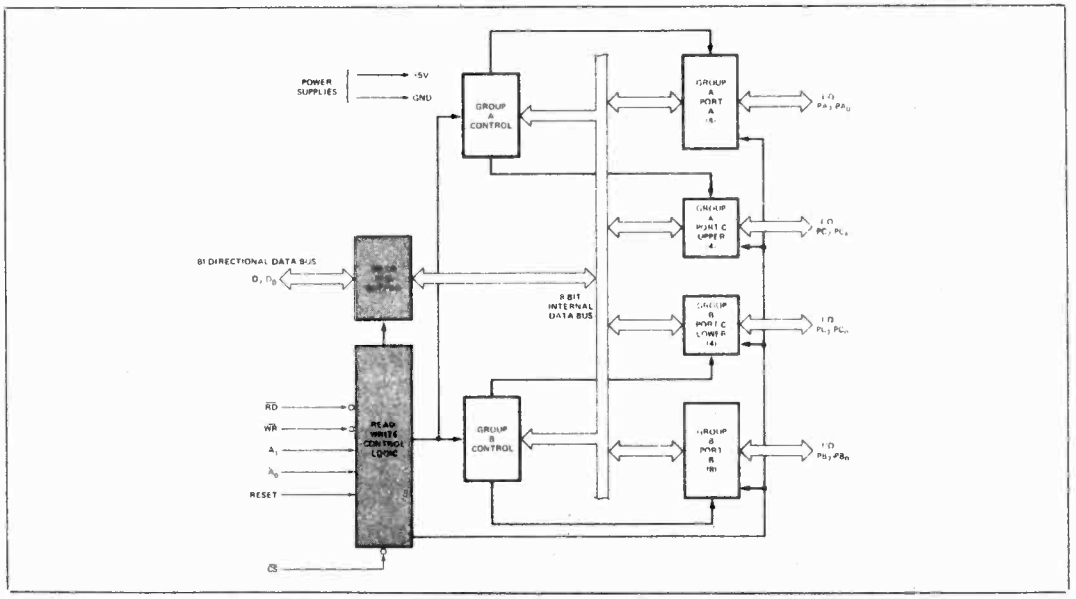


Figure 3. 8255A Block Diagram Showing Data Bus Buffer and Read/Write Control Logic Functions

8255A I/O Port

8255A FUNCTIONAL DESCRIPTION

The 8255A is a programmable peripheral interface (PPI) device designed for use in Intel® microcomputer systems. Its function is that of a general purpose I/O component to interface peripheral equipment to the microcomputer system bus. The functional configuration of the 8255A is programmed by the system software so that normally no external logic is necessary to interface peripheral devices or structures.

8255A OPERATIONAL DESCRIPTION

Mode Selection

There are three basic modes of operation that 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 (i.e., all 24 lines will be in the high impedance state). After the reset is removed the 8255A can remain in the input mode with no additional initialization required. During the execution of the system program any of the other modes may be selected using a single output instruction. This allows a single 8255A to service a variety of peripheral devices with a simple software maintenance routine.

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.

APPLICATIONS OF THE 8255A

The 8255A is a very powerful tool for interfacing peripheral equipment to the microcomputer system. It represents the optimum use of available pins and is 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 8255A 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 8255A to exactly "fit" the application.

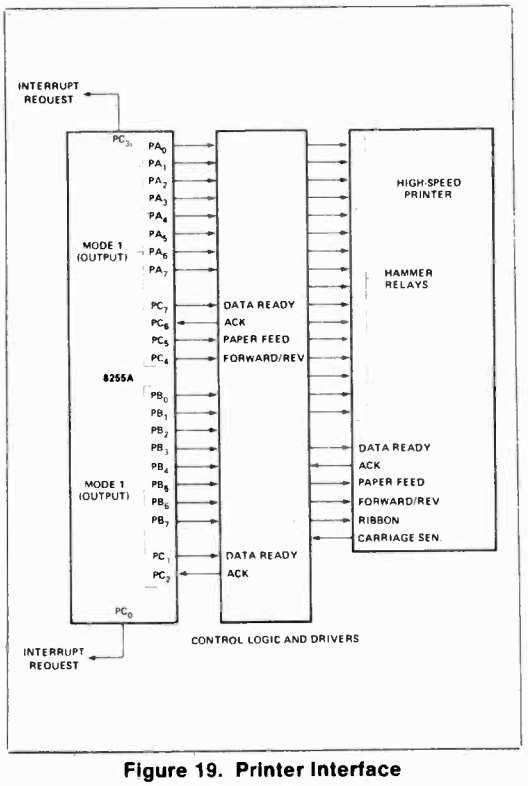


Figure 19. Printer Interface

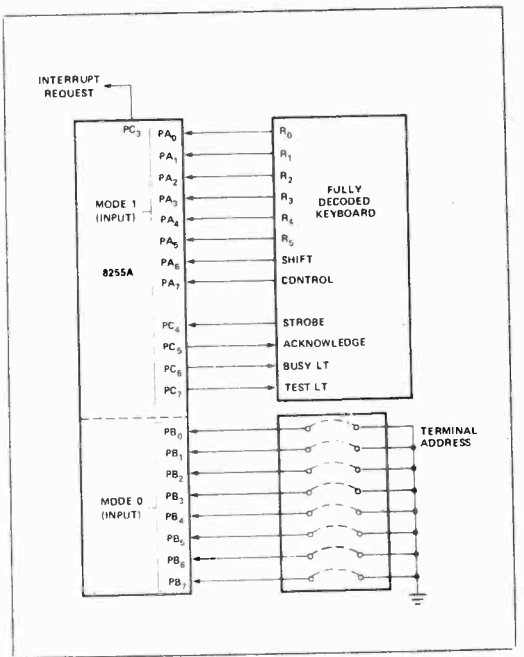


Figure 21. Keyboard and Terminal Address Interface

INTEL 8755A EPROM/I/O PORT

Data courtesy of the Intel Corporation (UK) Ltd.

- **2048 Words x 8 Bits**
- **Single +5V Power Supply (V_{CC})**
- **Directly Compatible with 8085A and 8088 Microprocessors**
- **U.V. Erasable and Electrically Reprogrammable**
- **2 General Purpose 8-Bit I/O Ports**
- **Each I/O Port Line Individually Programmable as Input or Output**
- **Multiplexed Address and Data Bus**
- **Internal Address Latch**

The Intel® 8755A is an erasable and electrically reprogrammable ROM (EPROM) and I/O chip to be used in the 8085A and iAPX 88 microprocessor systems. The EPROM portion is organized as 2048 words by 8 bits. It has a maximum access time of 450 ns to permit use with no wait states in an 8085A CPU.

The I/O portion consists of 2 general purpose I/O ports. Each I/O port has 8 port lines, and each I/O port line is individually programmable as input or output.

The 8755A-2 is a high speed selected version of the 8755A compatible with the 5 MHz 8085A-2 and the 5 MHz iAPX 88 microprocessor.

Figure 1. Block Diagram

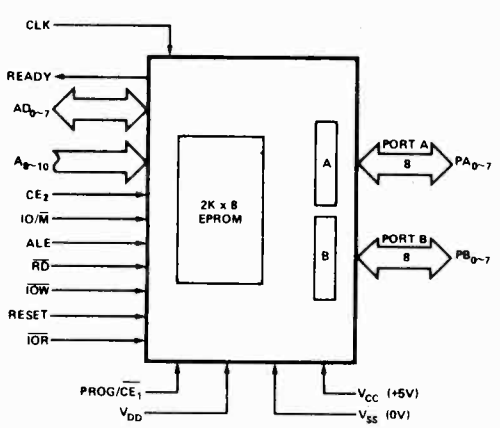
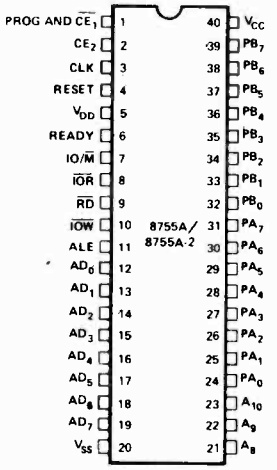


Figure 2. Pin Configuration



ERASURE CHARACTERISTICS

The erasure characteristics of the 8755A are such that erasure begins to occur when exposed to light with wavelengths shorter than approximately 4000 Angstroms (Å). It should be noted that sunlight and certain types of fluorescent lamps have wavelengths in the 3000-4000Å range. Data show that constant exposure to room level fluorescent lighting could erase the typical 8755A in approximately 3 years while it would take approximately 1 week to cause erasure when exposed to direct sunlight. If the 8755A is to be exposed to these types of lighting conditions for extended periods of time, opaque labels are available from Intel which should be placed over the 8755 window to prevent unintentional erasure.

The recommended erasure procedure for the 8755A is exposure to shortwave ultraviolet light which has a wavelength of 2537 Angstroms (Å). The integrated dose (i.e., UV intensity X exposure time) for erasure should be a minimum of 15W-sec/cm^2. The erasure time with this dosage is approximately 15 to 20 minutes using an ultraviolet lamp with a 12000μW/cm^2 power rating. The 8755A should be placed within one inch from the lamp tubes during erasure. Some lamps have a filter on their tubes and this filter should be removed before erasure.

PROGRAMMING

Initially, and after each erasure, all bits of the EPROM portions of the 8755A are in the "1" state. Information is introduced by selectively programming "0" into the desired bit locations. A programmed "0" can only be changed to a "1" by UV erasure.

The program mode itself consists of programming a single address at a time, giving a single 50 msec pulse for every address. Generally, it is desirable to have a verify cycle after a program cycle for the same address as shown in the attached timing diagram. In the verify cycle (i.e., normal memory read cycle) 'V_{DD}' should be at +5V.

MITSUBISHI M5L 8212P I/O

Data courtesy of Mitsubishi Electric (UK) Ltd.

DESCRIPTION

The M5L 8212P is an input/output port consisting of an 8-bit latch with 3-state output buffers along with control and device selection logic. Also included is a service request flip-flop for the generation and control of interrupts to a microprocessor. It is fabricated using bipolar Schottky TTL technology.

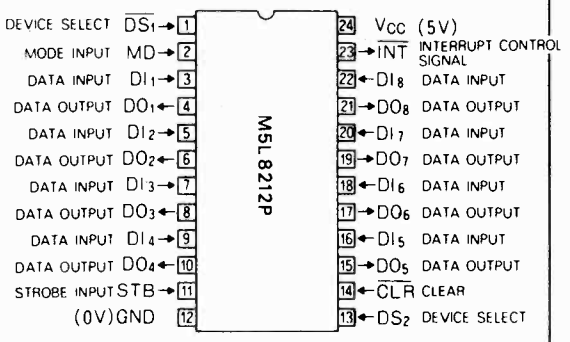
FEATURES

- Parallel 8-bit data register and buffer
- Service request flip-flop for interrupt generation
- Three-state outputs
- Low input load current: I_{IL} = absolute -250μA (max)
- High output sink current: I_{OL} = 16mA (max)
- High-level output voltage for direct interface to a M5L 8080AP, S CPU: V_{OH} = 3.65V (min)
- Interchangeable with Intel's 8212 in terms of electrical characteristics and pin configuration

APPLICATIONS

- Input/output port for a M5L 8080AP, S
- Latches, gate buffers or multiplexers
- Peripheral and input/output functions for microcomputer systems

PIN CONFIGURATION (TOP VIEW)



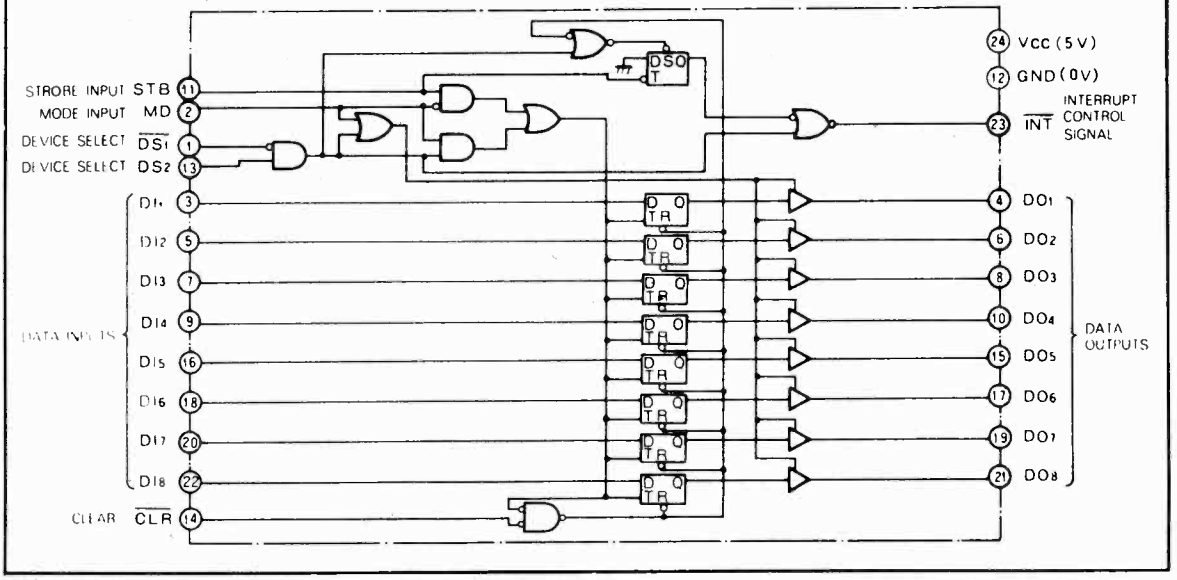
Outline 24P1

FUNCTION

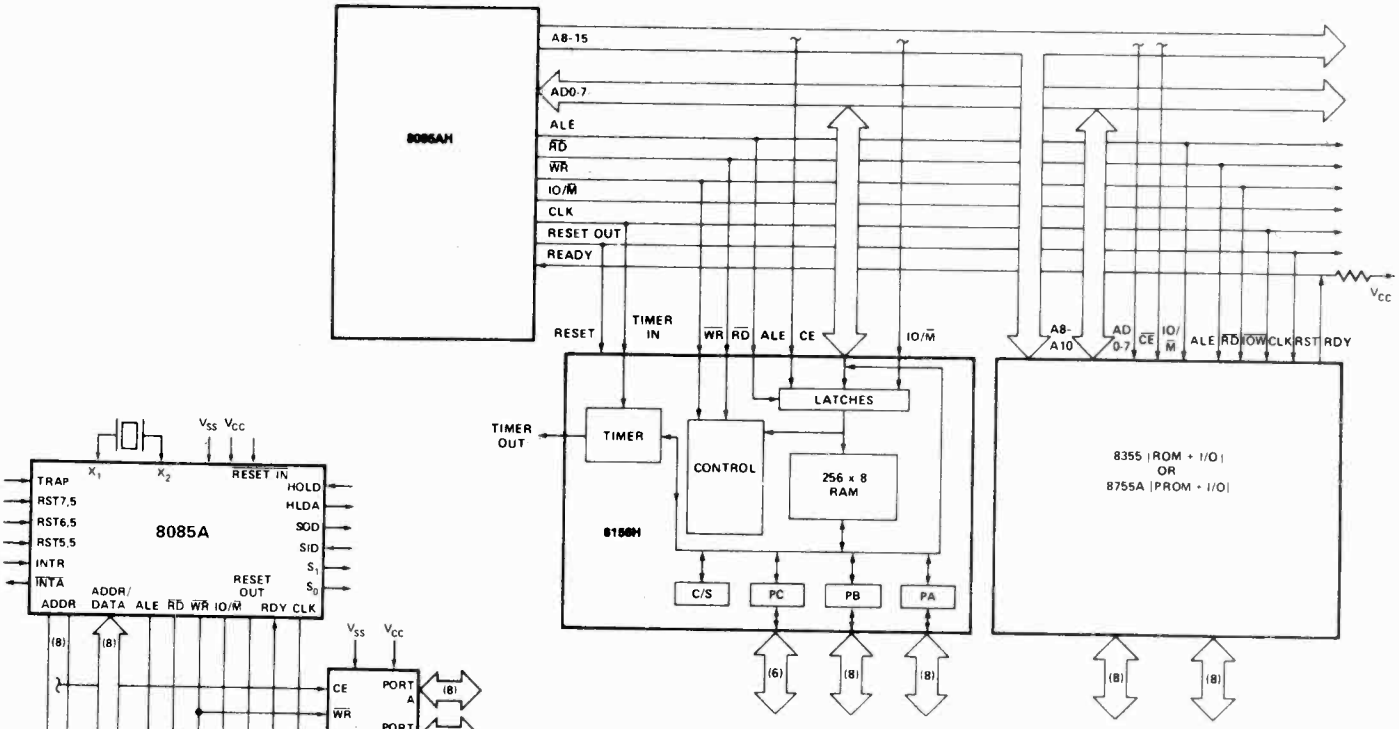
Device select 1 ($\overline{DS_1}$) and device select 2 (DS_2) are used for chip selection when the mode input MD is low. When $\overline{DS_1}$ is low and DS_2 is high, the data in the latches is transferred to the data outputs DO_1~DO_8; and the service request flip-flop SR is set. Also, the strobed input STB is active, the data inputs DI_1~DI_8 are latched in the data latches, and the service request flip-flop SR is reset.

When MD is high, the data in the data latches is transferred to the data outputs. When $\overline{DS_1}$ is low and DS_2 is high, the data inputs are latched in the data latches. The low-level clear input $\overline{CLR}$ resets the data latches and sets the service request flip-flop SR, but the state of the output buffers is not changed.

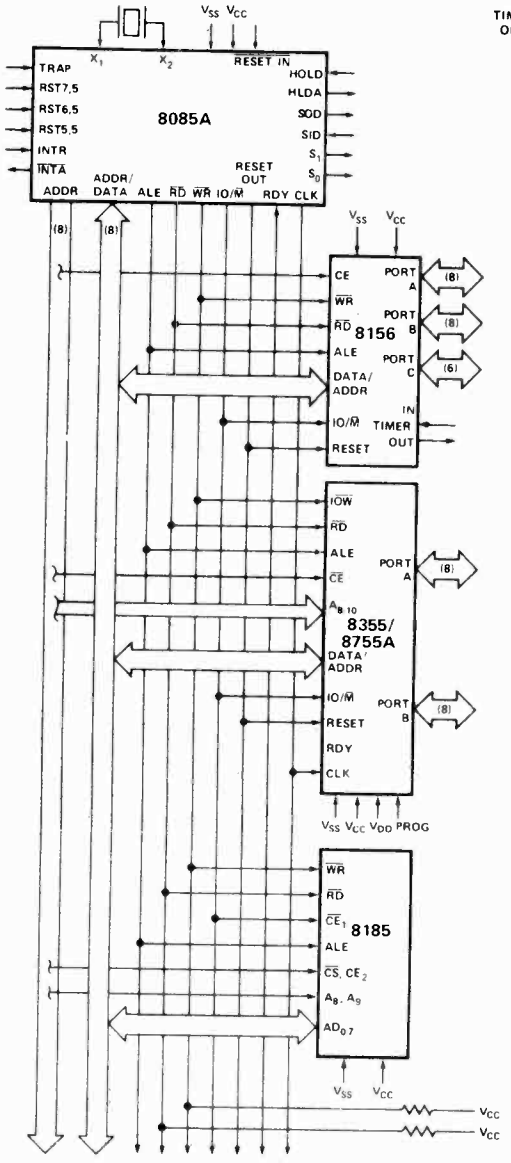
BLOCK DIAGRAM



8085 Additional Systems Diagrams



8085AH Minimum System Configuration (Memory Mapped I/O)



8185 in an MCS-85 System

SYSTEM APPLICATIONS

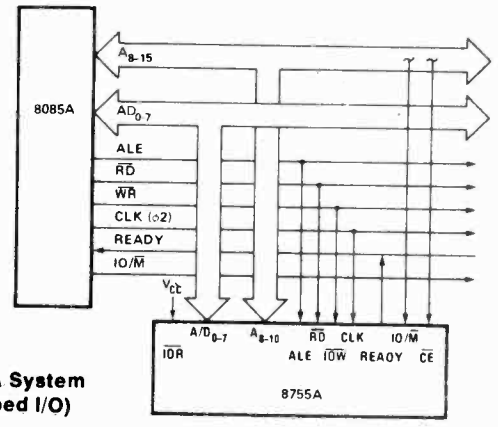
System Interface with 8085A and iAPX 88

A system using the 8755A can use either one of the two I/O Interface techniques:

- Standard I/O
- Memory Mapped I/O

If a standard I/O technique is used, the system can use the feature of both CE_2 and $\overline{CE_1}$. By using a combination of unused address lines A_{11-15} and the Chip Enable inputs, the 8085A system can use up to 5 each 8755A's without requiring a CE decoder. See Figure 4a and 4b.

If a memory mapped I/O approach is used the 8755A will be selected by the combination of both the Chip Enables and IO/$\overline{M}$ using AD_{8-15} address lines.



8755A in 8085A System (Memory-Mapped I/O)

Zilog Z8400 CPU (Z80 Family)

Data courtesy of Zilog (UK) Ltd.

The Z80 is a powerful and popular microprocessor whose instruction set is a superset of the 8080. One of the strong points of the instruction set is the range of block operations on memory, eg. block transfers.

On the hardware front, the Z80 CPU has a dynamic RAM refresh timer, so cheap dynamic RAMS can easily be used when a large memory is required.

The Z80 is used in several microprocessors, and Z80 plug-in cards are available for the popular Apple II.

Z80 Microprocessor Family

The Zilog Z80 microprocessor is the central element of a comprehensive microprocessor product family. This family works together in most applications with minimum requirements for additional logic, facilitating the design of efficient and cost-effective microcomputer-based systems.

Zilog has designed five components to provide extensive support for the Z80 microprocessor. These are:

- The PIO (Parallel Input/Output) operates in both data-byte I/O transfer mode (with handshaking) and in bit mode (without handshaking). The PIO may be configured to interface with standard parallel peripheral devices such as printers, tape punches, and keyboards.
- The CTC (Counter/Timer Circuit) features four programmable 8-bit counter/timers, each of which has an 8-bit prescaler. Each of the four channels may be configured to operate in either counter or timer mode.
- The DMA (Direct Memory Access) controller provides dual port data transfer operations and the ability to terminate data transfer as a result of a pattern match.
- The SIO (Serial Input/Output) controller offers two channels. It is capable of operating in a variety of programmable modes for both synchronous and asynchronous communication, including Bi-Sync and SDLC.
- The DART (Dual Asynchronous Receiver/Transmitter) device provides low cost asynchronous serial communication. It has two channels and a full modem control interface.

General Description

The Z80, Z80A, and Z80B CPUs are third-generation single-chip microprocessors with exceptional computational power. They offer higher system throughput and more efficient memory utilization than comparable second- and third-generation microprocessors. The internal registers contain 208 bits of read/write memory that are accessible to the programmer. These registers include two sets of six general-purpose registers which may be used individually as either 8-bit registers or as 16-bit register pairs. In addition, there are two sets of accumulator and flag registers. A group of "Exchange" instructions makes either set of main or alternate registers accessible to the programmer. The alternate set allows operation in foreground-background mode or it may be reserved for very fast interrupt response.

The Z80 also contains a Stack Pointer, Program Counter, two index registers, a Refresh register (counter), and an Interrupt register. The CPU is easy to incorporate into a system since it requires only a single +5 V power source. All output signals are fully decoded and timed to control standard memory or peripheral circuits, and it is supported by an extensive family of peripheral controllers. The internal block diagram (Figure 3) shows the primary functions of the Z80 processors. Subsequent text provides more detail on the Z80 I/O controller family, registers, instruction set, interrupts and daisy chaining, and CPU timing.

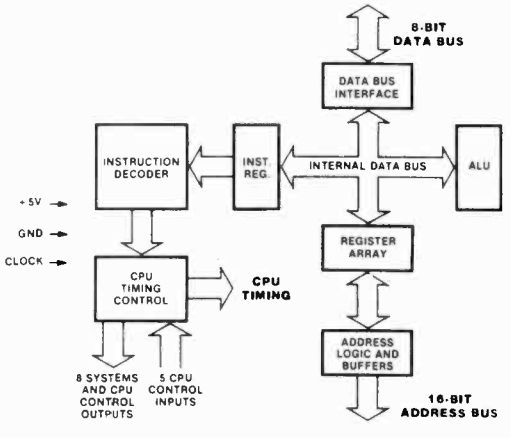


Figure 3. Z80 CPU Block Diagram

Features

- The instruction set contains 158 instructions. The 78 instructions of the 8080A are included as a subset; 8080A software compatibility is maintained.
- Six MHz, 4 MHz and 2.5 MHz clocks for the Z80B, Z80A, and Z80 CPU result in rapid instruction execution with consequent high data throughput.
- The extensive instruction set includes string, bit, byte, and word operations. Block searches and block transfers together with indexed and relative addressing result in the most powerful data handling capabilities in the microcomputer industry.
- The Z80 microprocessors and associated family of peripheral controllers are linked by a vectored interrupt system. This system may be daisy-chained to allow implementation of a priority interrupt scheme. Little, if any, additional logic is required for daisy-chaining.
- Duplicate sets of both general-purpose and flag registers are provided, easing the design and operation of system software through single-context switching, background-foreground programming, and single-level interrupt processing. In addition, two 16-bit index registers facilitate program processing of tables and arrays.
- There are three modes of high speed interrupt processing: 8080 compatible, non-Z80 peripheral device, and Z80 Family peripheral with or without daisy chain.
- On-chip dynamic memory refresh counter.

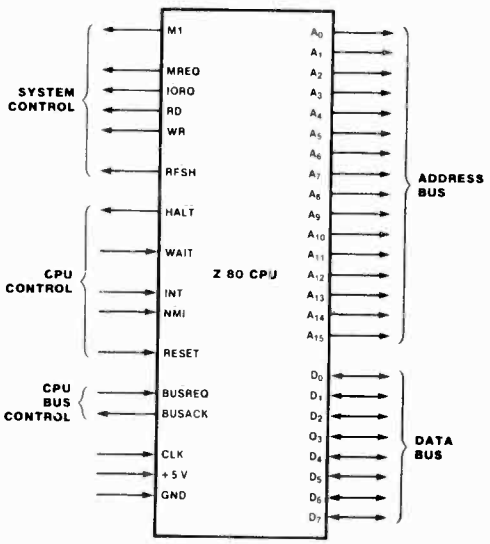


Figure 1. Pin Functions

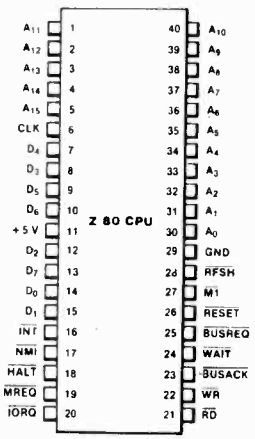


Figure 2. Pin Assignments

Z80 CPU Registers

Figure 4 shows three groups of registers within the Z80 CPU. The first group consists of duplicate sets of 8-bit registers: a principal set and an alternate set (designated by ' [prime], e.g., A'). Both sets consist of the Accumulator Register, the Flag Register, and six general-purpose registers. Transfer of data between these duplicate sets of registers is accomplished by use of "Exchange" instructions. The result is faster response to interrupts and easy, efficient implementation of such versatile programming techniques as background-foreground data processing. The second set of registers consists of six registers with assigned functions. These are the I (Interrupt Register), the R (Refresh Register), the IX and IY (Index Registers), the SP (Stack Pointer), and the PC (Program Counter). The third group consists of two interrupt status flip-flops, plus an additional pair of flip-flops which assists in identifying the interrupt mode at any particular time. Table 1 provides further information on these registers.

MAIN REGISTER SET		ALTERNATE REGISTER SET	
A ACCUMULATOR	F FLAG REGISTER	A' ACCUMULATOR	F' FLAG REGISTER
B GENERAL PURPOSE	C GENERAL PURPOSE	B' GENERAL PURPOSE	C' GENERAL PURPOSE
D GENERAL PURPOSE	E GENERAL PURPOSE	D' GENERAL PURPOSE	E' GENERAL PURPOSE
H GENERAL PURPOSE	L GENERAL PURPOSE	H' GENERAL PURPOSE	L' GENERAL PURPOSE

8 BITS

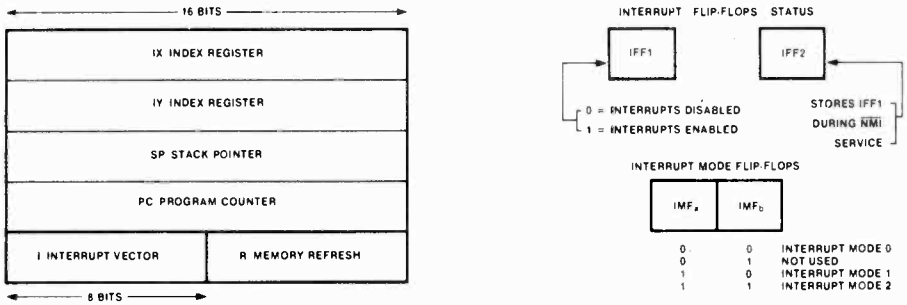


Figure 4. CPU Registers

Z80 CPU Registers (Continued)

Register		Size (Bits)	Remarks
A, A'	Accumulator	8	Stores an operand or the results of an operation.
F, F'	Flags	8	See Instruction Set.
B, B'	General Purpose	8	Can be used separately or as a 16-bit register with C.
C, C'	General Purpose	8	See B, above.
D, D'	General Purpose	8	Can be used separately or as a 16-bit register with E.
E, E'	General Purpose	8	See D, above.
H, H'	General Purpose	8	Can be used separately or as a 16-bit register with L.
L, L'	General Purpose	8	See H, above.
			Note: The (B,C), (D,E), and (H,L) sets are combined as follows: B — High byte C — Low byte D — High byte E — Low byte H — High byte L — Low byte
I	Interrupt Register	8	Stores upper eight bits of memory address for vectored interrupt processing.
R	Refresh Register	8	Provides user-transparent dynamic memory refresh. Automatically incremented and placed on the address bus during each instruction fetch cycle.
IX	Index Register	16	Used for indexed addressing.
IY	Index Register	16	Same as IX, above.
SP	Stack Pointer	16	Holds address of the top of the stack. See Push or Pop in instruction set.
PC	Program Counter	16	Holds address of next instruction.
IFF_1-IFF_2	Interrupt Enable	Flip-Flops	Set or reset to indicate interrupt status (see Figure 4).
IMFa-IMFb	Interrupt Mode	Flip-Flops	Reflect Interrupt mode (see Figure 4).

Table 1. Z80 CPU Registers

Interrupts: General Operation

The CPU accepts two interrupt input signals: $\overline{NMI}$ and $\overline{INT}$. The $\overline{NMI}$ is a non-maskable interrupt and has the highest priority. $\overline{INT}$ is a lower priority interrupt and it requires that interrupts be enabled in software in order to operate. $\overline{INT}$ can be connected to multiple peripheral devices in a wired-OR configuration.

The Z80 has a single response mode for interrupt service for the non-maskable interrupt. The maskable interrupt, $\overline{INT}$, has three programmable response modes available. These are:

- Mode 0 — compatible with the 8080 microprocessor.
- Mode 1 — Peripheral Interrupt service, for use with non-8080/Z80 systems.
- Mode 2 — a vectored interrupt scheme, usually daisy-chained, for use with Z80 Family and compatible peripheral devices.

The CPU services interrupts by sampling the $\overline{NMI}$ and $\overline{INT}$ signals at the rising edge of the last clock of an instruction. Further interrupt service processing depends upon the type of interrupt that was detected.

Non-Maskable Interrupt ($\overline{NMI}$). The non-maskable interrupt cannot be disabled by program control and therefore will be accepted at all times by the CPU. $\overline{NMI}$ is usually reserved for servicing only the highest priority type interrupts, such as that for orderly shutdown after power failure has been detected. After recognition of the $\overline{NMI}$ signal (providing $\overline{BUSREQ}$ is not active), the CPU jumps to restart location 0066H. Normally, software starting at this address contains the interrupt service routing.

Maskable Interrupt ($\overline{INT}$). Regardless of the interrupt mode set by the user, the Z80 response to a maskable interrupt input follows a common timing cycle. After the interrupt has been detected by the CPU (provided that interrupts are enabled and $\overline{BUSREQ}$ is not active) a special interrupt processing cycle begins. This is a special fetch ($\overline{M1}$) cycle in which $\overline{IORQ}$ becomes active rather than $\overline{MREQ}$, as in normal $\overline{M1}$ cycle. In addition, this special $\overline{M1}$ cycle is automatically extended by two $\overline{WAIT}$ states, to allow for the time required to acknowledge the interrupt request.

Mode 0 Interrupt Operation. This mode is compatible with the 8080 microprocessor interrupt service procedures. The interrupting device places an instruction on the data bus. This is normally a Restart Instruction, which will initiate a call to the selected one of eight restart locations in page zero of memory.

Mode 1 Interrupt Operation. Mode 1 operation is very similar to that for the $\overline{NMI}$. The principal difference is that the Mode I interrupt has a restart location of 0038H only.

Mode 2 Interrupt Operation. This interrupt mode has been designed to utilize most effectively the capabilities of the Z80 microprocessor and its associated peripheral family. The interrupting peripheral device selects the starting address of the interrupt service routine. It does this by placing an 8-bit vector on the data bus during the interrupt acknowledge cycle. The CPU forms a pointer using this byte as the lower 8-bits and the contents of the I register as the upper 8-bits. This points to an entry in a table of addresses for interrupt service routines. The CPU then jumps to the routine at that address. This flexibility in selecting the interrupt service routine address allows the peripheral device to use several different types of service routines. These routines may be located at any available location in memory. Since the interrupting device supplies the low-order byte of the 2-byte vector, bit 0 (A_0) must be a zero.

Interrupt Priority (Daisy Chaining and Nested Interrupts). The interrupt priority of each peripheral device is determined by its physical location within a daisy-chain configuration. Each device in the chain has an interrupt enable input line (IEI) and an interrupt enable output line (IEO), which is fed to the next lower priority device. The first device in the daisy chain has its IEI input hardwired to a High level. The first device has highest priority, while each succeeding device has a corresponding lower priority. This arrangement permits the CPU to select the highest priority interrupt from several simultaneously interrupting peripherals.

The interrupting device disables its IEO line to the next lower priority peripheral until it has been serviced. After servicing, its IEO line is raised, allowing lower priority peripherals to demand interrupt servicing.

The Z80 CPU will nest (queue) any pending interrupts or interrupts received while a selected peripheral is being serviced.

The Z80 microprocessor has one of the most powerful and versatile instruction sets available in any 8-bit microprocessor. It includes such unique operations as a block move for fast, efficient data transfers within memory or between memory and I/O. It also allows operations on any bit in any location in memory.

The following is a summary of the Z80 instruction set and shows the assembly language mnemonic, the operation, the flag status, and gives comments on each instruction. The *Z80 CPU Technical Manual* (03-0029-01) and *Assembly Language Programming Manual* (03-0002-01) contain significantly more details for programming use.

The instructions are divided into the following categories:

- 8-bit loads
- 16-bit loads
- Exchanges, block transfers, and searches
- 8-bit arithmetic and logic operations
- General-purpose arithmetic and CPU control
- 16-bit arithmetic operations
- Rotates and shifts
- Bit set, reset, and test operations
- Jumps
- Calls, returns, and restarts
- Input and output operations

A variety of addressing modes are implemented to permit efficient and fast data transfer between various registers, memory locations, and input/output devices. These addressing modes include:

- Immediate
- Immediate extended
- Modified page zero
- Relative
- Extended
- Indexed
- Register
- Register indirect
- Implied
- Bit

ZILOG Z8420 PIO

Data courtesy of Zilog (UK) Ltd.

Features

- Provides a direct interface between Z-80 microcomputer systems and peripheral devices.
- Both ports have interrupt-driven handshake for fast response.
- Four programmable operating modes: byte input, byte output, byte input/output (Port A only), and bit input/output.
- Programmable interrupts on peripheral status conditions.
- Standard Z-80 Family bus-request and prioritized interrupt-request daisy chains implemented without external logic.
- The eight Port B outputs can drive Darlington transistors (1.5 mA at 1.5 V).

General Description

The Z-80 PIO Parallel I/O Circuit is a programmable, dual-port device that provides a TTL-compatible interface between peripheral devices and the Z-80 CPU. The CPU configures the Z-80 PIO to interface with a wide range of peripheral devices with no other external logic. Typical peripheral devices that are compatible with the Z-80 PIO include most keyboards, paper tape readers and punches, printers, PROM programmers, etc.

One characteristic of the Z-80 peripheral controllers that separates them from other interface controllers is that all data transfer between the peripheral device and the CPU is accomplished under interrupt control. Thus, the interrupt logic of the PIO permits full use of the efficient interrupt capabilities of the Z-80 CPU during I/O transfers. All logic necessary to implement a fully nested interrupt structure is included in the PIO.

Another feature of the PIO is the ability to interrupt the CPU upon occurrence of specified status conditions in the peripheral device. For example, the PIO can be programmed to interrupt if any specified peripheral alarm conditions should occur. This interrupt capability reduces the time the processor must spend in polling peripheral status.

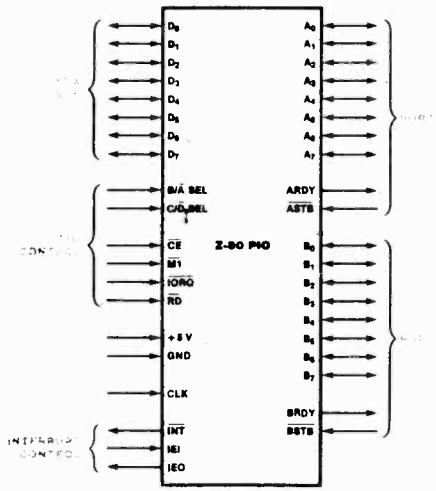


Figure 1. Pin Functions

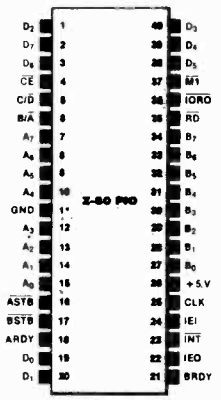


Figure 2. Pin Assignments

The Z-80 PIO interfaces to peripherals via two independent general-purpose I/O ports, designated Port A and Port B. Each port has eight data bits and two handshake signals, Ready and Strobe, which control data transfer. The Ready output indicates to the peripheral that the port is ready for a data transfer. Strobe is an input from the peripheral that indicates when a data transfer has occurred.

Operating Modes. The Z-80 PIO ports can be programmed to operate in four modes: byte output (Mode 0), byte input (Mode 1), byte input/output (Mode 2) and bit input/output (Mode 3).

In Mode 0, either Port A or Port B can be programmed to output data. Both ports have output registers that are individually addressed by the CPU; data can be written to either port at any time. When data is written to a port, an active Ready output indicates to the external device that data is available at the associated port and is ready for transfer to the external device. After the data transfer, the external device responds with an active Strobe input, which generates an interrupt, if enabled.

In Mode 1, either Port A or Port B can be configured in the input mode. Each port has an input register addressed by the CPU. When the CPU reads data from a port, the PIO sets the Ready signal, which is detected by the external device. The external device then places data on the I/O lines and strobes the I/O port, which latches the data into the Port Input Register, resets Ready, and triggers the

Interrupt Request, if enabled. The CPU can read the input data at any time, which again sets Ready.

Mode 2 is bidirectional and uses Port A, plus the interrupts and handshake signals from both ports. Port B must be set to Mode 3 and masked off. In operation, Port A is used for both data input and output. Output operation is similar to Mode 0 except that data is allowed out onto the Port A bus only when $\overline{ASTB}$ is Low. For input, operation is similar to Mode 1, except that the data input uses the Port B handshake signals and the Port B interrupt (if enabled).

Both ports can be used in Mode 3. In this mode, the individual bits are defined as either input or output bits. This provides up to eight separate, individually defined bits for each port. During operation, Ready and Strobe are not used. Instead, an interrupt is generated if the condition of one input changes, or if all inputs change. The requirements for generating an interrupt are defined during the programming operation; the active level is specified as either High or Low, and the logic condition is specified as either one input active (OR) or all inputs active (AND). For example, if the port is programmed for active Low inputs and the logic function is AND, then all inputs at the specified port must go Low to generate an interrupt.

Data outputs are controlled by the CPU and can be written or changed at any time.

- Individual bits can be masked off.
- The handshake signals are not used in Mode 3; Ready is held Low, and Strobe is disabled.
- When using the Z-80 PIO interrupts, the Z-80 CPU interrupt mode must be set to Mode 2.

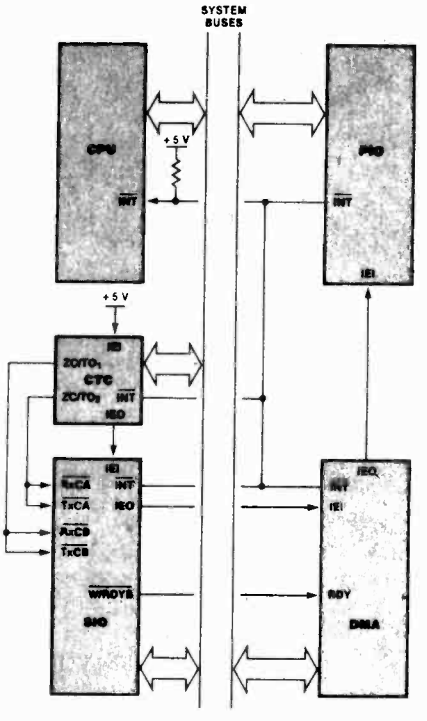


Figure 3. PIO in a Typical Z80 Family Environment

Pin Description

A_0-A_7. *Port A Bus* (bidirectional, 3-state).

ARDY. *Register A Ready* (output, active High). The meaning of this signal depends on the mode of operation selected for Port A as follows:

Output Mode. This signal goes active to indicate that the Port A output register has been loaded and the peripheral data bus is stable and ready for transfer to the peripheral device.

Input Mode. This signal is active when the Port A input register is empty and ready to accept data from the peripheral device.

Bidirectional Mode. This signal is active when data is available in the Port A output register for transfer to the peripheral device. In this mode, data is not placed on the Port A data bus, unless $\overline{ASTB}$ is active.

Control Mode. This signal is disabled and forced to a Low state.

$\overline{ASTB}$. *Port A Strobe Pulse From Peripheral Device* (input, active Low). The meaning of this signal depends on the mode of operation selected for Port A as follows:

Output Mode. The positive edge of this strobe is issued by the peripheral to acknowledge the receipt of data made available by the PIO.

Input Mode. The strobe is issued by the peripheral to load data from the peripheral into the Port A input register. Data is loaded into the PIO when this signal is active.

Bidirectional Mode. When this signal is active, data from the Port A output register is gated onto the Port A bidirectional data bus. The positive edge of the strobe acknowledges the receipt of the data.

Control Mode. The strobe is inhibited internally.

B_0-B_7. *Port B Bus* (bidirectional, 3-state). This 8-bit bus transfers data, status, or control information between Port B and a peripheral device. The Port B data bus can supply 1.5 mA at 1.5 V to drive Darlington transistors. B_0 is the least significant bit of the bus.

B/$\overline{A}$. *Port B Or A Select* (input, High = B). This pin defines which port is accessed during a data transfer between the CPU and the PIO. A Low on this pin selects Port A; a High selects Port B. Often address bit A_0 from the CPU is used for this selection function.

BRDY. *Register B Ready* (output, active High). This signal is similar to ARDY, except that in the Port A bidirectional mode this signal is High when the Port A input register is empty and ready to accept data from the peripheral device.

$\overline{BSTB}$. *Port B Strobe Pulse From Peripheral Device* (input, active Low). This signal is similar to $\overline{ASTB}$, except that in the Port A bidirectional mode this signal strobes data from the peripheral device into the Port A input register.

C/$\overline{D}$. *Control Or Data Select* (input, High = C). This pin defines the type of data transfer to be performed between the CPU and the PIO. A High on this pin during a CPU write to the PIO causes the Z-80 data bus to be

Pin Description (Continued)

interpreted as a *command* for the port selected by the B/$\overline{A}$ Select line. A Low on this pin means that the Z-80 data bus is being used to transfer data between the CPU and the PIO. Often address bit A_1 from the CPU is used for this function.

$\overline{CE}$. *Chip Enable* (input, active Low). A Low on this pin enables the PIO to accept command or data inputs from the CPU during a write cycle or to transmit data to the CPU during a read cycle. This signal is generally decoded from four I/O port numbers for Ports A and B, data, and control.

CLK. *System Clock* (input). The Z-80 PIO uses the standard single-phase Z-80 system clock.

D_0-D_7. *Z-80 CPU Data Bus* (bidirectional, 3-state). This bus is used to transfer all data and commands between the Z-80 CPU and the Z-80 PIO. D_0 is the least significant bit.

IEI. *Interrupt Enable In* (input, active High). This signal is used to form a priority-interrupt daisy chain when more than one interrupt-driven device is being used. A High level on this pin indicates that no other devices of higher priority are being serviced by a CPU interrupt service routine.

IEO. *Interrupt Enable Out* (output, active High). The IEO signal is the other signal required to form a daisy chain priority scheme. It is High only if IEI is High and the CPU is not servicing an interrupt from this PIO. Thus this signal blocks lower priority devices from interrupting while a higher priority device is being serviced by its CPU interrupt service routine.

$\overline{INT}$. *Interrupt Request* (output, open drain, active Low). When $\overline{INT}$ is active the Z-80 PIO is requesting an interrupt from the Z-80 CPU.

$\overline{IORQ}$. *Input/Output Request* (input from Z-80 CPU, active Low). $\overline{IORQ}$ is used in conjunction with B/$\overline{A}$, C/$\overline{D}$, $\overline{CE}$, and $\overline{RD}$ to transfer commands and data between the Z-80 CPU and the Z-80 PIO. When $\overline{CE}$, $\overline{RD}$, and $\overline{IORQ}$ are active, the port addressed by B/$\overline{A}$ transfers data to the CPU (a read operation). Conversely, when $\overline{CE}$ and $\overline{IORQ}$ are active but $\overline{RD}$ is not, the port addressed by B/$\overline{A}$ is written into from the CPU with either data or control information, as specified by C/$\overline{D}$.

M1. *Machine Cycle* (input from CPU, active Low). This signal is used as a sync pulse to control several internal PIO operations. When both the $\overline{M1}$ and $\overline{RD}$ signals are active, the Z-80 CPU is fetching an instruction from memory. Conversely, when both $\overline{M1}$ and $\overline{IORQ}$ are active, the CPU is acknowledging an interrupt. In addition, $\overline{M1}$ has two other functions within the Z-80 PIO: it synchronizes the PIO interrupt logic; when $\overline{M1}$ occurs without an active $\overline{RD}$ or $\overline{IORQ}$ signal, the PIO is reset.

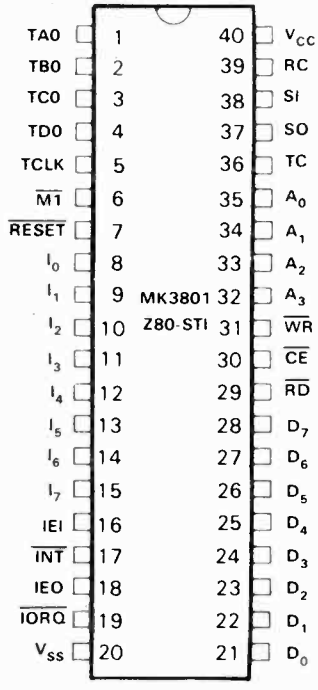
RD. *Read Cycle Status* (input from Z-80 CPU, active Low). If $\overline{RD}$ is active, or an I/O operation is in progress, $\overline{RD}$ is used with B/$\overline{A}$, C/$\overline{D}$, $\overline{CE}$, and $\overline{IORQ}$ to transfer data from the Z-80 PIO to the Z-80 CPU.

MOSTEK MK3801 STI

Data courtesy of Mostek UK Ltd.

FEATURES

- ☐ Full duplex USART with programmable DMA control signals
- ☐ Two binary delay timers
- ☐ Two full feature timers with
  - Delay to interrupt mode
  - Pulse width measurement mode
  - Event counter mode
- ☐ Eight general purpose lines with
  - Full bi-directional I/O capability
  - Edge triggered interrupts on either edge
- ☐ Full control of each interrupt channel
  - Enable/disable
  - Maskable
  - Automatic end-of-interrupt mode
  - Software end-of-interrupt mode
- ☐ 2.5, 4 MHz, and 6 MHz versions available



DEVICE PINOUT
Figure 1

INTRODUCTION

The MK3801 Z80 STI (Serial Timer Interrupt) is a multifunctional peripheral device for use in Z80 microprocessor based systems. It is designed to optimize current systems by reducing chip count and system costs. By providing a USART, four timers (two binary and two full function), and eight bi-directional I/O lines with individually programmable interrupts, the MK3801 can make substantial improvement to any Z80 based system.

Control and operation of the MK3801 are provided by 24 internal registers accessible by the Z80 bus. Sixteen of these registers are directly addressable and accessible; eight are indirectly addressable. Two of the four timers provide full service features, while the other two provide delay timer features only. Serial Communication is provided by the USART, which is capable of either asynchronous or synchronous operation, optional sync word recognition and stripping, and programmable DMA control handshake lines. Eight bi-directional I/O lines provide parallel I/O capability and individually programmable interrupt capability. The interrupt structure of the device is fully programmable for all interrupts, provides for interrupt vector generation, conforms to the Z80 daisy chain interrupt priority scheme, and supports automatic end of interrupt functions for the Z80.

INDIRECTLY ADDRESSABLE REGISTERS

Indirectly Addressable Registers are addressed by placing the indirect address in bits IA0-IA2 of the Pointer/Vector Register.

INTERRUPT CONTROL REGISTERS

The Interrupt Control Registers provide control of interrupt processing for all I/O facilities of the MK3801. These registers allow the programmer to enable or disable any or all of the 16 interrupts, provide masking for any interrupts, and access to the pending or in-service status of the interrupts. Optional End-of-Interrupt modes are available under software control.

INTERRUPT OPERATION

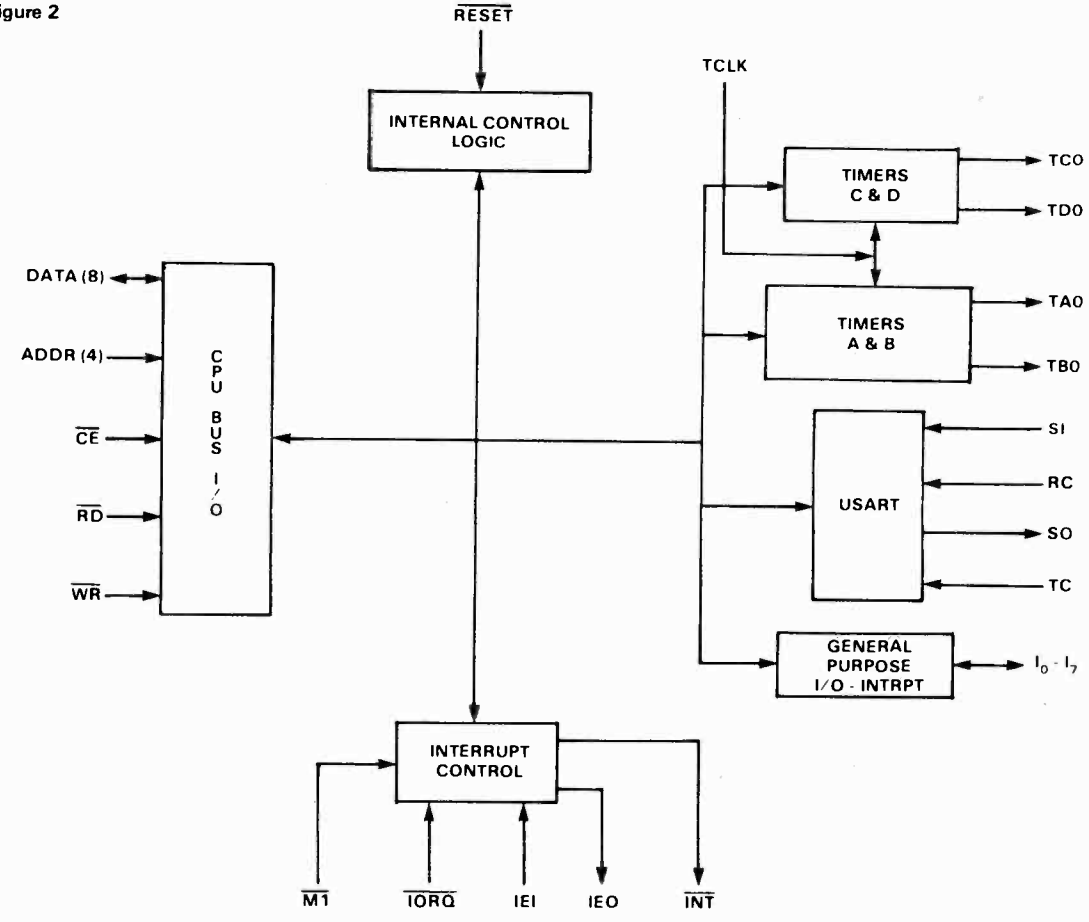
The Interrupt Enable Registers enable or disable the setting of an interrupt in the Interrupt Pending Registers. A 0 in a bit of the Interrupt Enable Registers disables the interrupt for the associated channel while a 1 enables the interrupt.

INTERRUPT VECTOR DEFINITION

Each individual function in the MK3801 is provided with a unique interrupt vector that is presented to the system during the interrupt acknowledge cycle.

INTERNAL ORGANIZATION

Figure 2



MOSTEK MK3882 CTC

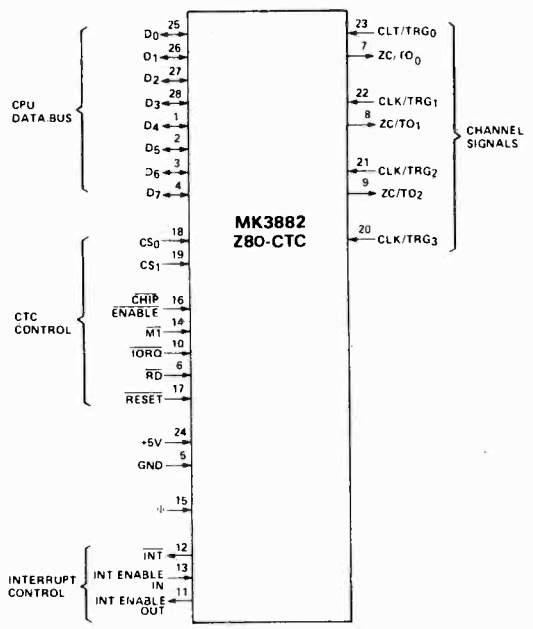
Data courtesy of Mostek UK Ltd.

FEATURES

- All inputs and outputs fully TTL compatible
- Each channel may be selected to operate in either Counter Mode or Timer Mode
- Used in either mode, a CPU-readable Down Counter indicates number of counts-to-go until zero
- A Time Constant Register can automatically reload the Down Counter at Count Zero in Counter and Timer Mode
- Selectable positive or negative trigger initiates time operation in Timer Mode. The same input is monitored for event counts in Counter Mode.
- Three channels have Zero Count/Timeout outputs capable of driving Darlington transistors
- Interrupts may be programmed to occur on the zero count condition in any channel
- Daisy chain priority interrupt logic included to provide for automatic interrupt vectoring without external logic

Z80-CTC PIN CONFIGURATION

Figure 1



INTRODUCTION

The Z80-Counter Timer Circuit (CTC) is a programmable component with four independent channels that provide counting and timing functions for microcomputer systems based on the Z80-CPU. The CPU can configure the CTC channels to operate under various modes and conditions as required to interface with a wide range of devices. In most applications, little or no external logic is required. The Z80-CTC utilizes N-channel silicon gate depletion load technology and is packaged in a 28-pin DIP. The Z80-CTC requires only a single 5 volt supply and a one-phase 5 volt clock.

CTC PIN DESCRIPTION

A diagram of the Z80-CTC pin configuration is shown in Figure 1. This section describes the function of each pin.

D_7-D_0 — Z80-CPU Data Bus (bidirectional, tristate)
This bus is used to transfer all data and command words between the Z80-CPU and the Z80-CTC. There are 8 bits on this bus, of which D_0 is the least significant.

CS1-CS0 — Channel Select (input, active high)
These pins form a 2-bit binary address code for selecting one of the four independent CTC channels for an I/O Write or Read (See truth table below.)

	CS1	CS0
Ch0	0	0
Ch1	0	1
Ch2	1	0
Ch3	1	1

$\overline{CE}$ — Chip Enable (input, active low)
A low level on this pin enables the CTC to accept control words, Interrupt Vectors, or time constant data words from the Z80 Data Bus during an I/O Write cycle, or to transmit the contents of the Down Counter to the CPU during an I/O Read cycle. In most applications this signal is decoded from the 8 least significant bits of the address bus for any of the four I/O port addresses that are mapped to the four Counter/Timer Channels.

Clock (Φ) — System Clock (input)
This single-phase clock is used by the CTC to synchronize certain signals internally.

$\overline{M1}$ — Machine Cycle One Signal from CPU (input, active low)
When $\overline{M1}$ is active and the $\overline{RD}$ signal is active, the CPU is fetching an instruction from

memory. When $\overline{M1}$ is active and $\overline{IORQ}$ is active, the CPU is acknowledging an interrupt, alerting the CTC to place an Interrupt Vector on the Z80 Data Bus if it has daisy chain priority and one of its channels has requested an interrupt.

$\overline{IORQ}$ — Input/Output Request from CPU (input, active low)
The $\overline{IORQ}$ signal is used in conjunction with the $\overline{CE}$ and $\overline{RD}$ signals to transfer data and Channel Control Words between the Z80-CPU and the CTC.

$\overline{RD}$ — Read Cycle Status from the CPU (input, active low)
The $\overline{RD}$ signal is used in conjunction with the $\overline{IORQ}$ and $\overline{CE}$ signals to transfer data and Channel Control Words between the Z80-CPU and the CTC. During a CTC Write Cycle, $\overline{IORQ}$ and $\overline{CE}$ must be true and $\overline{RD}$ false. The CTC does not receive a specific write signal, instead generating its own internally from the inverse of a valid $\overline{RD}$ signal. In a CTC Read Cycle, $\overline{IORQ}$, $\overline{CE}$ and $\overline{RD}$ must be active to place the contents of the Down Counter on the Z80 Data Bus.

IEI — Interrupt Enable In (input, active high)
This signal is used to help form a system-wide interrupt daisy chain which establishes priorities when more than one peripheral device in the system has interrupting capability. A high level on this pin indicates that no other interrupting devices of higher priority are being serviced by the Z80-CPU.

IEO — Interrupt Enable Out (output, active high)
The IEO signal, in conjunction with IEI, is used to form a system-wide interrupt priority daisy chain. IEO is high only if IEI is high and the CPU is not servicing an interrupt from any CTC channel. Thus this signal blocks lower priority devices from interrupting while a higher priority device is being serviced by the CPU.

$\overline{INT}$ — Interrupt Request (output, open drain, active low)

$\overline{RESET}$ — Reset (input, active low)
This signal stops all channels from counting and resets channel interrupt enable bits in all control registers, thereby disabling CTC-generated interrupts. The ZC/TO and $\overline{INT}$ outputs go to their inactive states, IEO reflects IEI, and the CTC's data bus output drivers go to the high impedance state.

CLK/TRG3-CLK/TRG0 — External Clock/Timer Trigger (input, user-selectable active high or low)

ZC/TO2—ZC/TO0 — Zero Count/Timeout (output, active high)
There are three ZC/TO pins, corresponding to CTC channels 2 through 0.

MOSTEK MK3883 DMA

Data courtesy of Mostek UK Ltd.

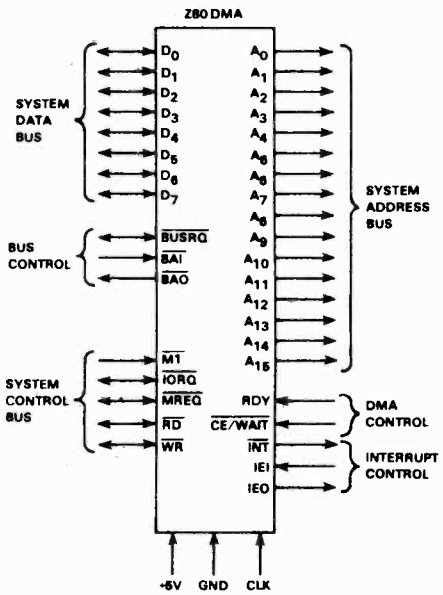
FEATURES

- Transfers, searches and search/transfers in byte-at-a-time, burst or continuous modes. Cycle length and edge timing can be programmed to match the speed of any port.
- Dual port addresses (source and destination) generated for memory-to-I/O, memory-to-memory, or I/O-to-I/O operations. Addresses may be fixed or automatically incremented/decremented.
- Next-operation loading without disturbing current operations via buffered starting-address registers. An entire previous sequence can be repeated automatically.
- Extensive programmability of functions. CPU can read complete channel status.
- Standard Z80 Family bus-request and prioritized interrupt-request daisy chains implemented without external logic. Sophisticated, internally modifiable interrupt vectoring.
- Direct interfacing to system buses without external logic.

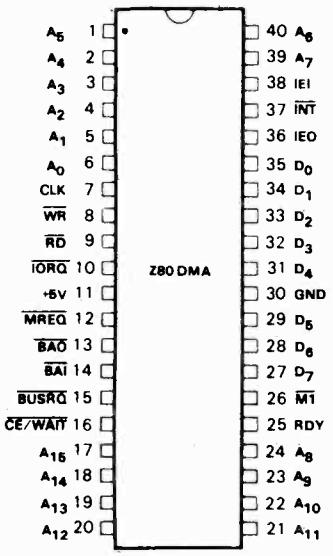
GENERAL DESCRIPTION

The MK3883 Z80 DMA (Direct Memory Access) is a powerful and versatile device for controlling and processing transfers of data. Its basic function of managing CPU-independent transfers between two ports is augmented by an array of features that optimize transfer speed and control with little or no external logic in systems using an 8- or 16-bit data bus and a 16-bit address bus.

PIN FUNCTIONS
Figure 1



PIN ASSIGNMENTS
Figure 2



Transfers can be done between any two ports (source and destination), including memory-to-I/O, memory-to-memory, and I/O-to-I/O. Dual port addresses are automatically generated for each transaction and may be either fixed or incrementing/decrementing. In addition, bit-maskable byte searches can be performed either concurrently with transfers or as an operation in itself.

The MK3883 Z80 DMA contains direct interfacing to and independent control of system buses, as well as sophisticated bus and interrupt controls. Many programmable features, including variable cycle timing and auto-restart minimize CPU software overhead. They are especially useful in adapting this special-purpose transfer processor to a broad variety of memory, I/O and CPU environments.

The MK3883 Z80 DMA is an n-channel silicon-gate depletion-load device packaged in a 40-pin plastic, or ceramic DIP. It uses a single +5V power supply and the standard Z80 Family single-phase clock.

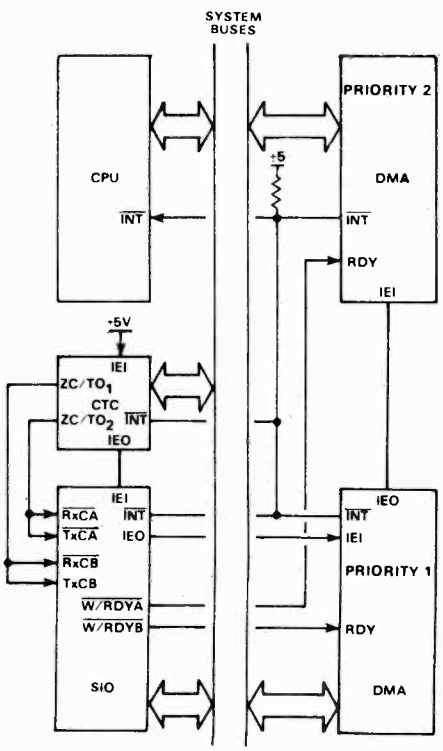
FUNCTIONAL DESCRIPTION

Classes of Operation

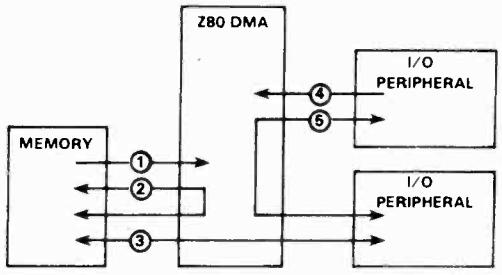
The MK3883 Z80 DMA has three basic classes of operation:

- Transfers of data between two ports (memory or I/O peripheral)
- Searches for a particular 8-bit maskable byte at a single port in memory or an I/O peripheral
- Combined transfers with simultaneous search between two ports

Z80 ENVIRONMENT WITH MULTIPLE DMA CONTROLLERS
Figure 3



BASIC FUNCTIONS OF THE Z80 DMA
Figure 4



1. Search memory
2. Transfer memory-to-memory (optional search)
3. Transfer memory-to-I/O (optional search)
4. Search I/O
5. Transfer I/O-to-I/O (optional search)

During a transfer, the DMA assumes control of the system control, address, and data buses. Data is read from one addressable port and written to the other addressable port, byte by byte. The ports may be programmed to be either system main memory or peripheral I/O devices. Thus, a block of data may be written from one peripheral to another, from one area of main memory to another, or from a peripheral to main memory and vice versa.

During a search-only operation, data is read from the source port and compared byte by byte with the DMA-internal register containing a programmable match byte. This match byte may optionally be masked so that only certain bits within the match byte are compared. Search rates up to 1.25M bytes per second can be obtained with the 2.5MHz MK3883 Z80 DMA or 2M bytes per second with the 4MHz MK3883-4 Z80 DMA. In combined searches and transfers, data is transferred between two ports while simultaneously searching for a bit-maskable byte match.

Data transfers or searches can be programmed to stop or interrupt under various conditions. In addition, CPU-readable status bits can be programmed to reflect the condition.

Modes of Operation

The MK3883 Z80 DMA can be programmed to operate in one of three transfer and/or search modes:

- Byte-at-a-time: data operations are performed one byte at a time. Between each byte operation the system buses are released to the CPU. The buses are requested again for each succeeding byte operation.
- Burst: data operations continue until a port's Ready line to the DMA goes inactive. The DMA then stops and releases the system buses after completing its current byte operation.
- Continuous: data operations continue until the end of the programmed block of data is reached before the system buses are released. If a port's Ready line goes inactive before this occurs, the DMA simply pauses until the Ready line comes active again.

MOSTEK MK3884 SIO

Data courtesy of Mostek UK Ltd.

FEATURES

- Two independent full-duplex channels, with separate control and status lines for modems or other devices.
- Data rates of 0 to 500K bits/second in the x1 clock mode with a 2.5MHz clock (MK3884 Z80 SIO), or 0 to 800K bits/second with a 4.0MHz clock (MK3884-4 Z80 SIO).
- Asynchronous protocols: everything necessary for complete messages in 5, 6, 7 or 8 bits/character. Includes variable stop bits and several clock-rate multipliers; break generation and detection; parity; overrun and framing error detection.
- Synchronous protocols: everything necessary for complete bit- or byte-oriented messages in 5, 6, 7 or 8 bits/character, including IBM Bisync, SDLC, HDLC, CCITT-X.25 and others. Automatic CRC generation/-checking, sync character and zero insertion/-deletion, abort generation/detection and flag insertion.
- Receiver data registers quadruply buffered, transmitter registers doubly buffered.
- Highly sophisticated and flexible daisy-chain interrupt vectoring for interrupts without external logic.

DESCRIPTION

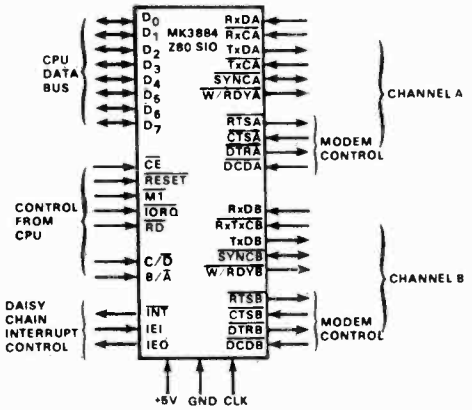
The MK3884 Z80 SIO Serial Input/Output Controller is a dual-channel data communication interface with extraordinary versatility and capability. Its basic functions as a serial-to-parallel, parallel-to-serial converter/controller can be programmed by a CPU for a broad range of serial communication applications.

The device supports all common asynchronous and synchronous protocols, byte- or bit-oriented, and performs all of the functions traditionally done by UARTs, USARTs and synchronous communication controllers combined, plus additional functions traditionally performed by the CPU. Moreover, it does this on two fully-independent channels, with an exceptionally sophisticated interrupt structure that allows very fast transfers.

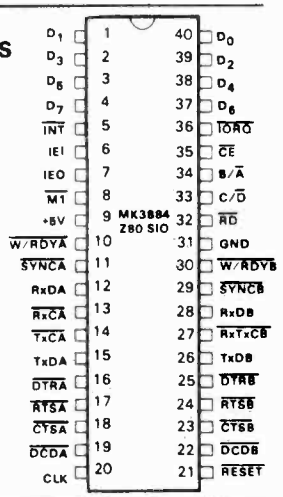
Full interfacing is provided for CPU or DMA control. In addition to data communication, the circuit can handle virtually all types of serial I/O with fast (or slow) peripheral devices. While designed primarily as a member of the Z80 family, its versatility makes it well suited to many other CPUs.

The Z80 SIO is an n-channel silicon-gate depletion-load device packaged in a 40-pin plastic, or ceramic DIP. It uses a single +5V power supply and the standard Z80 family single-phase clock.

MK3884 Z80 SIO PIN FUNCTIONS
Figure 1



MK3884 Z80 SIO PIN ASSIGNMENTS
Figure 2



PIN DESCRIPTIONS

Figures 1 through 6 illustrate the three pin configurations (bonding options) available in the SIO. The constraints of a 40-pin package make it impossible to bring out the Receive Clock ($\overline{RxC}$), Transmit Clock ($\overline{TxC}$), Data Terminal Ready ($\overline{DTR}$) and Sync ($\overline{SYNC}$) signals for both channels. Therefore, either Channel B lacks a signal or two signals are bonded together in the three bonding options offered:

- MK3887 Z80 SIO lacks $\overline{SYNCB}$
- MK3885 Z80 SIO lacks $\overline{DTRB}$
- MK3884 Z80 SIO has all four signals, but $\overline{TxCB}$ and $\overline{RxCB}$ are bonded together

The pin descriptions are as follows:

B/$\overline{A}$. Channel A Or B Select (input, High selects Channel B). This input defines which channel is accessed during a data transfer between the CPU and the SIO. Address bit A_0 from the CPU is often used for the selection function.

C/$\overline{D}$. Control Or Data Select (input, High selects Control). This input defines the type of information transfer performed between the CPU and the SIO. A High at this input during a CPU write to the SIO causes the information on the data bus to be interpreted as a command for the channel selected by B/$\overline{A}$. A Low at C/$\overline{D}$ means that the information on the data bus is data. Address bit A_1 is often used for this function.

$\overline{CE}$. Chip Enable (Input, active Low). A Low level at this input enables the SIO to accept command or data input from the CPU during a write cycle, or to transmit data to the CPU during a read cycle.

CLK. System Clock (input). The SIO uses the standard Z80 System Clock to synchronize internal signals. This is a single-phase clock.

$\overline{CTSA}$, $\overline{CTSB}$. Clear To Send (inputs, active Low). When programmed as Auto Enables, a Low on these inputs enables the respective transmitter. If not programmed as Auto Enables, these inputs may be programmed as general-purpose inputs. Both inputs are Schmitt-trigger buffered to accommodate slow-risetime signals. The SIO detects pulses on these inputs and interrupts the CPU on both logic level transitions. The Schmitt-trigger buffering does not guarantee a specified noise-level margin.

D_0-D_7. System Data Bus (bidirectional, 3-state). The system data bus transfers data and commands between the CPU and the Z80 SIO. D_0 is the least significant bit.

$\overline{DCDA}$, $\overline{DCDB}$. Data Carrier Detect (inputs, active Low). These pins function as receiver enables if the SIO is programmed for Auto Enables; otherwise they may be used as general-purpose input pins. Both pins are Schmitt-trigger buffered to accommodate slow-risetime signals. The SIO detects pulses on these pins and interrupts the CPU on both logic level transitions. Schmitt-trigger buffering does not guarantee a specific noise-level margin.

$\overline{DTRA}$, $\overline{DTRB}$. Data Terminal Ready (outputs, active Low). These outputs follow the state programmed into Z80 SIO. They can also be programmed as general-purpose outputs.
In the MK3885 bonding option, $\overline{DTRB}$ is omitted.

IEI. Interrupt Enable In (input, active High). This signal is used with IEO to form a priority daisy chain when there is more than one interrupt-driven device. A High on this line indicates that no other device of higher priority is being serviced by a CPU interrupt service routine.

IEO. Interrupt Enable Out (output, active High). IEO is High only if IEI is High and the CPU is not servicing an interrupt from this SIO. Thus, this signal blocks lower priority devices from interrupting while a higher priority device is being serviced by its CPU interrupt service routine.

$\overline{INT}$. Interrupt Request (output, open drain, active Low). When the SIO is requesting an interrupt, it pulls $\overline{INT}$ Low.

$\overline{IORQ}$. Input/Output Request (input from CPU, active Low). $\overline{IORQ}$ is used in conjunction with B/$\overline{A}$, C/$\overline{D}$, $\overline{CE}$

and $\overline{RD}$ to transfer commands and data between the CPU and the SIO. When $\overline{CE}$, $\overline{RD}$ and $\overline{IORQ}$ are all active, the channel selected by B/$\overline{A}$ transfers data to the CPU (a read operation). When $\overline{CE}$ and $\overline{IORQ}$ are active, but $\overline{RD}$ is inactive, the channel selected by B/$\overline{A}$ is written to by the CPU with either data or control information as specified by C/$\overline{D}$. As mentioned previously, if $\overline{IORQ}$ and $\overline{M1}$ are active simultaneously, the CPU is acknowledging an interrupt and the SIO automatically places its interrupt vector on the CPU data bus if it is the highest priority device requesting an interrupt.

$\overline{M1}$. Machine Cycle (input from Z80 CPU, active Low). When $\overline{M1}$ is active and $\overline{RD}$ is also active, the Z80 CPU is fetching an instruction from memory; when $\overline{M1}$ is active while $\overline{IORQ}$ is active, the SIO accepts $\overline{M1}$ and $\overline{IORQ}$ as an interrupt acknowledge if the SIO is the highest priority device that has interrupted the Z80 CPU.

$\overline{RxCA}$, $\overline{RxCB}$. Receiver Clocks (inputs). Receive data is sampled on the rising edge of $\overline{RxC}$. The Receive Clocks may be 1, 16, 32 or 64 times the data rate in asynchronous modes. These clocks may be driven by the Z80 CTC Counter Timer Circuit for programmable baud rate generation. Both inputs are Schmitt-trigger buffered (no noise level margin is specified).

In the MK3884 bonding option, $\overline{RxCB}$ is bonded together with $\overline{TxCB}$.

$\overline{RD}$. Read Cycle Status (input from CPU, active Low). If $\overline{RD}$ is active, a memory or I/O read operation is in progress. $\overline{RD}$ is used with B/$\overline{A}$, $\overline{CE}$ and $\overline{IORQ}$ to transfer data from the SIO to the CPU.

RxDA, RxDB. Receive Data (inputs, active High). Serial data at TTL levels.

$\overline{RESET}$. Reset (input, active Low). A Low $\overline{RESET}$ disables both receivers and transmitters, forces TxDA and TxDB marking, forces the modem controls High and disables all interrupts. The control registers must be rewritten after the SIO is reset and before data is transmitted or received.

$\overline{RTSA}$, $\overline{RTSB}$. Request To Send (outputs, active Low). When the RTS bit in Write Register 5 (Figure 14) is set, the $\overline{RTS}$ output goes Low. When the $\overline{RTS}$ bit is reset in the Asynchronous mode, the output goes High after the transmitter is empty. In Synchronous modes, the $\overline{RTS}$ pin strictly follows the state of the RTS bit. Both pins can be used as general-purpose outputs.

$\overline{SYNCA}$, $\overline{SYNCB}$. Synchronization (inputs/outputs, active Low). These pins can act either as inputs or outputs. In the asynchronous receive mode, they are inputs similar to $\overline{CTS}$ and $\overline{DCD}$. In this mode, the transitions on these lines affect the state of the Sync/-Hunt status bit in Read Register 0 (Figure 13), but have no other function. In the External Sync mode, these lines also act as inputs. When external synchronization is achieved, $\overline{SYNC}$ must be driven Low on the second rising edge of $\overline{RxC}$ after that rising edge of $\overline{RxC}$ on which the last bit of the sync character was received.

In the internal synchronization mode (Monosync and Bisync), these pins act as outputs that are active during the part of the receive clock ($\overline{RxC}$) cycle in which sync characters are recognized. The sync condition is not latched, so these outputs are active each time a sync pattern is recognized, regardless of character boundaries.

In the MK3887 bonding option, $\overline{SYNCB}$ is omitted.

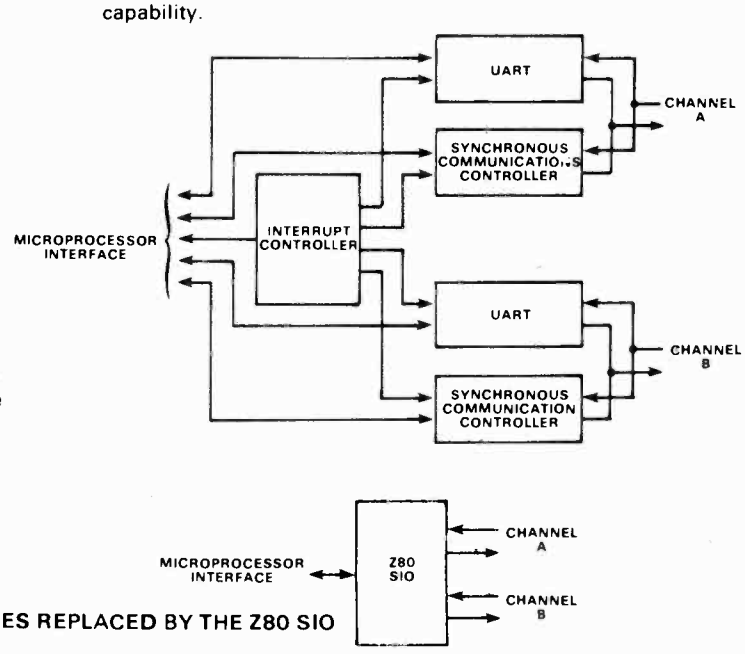
$\overline{TxCA}$, $\overline{TxCB}$. Transmitter Clocks (inputs). TxD changes from the falling edge of $\overline{TxC}$. In asynchronous modes, the Transmitter Clocks may be 1, 16, 32 or 64 times the data rate; however, the clock multiplier for the transmitter and the receiver must be the same. The Transmit Clock inputs are Schmitt-trigger buffered for relaxed rise- and fall-time requirements (no noise level margin is specified). Transmitter Clocks may be driven by the Z80 CTC Counter Timer Circuit for programmable baud rate generation.
In the MK3884 bonding option, $\overline{TxCB}$ is bonded together with $\overline{RxCB}$.

TxDA, TxDB. Transmit Data (outputs, active High). Serial data at TTL levels.

$\overline{W/RDYA}$, $\overline{W/RDYB}$. Wait/Ready A, Wait/Ready B (outputs, open drain, when programmed for Wait function; driven High and Low when programmed for Ready function). These dual-purpose outputs may be programmed as Ready lines for a DMA controller or as Wait lines that synchronize the CPU to the SIO data rate. The reset state is open drain.

FUNCTIONAL CAPABILITIES

The functional capabilities of the Z80 SIO can be described from two different points of view: as a data communications device, it transmits and receives serial data in a wide variety of data-communication protocols; as a Z80 family peripheral, it interacts with the Z80 CPU and other peripheral circuits, sharing the data, address and control buses, as well as being a part of the Z80 interrupt structure. As a peripheral to other microprocessors, the SIO offers valuable features such as non-vectored interrupts, polling and simple handshake capability.



CONVENTIONAL DEVICES REPLACED BY THE Z80 SIO
Figure 8

SGS-ATES Z80 DART

Data courtesy of SGS-ATES (UK) Ltd.

Features

- Two independent full-duplex channels with separate modem controls. Modem status can be monitored.
- Receiver data registers are quadruply buffered; the transmitter is doubly buffered.
- Interrupt features include a programmable interrupt vector, a "status affects vector" mode for fast interrupt processing, and the standard Z80 peripheral daisy-chain interrupt structure that provides automatic interrrupt vectoring with no external logic.
- In x1 clock mode, data rates are 0 to 500K bits/second with a 2.5 MHz clock, or 0 to 800K bits/second with a 4.0 MHz clock.
- Programmable options include 1, 1½ or 2 stop bits; even, odd or no parity; and x1, x16, x32 and x64 clock modes.
- Break generation and detection as well as parity-, overrun-and framing-error detection are available.

Description

The Z80 DART (Dual-Channel Asynchronous Receiver/Transmitter) is a dual-channel multi-function peripheral component that satisfies a wide variety of asynchronous serial data communications requirements in micro-computer systems. The Z80 DART is used as a serial-to-parallel, parallel-to-serial converter/controller in asynchronous applications. In addition, the device also provides modem controls for both channel. In application where modem controls are not needed, these lines can be used for general-purpose I/O.

SGS-ATES also offers the Z80 SIO, a more versatile device that provides synchronous (Bisync, HDLC and SDLC) as well as asynchronous operation).

The Z80 DART is fabricated with n-channel silicon-gate depletion-load technology, and is packaged in a 40-pin plastic or ceramic DIP.

Fig. 1 - PIN FUNCTION

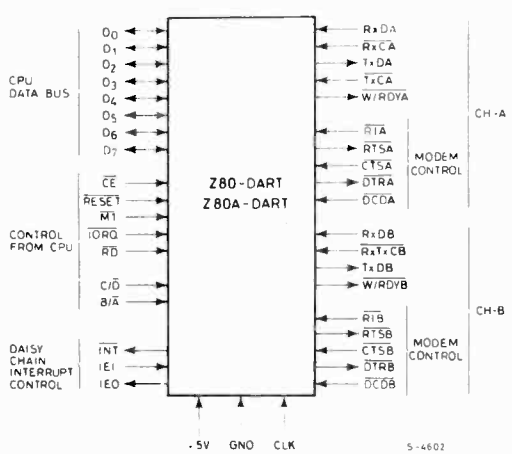
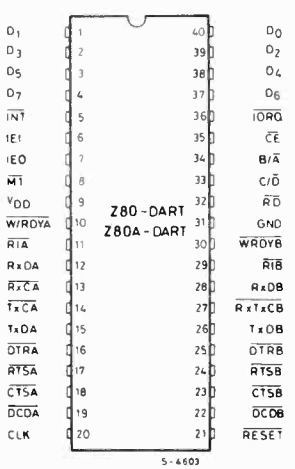


Fig. 2 - PIN ASSIGNMENTS



B/$\overline{A}$. *Channel A Or B Select* (input, High selects Channel B). This input defines which channel is accessed during a data transfer between the CPU and the Z80 DART.

C/$\overline{D}$. *Control Or Data Select* (input, High selects Control). This input specifies the type of information (control or data) transferred on the data bus between the CPU and the Z80 DART.

$\overline{CE}$. *Chip Enable* (input, active Low). A low at this input enables the Z80 DART to accept command or data input from the CPU during a write cycle, or to transmit data to the CPU during a read cycle.

CLK. *System Clock* (input). The Z80 DART uses the standard Z80 single-phase system clock to synchronize internal signals.

$\overline{CTSA}$, $\overline{CTSB}$. *Clear To Send* (inputs, active Low). When programmed as Auto Enables, a Low on these inputs enables the respective transmitter. If not programmed as Auto Enables, these inputs may be programmed as general-purpose inputs. Both inputs are Schmitt-trigger buffered to accomodate slow-risetime signals.

D_0-D_7. *System Data Bus* (bidirectional, 3-state) transfers data and commands between the CPU and the Z80 DART.

$\overline{DCDA}$, $\overline{DCDB}$. *Data Carrier Detect* (inputs, active Low). These pins function as receiver enables if the Z80 DART is programmed for Auto Enables; otherwise they may be used as general-purpose input pins. Both pins are Schmitt-trigger buffered.

$\overline{DTRA}$, $\overline{DTRB}$. *Data Terminal Ready* (outputs, active Low). These outputs follow the state programmed into the DTR bit. They can also be programmed as general-purpose outputs.

IEI. *Interrupt Enable In* (input, active High) is used with IEO to form a priority daisy chain when there is more than one interrupt-driven device. A High on this line indicates that no other device of higher priority is being serviced by a CPU interrupt service routine.

IEO. *Interrupt Enable Out* (output, active High). IEO is High only if IEI is High and the CPU is not servicing an interrupt from this Z80 DART. Thus, this signal blocks lower priority devices from interrupting while a higher priority device is being serviced by its CPU interrupt service routine.

$\overline{INT}$. *Interrupt Request* (output, open drain, active Low). When the Z80 DART is requesting an interrupt, it pulls $\overline{INT}$ Low.

$\overline{M1}$. *Machine Cycle One* (input from Z80 CPU, active Low). When $\overline{M1}$ and $\overline{RD}$ are both active, the Z80 CPU is fetching an instruction from memory; when $\overline{M1}$ is active while $\overline{IORQ}$ is active, the Z80 DART accepts $\overline{M1}$ and $\overline{IORQ}$ as an interrupt aknowledge if the Z80 DART is the highest priority device that has interrupted the Z80 CPU.

$\overline{IORQ}$. *Input/Output Request* (input from CPU, active Low). $\overline{IORQ}$ is used in conjunction with B/$\overline{A}$, C/$\overline{D}$, $\overline{CE}$ and $\overline{RD}$ to transfer commands and data between the CPU and the Z80 DART. When $\overline{CE}$, $\overline{RD}$ and $\overline{IORQ}$ are all active, the channel selected by B/$\overline{A}$ transfers data to the CPU (a read operation). When $\overline{CE}$ and $\overline{IORQ}$ are active, but $\overline{RD}$ is inactive, the channel selected by B/$\overline{A}$ is written to by the CPU with either data or control information as specified by C/$\overline{D}$.

$\overline{RxCA}$, $\overline{RxCB}$. *Receiver Clocks* (inputs). Receive data is sampled on the rising edge of $\overline{RxC}$. The Receive Clocks may be 1, 16, 32 or 64 times the data rate.

$\overline{RD}$. *Read Cycle Status* (input from CPU, active Low). If $\overline{RD}$ is active, a memory or I/O read operation is in progress.

RxDA, RxDB. *Receive Data* (inputs, active High).

$\overline{RESET}$. *Reset* (input, active Low). Disables both receivers and transmitters, forces TxDA and TxDB marking, forces the modem controls High and disables all interrupts.

$\overline{RIA}$, $\overline{RIB}$. *Ring Indicator* (inputs, Active Low). These inputs are similar to $\overline{CTS}$ and $\overline{DCD}$. The Z80 DART detectes both logic level transitions and interrupts the CPU When not used in switched-line applications, these inputs can be used as general-purpose inputs.

$\overline{RTSA}$, $\overline{RTSB}$. *Request to Send* (outputs, active Low). When the RTS bit is set, the $\overline{RTS}$ output goes Low. When the RTS bit is reset, the output goes High after the transmitter empties.

$\overline{TxCA}$, $\overline{TxCB}$. *Transmitter Clocks* (inputs). TxD changes on the falling of TxC. The Transmitter Clocks may be 1, 16, 32 or 64 times the data rate; however, the clock multiplier for the transmitter and the receiver must be the same. The Transmit Clock inputs are Schmitt-trigger buffered. Both the Receiver and Transmitter Clocks may be driven by the Z80 CTC Counter Time Circuit for programmable baud rate generation.

TxDA, TxDB. *Transmit Data* (outputs, active High).

$\overline{W/RDYA}$, $\overline{W/RDYB}$. *Wait/Ready* (outputs, open drain when programmed for Wait function, driven High and Low when programmed for Ready function). These dual-purpose outputs may be programmed as Ready lines for a DMA controller or as Wait lines that synchronize the CPU to the Z80 DART data rate. The reset state is open drain.

I/O Interface Capabilities. The Z80 DART offers the choice of Polling, Interrupt (vectored or non-vectored) and Block Transfer modes to transfer data, status and control information to and from the CPU. The Block Transfer mode can be implemented under CPU or DMA control.

POLLING. There are no interrupts in the Polled mode. Status registers RR0 and RR1 are updated at appropriate times for each function being performed. All the interrupt modes of the Z80 DART must be disabled to operate the device in a Polled environment.

While in its Polling sequence, the CPU examines the status contained in RR0 for each channel; the RR0 status bits serve as an acknowledge to the Poll inquiry. The two RR0 status bits D_0 and D_2 indicate that a data transfer is needed. The status also indicates Error or other special status conditions (see "Z80 DART Programming"). The Special Receive Condition status contained in RR1 does not have to be read in a Polling sequence because the status bits in RR1 are accompanied by a Receive Character Available status in RR0.

INTERRUPTS. The Z80 DART offers an elaborate interrupt scheme that provides fast interrupt response in real-time applications. As a member of the Z80 family, the Z80 DART can be daisy-chained along with other Z80 peripherals for peripheral interrupt-priority resolution. In addition, the internal interrupts of the Z80 DART are nested to prioritize the various interrupts generated by Channels A and B. Channel B registers WR2 and RR2 contain the interrupt vector that points to an interrupt service routine in the memory. To eliminate the necessity of writing a status analysis routine, the Z80 DART can modify the interrupt vector in RR2 so it points directly to one of eight interrupt service routines. This is done under program control by setting a program bit (WR1, D_2) in Channel B called "Status Affects Vector". When this bit is set, the interrupt vector in RR2 is modified according to the assigned priority of the various interrupting conditions.

Transmit interrupts, Receive interrupts and External/Status interrupts are the main sources of interrupts. Each interrupts source is enabled under program control with Channel A having a higher priority than Channel B, and with Receiver, Transmit and External/Status interrupts prioritized in that order within each channel. When the Transmit interrupt is enabled, the CPU is interrupted by the transmit buffer *becoming* empty. (This implies that the transmitter must have had a data character written into it so it can become empty.) When enabled, the receiver can interrupt the CPU in one of three ways:

- Interrupt on the first received character
- Interrupt on all received characters
- Interrupt on a Special Receive condition

Interrupt On First Character is typically used with the Block Transfer mode. Interrupt On All Receive Characters can optionally modify the interrupt vector in the event of a parity error. The Special Receive Condition interrupt can occur on a character basis. The Special Receive condition can cause an interrupt only if the Interrupt On First Receive Character or Interrupt On All Receive Characters mode is selected. In Interrupt On First Receive Character, an interrupt can occur from Special Receive conditions (except Parity Error) after the first receive character interrupt (example: Receive Overrun interrupt).

The main function of the External/Status interrupt is to monitor the signal transitions of the $\overline{CTS}$, $\overline{DCD}$ and $\overline{RI}$ pins; however, an External/Status interrupt is also caused by the detection of a Break sequence in the data stream. The interrupt caused by the Break sequence has a special feature that allows the Z80 DART to interrupt when the Break sequence is detected or terminated. This feature facilitates the proper termination of the current message, correct initialization of the next message, and the accurate timing of the Break condition.

CPU/DMA BLOCK TRANSFER. The Z80 DART provides a Block Transfer mode to accomodate CPU block transfer functions and DMA block transfers (Z80 DMA or other designs). The Block Transfer mode uses the $\overline{W/RDY}$ output in conjunction with the Wait/Ready bits of Write Register 1. The $\overline{W/RDY}$ output can be defined under software control as a Wait line in the CPU Block Transfer mode or as a Ready line in the DMA Block Transfer mode.

To a DMA controller, the Z80 DART Ready output indicates that the Z80 DART is ready to transfer data to or from memory. To the CPU, the Wait output indicates that the Z80 DART is not ready to transfer data thereby requesting the CPU to extend the I/O cycle.

The functional capabilities of the Z80 DART can be described from two different points of view; as a data communications device, it transmits and receives serial data, and meets the requirements of asynchronous data communications protocols; as a Z80 family peripheral, it interacts with the Z80 CPU and other Z80 peripheral circuits, and shares the data, address and control buses, as well as being a part of the Z80 interrupt structure. As a peripheral to other microprocessor, the Z80 DART offers valuable features such as non-vectored interrupts, polling and simple handshake capability.

The first part of the following functional description introduces Z80 DART data communications capabilities; the second part describes the interaction between the CPU and the Z80 DART.

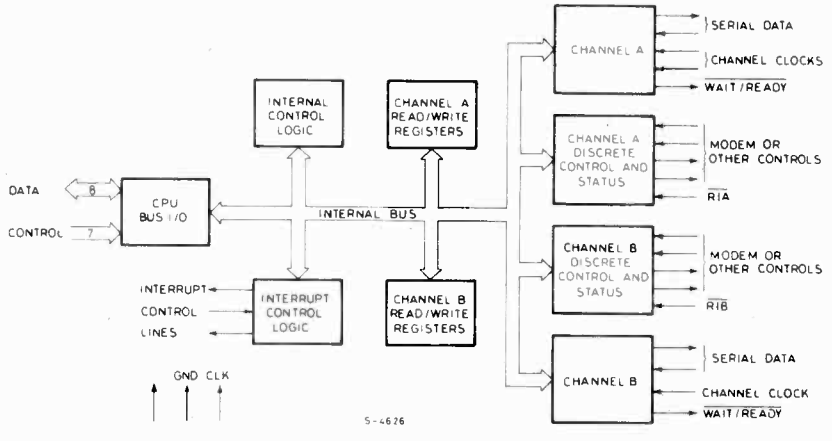
A more detailed explanation of Z80 DART operation can be found in the *Z80 SIO Technical Manual* (Document Number 03-3033-01). Because this manual was written for the Z80 SIO, it contains information about synchronous as well as asynchronous operation.

Communications Capabilities. The Z80 DART provides two independent full-duplex channels for use as an asynchronous receiver/transmitter. The following is a short description of receiver/transmitter capabilities. For more details, refer to the Asynchronous Mode section of the *Z80 SIO Technical Manual*. The Z80 DART offers transmission and reception of five to eight bits per character, plus optional even or odd parity. The transmitter can supply one, one and a half or two stop bits per character and can provide a break output at any time. The receiver break detection logic interrupts the CPU both at the start and end of a received break. Reception is protected from spikes by a transient spike rejection mechanism that checks the signal one-half a bit time after a Low level is detected on the Receive Data input. If the Low does not persist — as in the case of a transient — the character assembly process is not started.

Framing errors and overrun errors are detected and buffered together with the character on which they occurred. Vectored interrupts allow fast servicing of interrupting conditions using dedicated routines. Furthermore, a built-in checking process avoids interpreting a framing error as a new start bit; a framing error results in the addition of one-half a bit time to the point at which the search for the next start bit is begun.

The Z80 DART does not require symmetric Transmit and Receive Clock signals — a feature that allows it to be used with a Z80 CTC or any other clock source. The transmitter and receiver can handle data at a rate of 1, 1/16, 1/32 or 1/64 of the clock rate supplied to the Receive and Transmit Clock inputs. When using Channel B, the bit rates for transmit and receive operations must be the same because $\overline{RxC}$ and $\overline{TxC}$ are bonded together ($\overline{RxTxCB}$).

Fig. 3 - BLOCK DIAGRAM



NATIONAL SEMICONDUCTOR NSC800 CPU and SERIES SUPPORT ICs

Data courtesy of National Semiconductor (UK) Ltd.

The **NSC 800** is a CMOS processor and hence has a low power consumption. It bears a lot of similarity to the Z80 (to which it is about equal in computing power). It also has some of the features of the 8085, notably the serial data communication capability and multi-level interrupt structure.

It can use **Z80** and probably **8080** support chips, though some CMOS support chips are available, eg the NSC 810 RAM/IO/TIMER.

NSC800 High-Performance Low-Power Microprocessor

General Description

The NSC800 is an 8-bit microprocessor that functions as the central processing unit (CPU) in National Semiconductor's NSC800 microcomputer family. The device is fabricated using National's P²CMOS technology. This technology provides the system designer with devices that equal the performance levels of comparable NMOS products, combined with the low-power advantages of CMOS. Many system functions are incorporated on the device, such as: vectored priority interrupts, refresh control, power-save feature and interrupt acknowledge. The NSC800 is housed in a 40 pin, dual-in-line package.

Dedicated memories (NSC810 RAM-I/O Timer and NSC830 ROM-I/O) have on-chip logic for direct interface to the NSC800. In addition, National also offers a full line of P²CMOS and CMOS components to allow a full low-power solution to system designs.

Features

- Single 5V Power Supply
- Fully Compatible with Z80™ Instruction Set
- Powerful Set of 158 Instructions
- 10 Addressing Modes
- 22 Internal Registers
- Low Power: 50mW at 5V Vcc
- Multiplexed Bus Structure
- On-Chip Bus Controller and Clock Generator
- On-Chip 8-Bit Dynamic RAM Refresh Circuitry
- Standard Speed: 1.6 μs Instruction Cycle at 2.5 MHz
- Fast Version (NSC800-4): 1 μs Instruction Cycle at 4 MH_z (Availability End '82)
- Capable of Addressing 64K Bytes of Memory and 256 I/O devices
- Five Interrupt Request Lines On-Chip
- Schmitt Trigger Input on Reset
- Unique Standby-Current (Power Save) Feature

OUTPUT SIGNALS

Bus Acknowledge (BACK): Active low. BACK indicates to the bus requesting device that the CPU bus and its control signals are in the TRI-STATE mode. The requesting device may then take control of the bus and its control signals.

Address Bits 8-15 [A(8-15)]: Active high. These are the most significant 8 bits of the memory address bus, or of the input/output address. During a BREQ/BACK cycle, the A (8-15) bus is in the TRI-STATE mode.

Reset Out (RESET OUT): Active high. When RESET OUT is high, it indicates the CPU is being reset. The signal is normally used to reset the peripheral devices.

Input/Output/Memory (IO/M): An active high on the IO/M output signifies that the current machine cycle is relative to an input/output device. An active low on the IO/M output signifies that the current machine cycle is relative to memory. It is TRI-STATE during BREQ/BACK cycles.

Refresh (RFSH): Active low. The refresh output indicates that the dynamic RAM refresh cycle is in progress. RFSH goes low during T3 and T4 states of all M1 cycles.

Address Latch Enable (ALE): ALE is active only during the T1 state of M cycles and T3 state of M1 cycles. The high to low transition of ALE indicates that a valid memory /I-O/refresh address is available on the AD (0-7) lines.

Read Strobe (RD): Active low. On the trailing edge of the RD strobe, data are input to the CPU via the AD (0-7) lines. The RD line is in the TRI-STATE mode during BREQ/BACK cycles.

Write Strobe (WR): While the WR line is low, valid data are output by the CPU on the AD (0-7) lines. The WR line is in the TRI-STATE mode during BREQ/BACK cycles.

Clock (CLK): CLK is an output provided for use as a system clock. The CLK output is a square wave at one half the input frequency.

Interrupt Acknowledge (INTA): Active low. The interrupt acknowledge output is activated in the M1 cycle (S) immediately following the t state in which the INTR input is recognized. (Output is normally used to gate the interrupt response vector from the peripheral controller onto the AD (0-7) lines). It is used in two of the three interrupt modes. In mode 0, an instruction is gated onto the AD (0-7) line during INTA. In mode 2, a single interrupt response vector is gated onto the data bus.

Status (S0, S1): Bus status outputs indicate encoded information regarding the ensuing M cycle as follows:

S1	S0	STATE
0	0	HALT
0	1	WRITE
1	0	READ
1	1	(OPCODE) FETCH

NSC810 RAM-I/O-Timer

General Description

The NSC810 is a RAM-I/O-Timer device contained in a standard 40-pin, dual-in-line package. The chip, which is fabricated using P²CMOS silicon gate technology, functions as a memory, an input/output peripheral interface and a timing device. The memory is comprised of 1024 bits of static RAM organized as 128 x 8. The I/O portion consists of 22 programmable input/output bits arranged as three separate ports, with each bit individually definable as an input or output. The port bits can be set or cleared individually and can be written or read in bytes. Several types of strobed mode operations are available through Port A. The timer portion of the device consists of two programmable 16-bit binary down-counters each capable of operation in any one of six modes. Timer counts are extendable by one of the available prescale values.

Features

- 128 x 8 Random Access Memory
- Three Programmable I/O Ports
- Two 16-Bit Programmable Counter/Timers
- Single 5V Power Supply
- Very Low Power Consumption
- Fully Static Operation
- Single-Instruction I/O Bit Operations
- Timer Operation — DC to 4MHz
- Directly Compatible with NSC800 Family

NSC830 ROM-I/O; NSC831 I/O Only

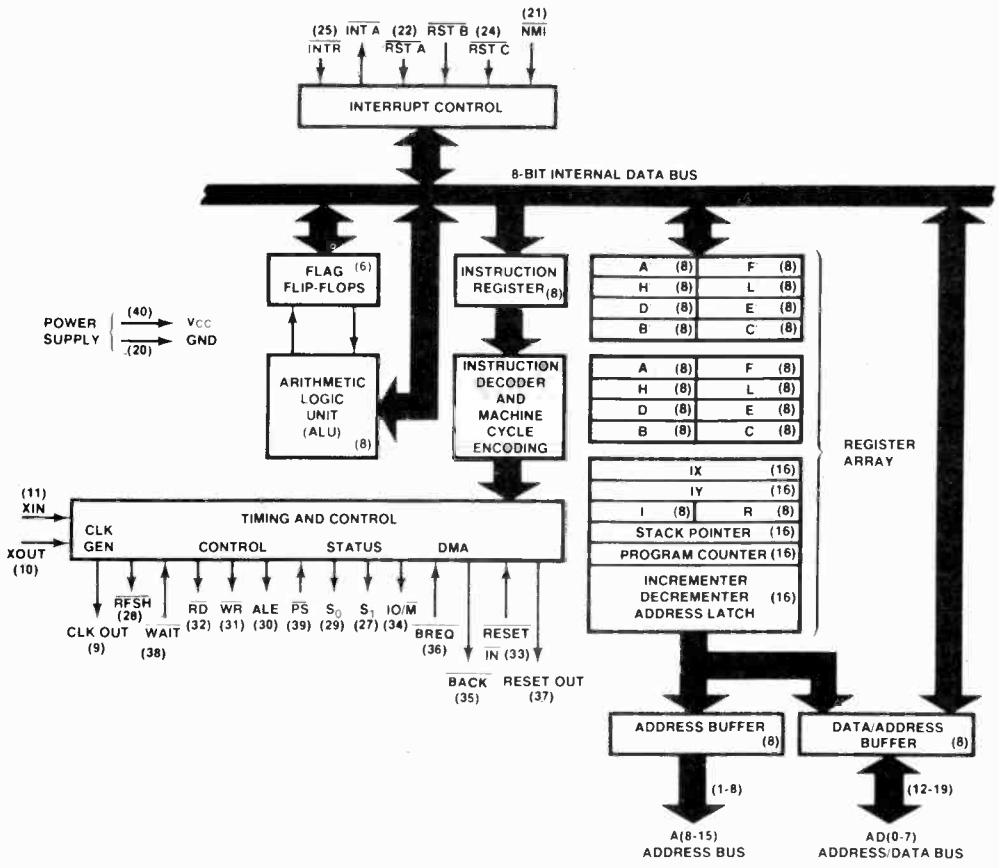
General Description

The NSC830 is a ROM-I/O device contained in a standard 40-pin, dual-in-line package. The chip, which is fabricated using P²CMOS silicon gate technology, functions as a memory, and an input/output peripheral interface device. The memory is comprised of 16,384 bits of ROM organized as 2048 x 8. The I/O portion consists of 20 programmable input/output bits arranged as three separate ports, with each bit individually definable as an input or output. The port bits can be set or cleared individually and can be written to or read from in bytes. Several types of strobed mode operations are available through Port A.

The NSC831 I/O Only is similar to the NSC830 except it has no ROM. The NSC831 is useful for prototyping work prior to ordering the NSC830, and when on-chip ROM is not required.

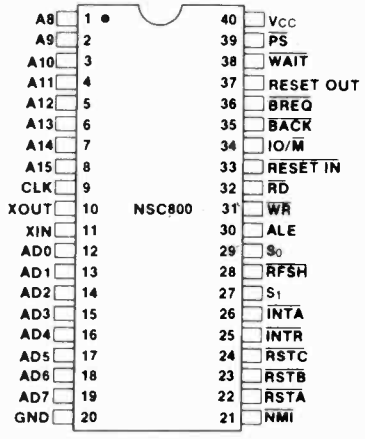
Features

- 2K x 8 Read Only Memory
- Three Programmable I/O Ports
- Single 5V Power Supply
- Very Low Power Consumption
- Fully Static Operation
- Single-Instruction I/O Bit Operations
- Directly Compatible with NSC800 Family
- Strobed Modes Available on Port A

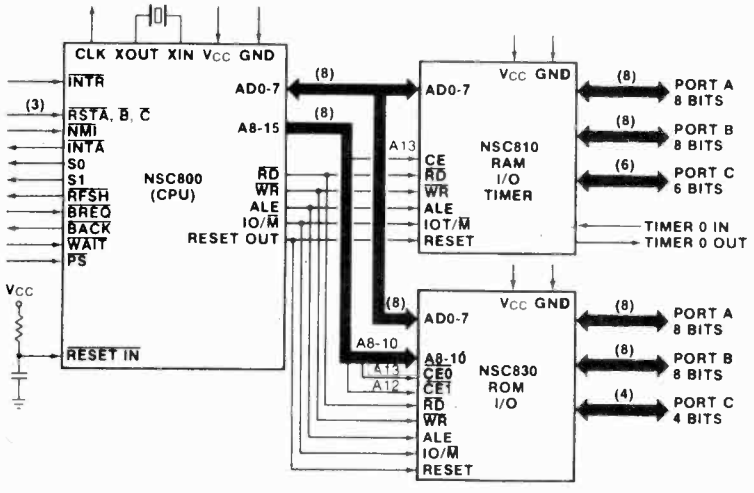


NSC800 CPU Functional Block Diagram

PIN CONFIGURATION



NSC800 Microcomputer Family Block Diagram



INPUT/OUTPUT SIGNALS

Power (V_{CC}): +5-volt supply.

Ground (GND): 0-volt reference.

Crystal (XIN, XOUT): XIN may be used as an external clock input.

Multiplexed Address/Data (AD 0-7): Active High

At $\overline{RD}$ Time:	Input data to CPU
At $\overline{WR}$ Time:	Output data from CPU
At Falling Edge of ALE Time:	Least significant byte of address during memory reference cycle. 8-bit port address during I/O reference cycle.
During $\overline{BREQ}$/ $\overline{BACK}$ Cycle:	High Impedance

INPUT PROTECTION

All inputs are protected from static charge with diode clamps to both V_{CC} and GND. Normal precautions taken with MOS devices are recommended.

TEXAS TMS 9995 CPU

Data courtesy of Texas Instruments Ltd.

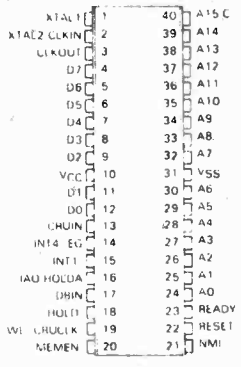
The **Texas Instruments TMS 9900** family is less of a household word than the 6800 and 8080 subspecies. It is likely to become more familiar, however, as Texas have produced low-cost evaluation systems based on this processor for use in schools – and this is where a lot of people first make their acquaintance with microprocessors.

The CPU which has been included from this family is the **TMS9995**. This is a powerful sixteen bit processor featuring high operating speeds, and a high level of integration. On the other hand, it is arranged to use standard eight bit wide memory, taking two bytes for each instruction.

FEATURES:

- 64K bytes of memory expansion
- 256 bytes of on-chip RAM.
- 12 MHz crystal oscillator.
- Separate memory, I/O, and Interrupt bus structure.
- Serial I/O via Communications Register Unit (CRU).
- 5 prioritized, vectored interrupts.
- 9900 family instruction set plus signed multiply and divide, load status, load workspace.
- Optional automatic first-wait-state generation.
- Single 5-volt operation, NMOS technology.
- 40-pin package (Plastic and Ceramic).
- 16-bit instruction word, 8-bit data bus.
- On-chip timer/event counter.
- On-chip programmable flags (16).
- Illegal opcode detection.

PIN ASSIGNMENTS:



600 mil
40 PIN

DESCRIPTION:

The TMS 9995 microprocessor is a single chip 16-bit central processing unit (CPU) with 256 bytes of on-chip RAM produced using N-channel silicon-gate MOS technology. The instruction set of the TMS 9995 includes capabilities offered in minicomputers, and its unique memory-to-memory architecture features multiple register files, resident in memory, which allow faster response to interrupts and increased programming flexibility. The separate bus structure simplifies the system design effort.

2.2 TMS 9995 ORGANIZATION

The block diagram of the TMS 9995 is shown in Figure 3.

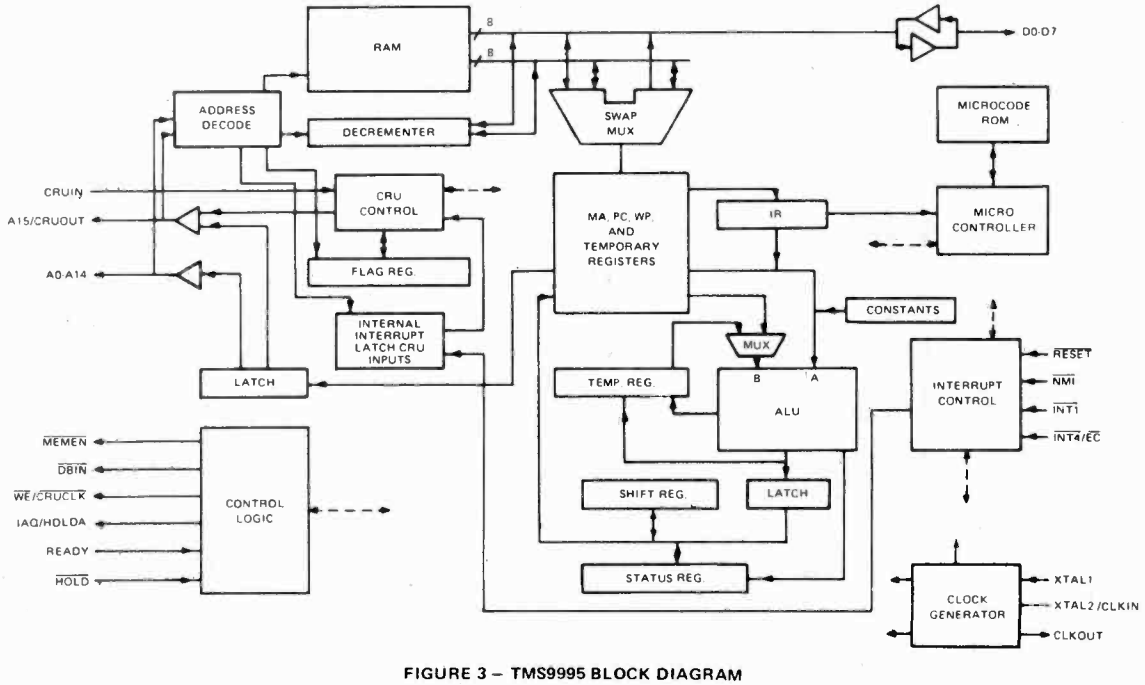


FIGURE 3 – TMS9995 BLOCK DIAGRAM

TMS 9995 Interrupts

The TMS 9995 implements seven prioritized, vectored interrupts, some of which are dedicated to predefined functions and the remaining are user-definable.
The TMS 9995 will grant interrupt requests only between instructions (except for Level 0 Reset), which will be granted whenever it is requested, i.e., in the middle of an instruction). The TMS 9995 performs additional functions for certain interrupts, and these functions will be detailed in subsequent sections. The basic sequence that the TMS 9995 performs to service all interrupt requests is as follows:

(1) Prioritize all pending requests and grant the request for the highest priority interrupt that is not masked by the current value of the interrupt mask in the status register or the instruction that has just been executed. (See Section 4.5 for these instructions.)

(2) Make a context switch using the trap vector specified for the interrupt being granted.

(3) Reset ST7 through ST11 in the status register to zero, and change the interrupt mask (ST12 through ST15) as appropriate for the level of the interrupt being granted.

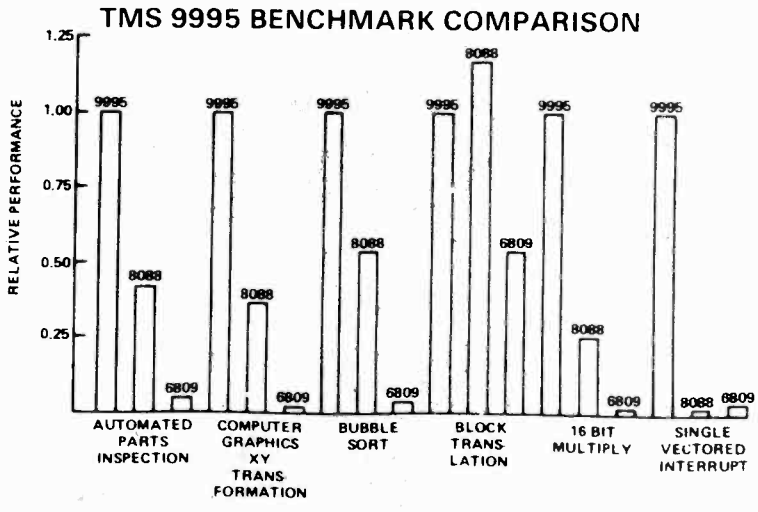
(4) Resume execution with the instruction located at the new address contained in the PC, and using the new WP. All interrupts will be disabled until after this first instruction is executed, unless: (a) $\overline{\text{RESET}}$ is requested, in which case it will be granted, or (b) the interrupt being granted is the MID request and the $\overline{\text{NMI}}$ interrupt is requested simultaneously (in which case the NMI request will be granted before the first instruction indicated by the MID trap vector is executed.)

This sequence has several important characteristics. First of all, for those interrupts that are maskable with the interrupt mask in the status register, the mask will get changed to a value that will permit only interrupts of higher priority to interrupt their service routines. Secondly, status bit ST10 (overflow interrupt enable) is reset to zero by the servicing of any interrupt so that overflow interrupt requests cannot be generated by an unrelated program segment. Thirdly, the disabling of other interrupts until after the first instruction of the service routine is executed permits the routine to disable other interrupts by changing the interrupt mask with the first instruction. (The exception with MID and NMI is explained in Section 2.3.2.2.1.) Lastly, the vectoring and prioritizing scheme of the TMS 9995 permits interrupts to be automatically nested in most cases. If a higher priority interrupt occurs while in an interrupt service routine, a second context switch occurs to service the higher priority interrupt. When that routine is complete, a return instruction (RTWP) restores the saved context to complete processing of the lower priority interrupt. Interrupt routines should, therefore, terminate with the return instruction to restore original program parameters.

ADDRESSING MODES

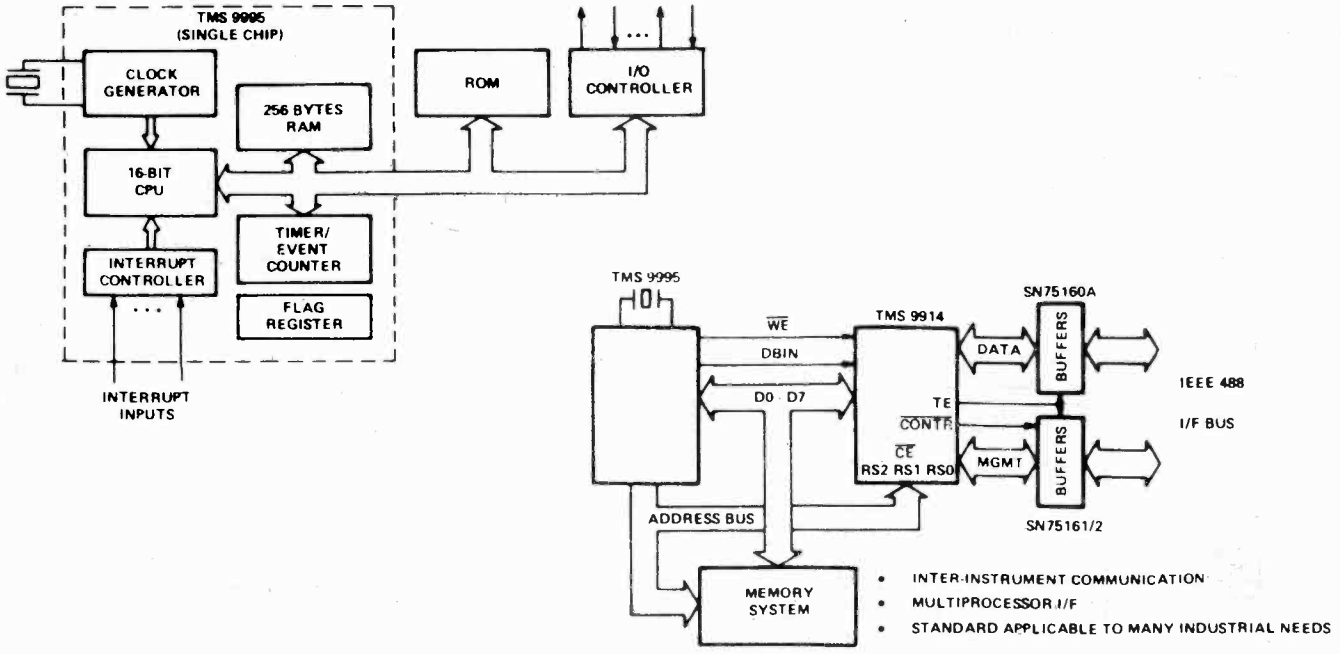
The TMS 9995 instructions contain a variety of available modes for addressing random memory data, e.g., program parameters and flags, or formatted memory data (character strings, data lists, etc.). These addressing modes are:

- Workspace Register Addressing
- Workspace Register Indirect Addressing
- Workspace Register Indirect Auto Increment Addressing
- Symbolic (Direct) Addressing
- Indexed Addressing
- Immediate Addressing
- Program Counter Relative Addressing
- CRU Relative Addressing



TMS9995 CPU

APPLICATION:



ORDERING INFORMATION:

TMS 9995 NL – 12 MHz PLASTIC 40-PIN DIL (0/70°C)
TMS 9995 JDL – 12 MHz CERAMIC 40-PIN DIL (0/70°C)

"FOR MINIMUM CHIP/HIGH PERFORMANCE MICROPROCESSOR SYSTEMS"

FEATURE	BENEFIT
256 BYTES ON-CHIP RAM	• ELIMINATES NEED FOR EXTERNAL RAM IN SMALL SYSTEMS • INCREASED SYSTEM THROUGHPUT
ON-CHIP TIMER/EVENT COUNTER	• REDUCE I/O PACKAGES
SEVEN PRIORITIZED INTERRUPTS	• EASE OF IMPLEMENTATING CONTROLLERS
AUTO WAIT STATE	• SYSTEM IMPLEMENTATION WITH SLOW MEMORIES W/O EXT. LOGIC
16-BIT INSTRUCTION SET	• COMPUTATIONAL PRECISION & EXECUTION SPEED
LOAD STATUS & WP REGISTER INSTRUCTION	• FACILITATES USE OF HLL

FEATURES COMPARISON

	9995	6809	8088
ON-CHIP RAM	256	0	0
ON-CHIP TIMER	YES	NO	NO
ON-CHIP PROGRAMMABLE FLAGS	YES	NO	NO
ON-CHIP INTERRUPT CONTROLLER	YES	NO	NO
ON-CHIP CLOCK	YES	YES	NO
BIT SERIAL I/O	YES	NO	NO
INSTRUCTION PREFETCH	YES	NO	YES
PRIORITIZED INTERRUPTS	7	2	2
NUMBER OF INTERRUPTS	7	3	3
CLOCK FREQUENCY	12 MHz	2 MHz	5 MHz
MINIMUM CHIP COUNT	3	4	8

BENCHMARK* COMPARISON

	AUTOMATED PARTS INSPECTION (sec)	COMPUTER GRAPHICS XY TRANSFORM (sec)	BUBBLE SORT (ms)	BLOCK TRANSLATION (ms)	16 BIT MULTIPLY (μs)	SINGLE VECTORED INTERRUPT (μs)
9995 (12 MHz) w/120 ns PROM	0.666	0.863	1.240	1.767	10.00**	8.00
9995 (12 MHz) w/450 ns EPROM	0.950	1.081	1.956	2.696	12.67	10.67
8088 (5 MHz) w/450 ns EPROM	1.596	2.402	2.254	1.522	40.8	77.6
6809 (2 MHz) w/450 ns EPROM	9.67	57.1	2.376	3.01	91.9	27.6
9900 (3 MHz) w/450 ns EPROM	2.053	2.709	3.068	4.593	22.0	12.0

* Intel application note AFN01551A Intel Corporation Santa Clara California 1980

** 7.67 μs if multiplicands don't have to be saved

AMI S9902 ACC

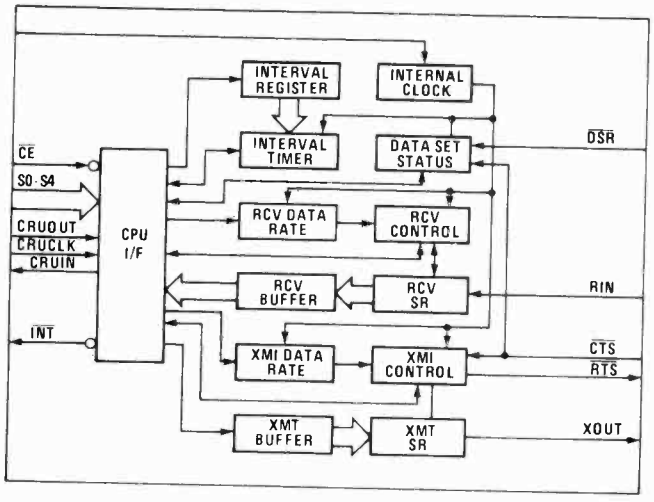
Data courtesy of AMI Microsystems Ltd.

Features

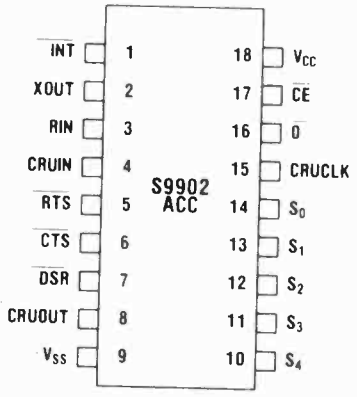
- 5- to 8-Bit Character Length
- 1, 1½, or 2 Stop Bits
- Even, Odd, or No Parity
- Fully Programmable Data Rate Generation
- Interval Timer with Resolution from 64 to 16,320 μs
- Fully TTL Compatible, Including Single Power Supply

General Description

The S9902 Asynchronous Communication Controller (ACC) is a peripheral device for the S9900 family of microprocessors. The ACC provides an interface between the microprocessor and a serial asynchronous communication channel, performing the timing and data serialization and deserialization, thus facilitating the control of the asynchronous channel by the microprocessor.



S9902 Pin Configuration



MOSTEK MK3805 Real Time Clock/Scratchpad RAM

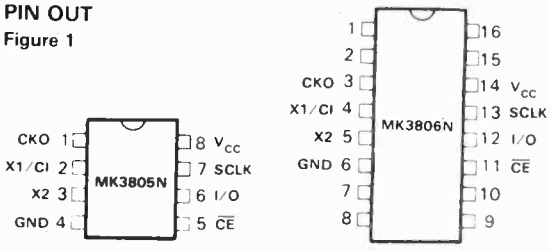
Data courtesy of Mostek UK Ltd.

- Serial I/O for minimum pin count (8 pins)
- 24 x 8 RAM for scratchpad data storage
- High speed shift clock independent of crystal oscillator frequency
- Single byte or multiple byte (Burst Mode) data transfer capability for read or write of clock or RAM data.
- $I_{CC} \leq 2mA$ (V_{CC} = 5 V)
- $+3V \leq V_{CC} \leq 9.5V$

Many microprocessor applications require a real-time clock and/or memory that can be battery powered with very low power drain. The MK3805N/MK3806N are specifically designed for these applications. The device contains a real-time clock/calendar, 24 bytes of static RAM, an on-chip oscillator, and it communicates with the microprocessor via a simple serial interface. The MK3805N/MK3806N are fabricated using CMOS technology, thus insuring very low power consumption.

The real-time clock/calendar provides seconds, minutes, hours, day, date, month, and year information to the microprocessor. The end of the month date is automatically adjusted for months with less than 31 days, including correction for leap year every 4 years. The clock operates in either the 24 hour or 12 hour format with an AM/PM indicator.

PIN OUT
Figure 1



PIN 3805N	PIN 3806N	NAME	DESCRIPTION
1	3	CKO	Buffered Sytem Clock Output
2	4	X1/C1	Crystal or External Clock Input
3	5	X2	Crystal Input
4	6	GND	Power Supply Pin
5	11	$\overline{CE}$	Chip Enable for Serial I/O Transfer
6	12	I/O	Data Input/Output Pin
7	13	SCLK	Shift Clock for Serial I/O Transfer
8	14	V_{CC}	Power Supply Pin

A separately programmable divider provides several different output frequencies for any given crystal frequency. This feature can eliminate having to use a separate crystal or external oscillator for the microprocessor, thereby reducing system cost.

AMI S6508 1024 x 1 Static RAM

Data courtesy of AMI Microsystems Ltd.

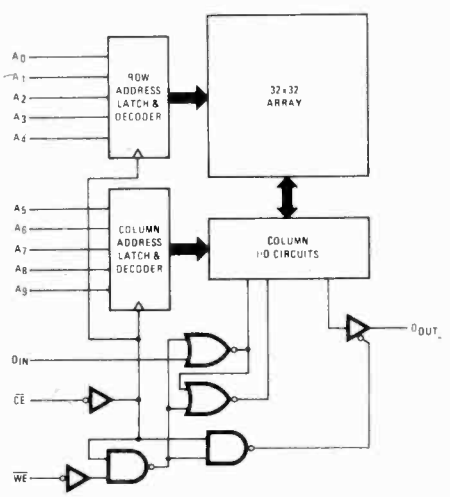
Features

- Ultra Low Standby Power
- S6508 Completely TTL Compatible
- S6508A Completely CMOS Compatible
- 4V to 11V Operation (S6508)
- Data Retention at 2V
- Three-State Output
- Low Operating Power: 10mW @ 1MHz (5V)
- Fast Access Time: 185ns @ 10V

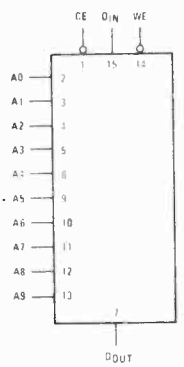
General Description

The AMI S6508 family of 1024×1 bit static CMOS RAMs offers ultra low power dissipation with a single power supply. The device is available in two versions. The basic part (S6508) operates on 5V and is directly TTL compatible on all inputs and the three-state output. The S6508 "A" operates from 4V to 11V and is fully CMOS compatible. The data is stored in ultra low power CMOS static RAM cells (six transistor). The stored data is read out nondestructively and is the same polarity as the original input data. The address is buffered by on-chip address registers. These internal registers are latched by the HIGH to LOW transition of chip enable ($\overline{CE}$). The write enable and chip enable functions are designed such that either separate or common data I/O operations can be easily implemented for maximum design flexibility.

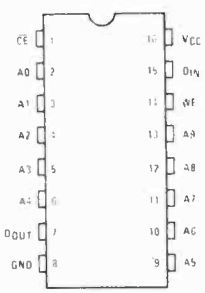
Block Diagram



Logic Symbol



Pin Configuration



Pin Names

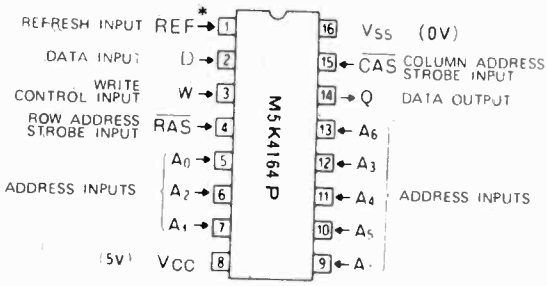
A_0—A_9	Address Inputs
D_{IN}	Data Input
D_{OUT}	Data Output
$\overline{CE}$	Chip Enable
$\overline{WE}$	Write Enable
V_{CC}	Power Supply

Mitsubishi M5K 4164 536 x 1 Static RAM

Data courtesy of Mitsubishi Electric (UK) Ltd.

DESCRIPTION

This is a family of 65 536-word by 1-bit dynamic RAMs, fabricated with the high performance N-channel silicon-gate MOS process, and is ideal for large-capacity memory systems where high speed, low power dissipation, and low costs are essential.



* If the pin 1 ($\overline{REF}$) function is not used, pin 1 may be left open (not connect).

FEATURES

- Standard 16-pin package
- Single 5V ±10% supply
- Low standby power dissipation: 22mW (max)
- Low operating power dissipation:
  M5K4164P-15 275mW (max)
  M5K4164P-20 250mW (max)
- Unlatched output enables two-dimensional chip selection and extended page boundary
- Early-write operation gives common I/O capability
- Read-modify-write, $\overline{RAS}$-only refresh, and page-mode capabilities
- All input terminals have low input capacitance and are directly TTL-compatible
- Output is three-state and directly TTL-compatible
- 128 refresh cycles every 2ms
  (16K dynamic RAMs M5K4116P, S compatible)
- Pin 1 controls automatic and self-refresh mode
- $\overline{CAS}$ controlled output allows hidden refresh, hidden automatic refresh and hidden self-refresh
- Output data can be held infinitely by $\overline{CAS}$

MOSTEK MK 4027 4096 x 1 Static RAM

Data courtesy of Mostek UK Ltd.

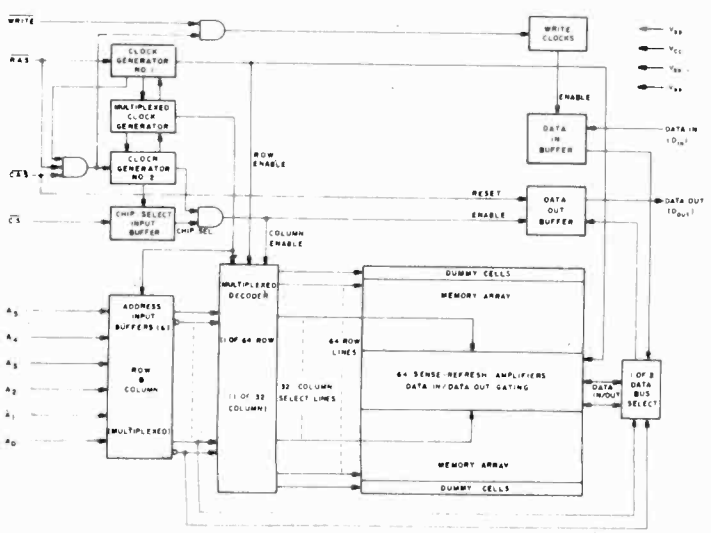
DESCRIPTION

The MK 4027 is a 4096 word by 1 bit MOS random access memory circuit

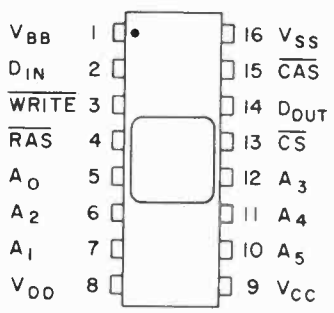
FEATURES

- ☐ Industry standard 16-pin DIP (MK 4096) configuration
- ☐ 120ns access time, 320ns cycle (MK4027-1)
  150ns access time, 320ns cycle (MK4027-2)
  200ns access time, 375ns cycle (MK4027-3)
- ☐ ±10% tolerance on all supplies (+12V, ±5V)
- ☐ ECL compatible on V_{BB} power supply (−5.7V)
- ☐ Low Power: 462mW active (max)
  27mW standby (max)
- ☐ Improved performance with "gated $\overline{CAS}$", "$\overline{RAS}$ only" refresh and page mode capability
- ☐ All inputs are low capacitance and TTL compatible
- ☐ Input latches for addresses, chip select and data in
- ☐ Three-state TTL compatible output
- ☐ Output data latched and valid into next cycle

FUNCTIONAL DIAGRAM



PIN CONNECTIONS



PIN NAMES

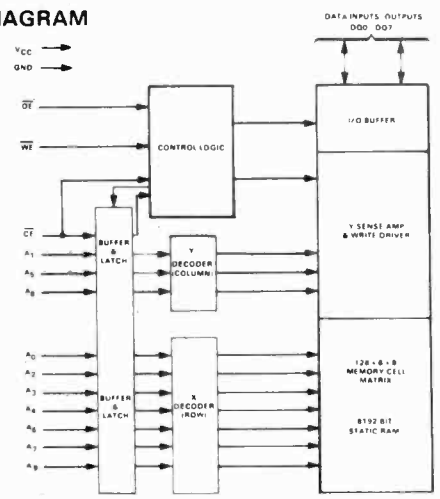
A_0-A_5	ADDRESS INPUTS
$\overline{CAS}$	COLUMN ADDRESS STROBE
$\overline{CS}$	CHIP SELECT
D_{IN}	DATA IN
D_{OUT}	DATA OUT
$\overline{RAS}$	ROW ADDRESS STROBE
$\overline{WRITE}$	READ/WRITE INPUT
V_{BB}	POWER (−5V)
V_{CC}	POWER (+5V)
V_{DD}	POWER (+ 12V)
V_{SS}	GROUND

MOSTEK MK4118 1K × 8 STATIC RAM

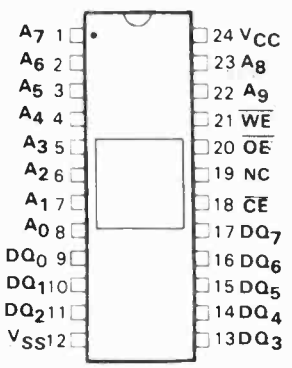
FEATURES

- □ Static operation
- □ Organization: 1K x 8 bit RAM JEDEC pinout
- □ 24/28 pin ROM/PROM compatible pin configuration
- □ $\overline{CE}$ and $\overline{OE}$ functions facilitate bus control

BLOCK DIAGRAM
Figure 1



PIN CONNECTIONS
Figure 2



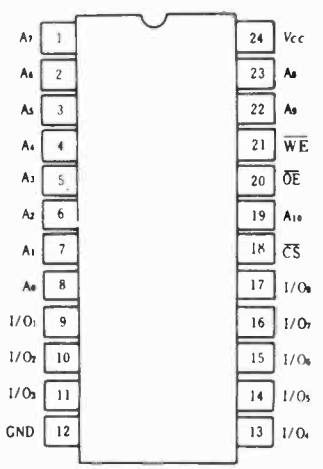
HITACHI HD6116 2K × 8 STATIC RAM

2048-word × 8-bit High Speed Static CMOS RAM

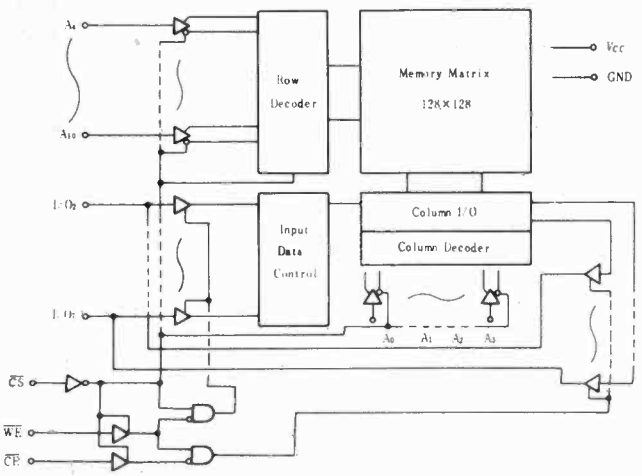
■ FEATURES

- High speed: Fast Access Time 100ns/120ns/150ns/200ns (max.)
- Low Power Standby and Low Power Operation — Standby: 100μW (typ.); Operation: 15mW (typ.) (f = 1MHz)
- Single 5V Supply and High Density 24 Pin Package
- Completely Static RAM: No clock or Timing Strobe Required
- Directly TTL Compatible: All Input and Output
- Pin Out Compatible with Standard 16K EPROM/MASK ROM
- Equal Access and Cycle Time

■PIN ARRANGEMENT



■FUNCTIONAL BLOCK DIAGRAM



AMI S5101 256 × 4 STATIC RAM

Data courtesy of AMI Microsystems Ltd.

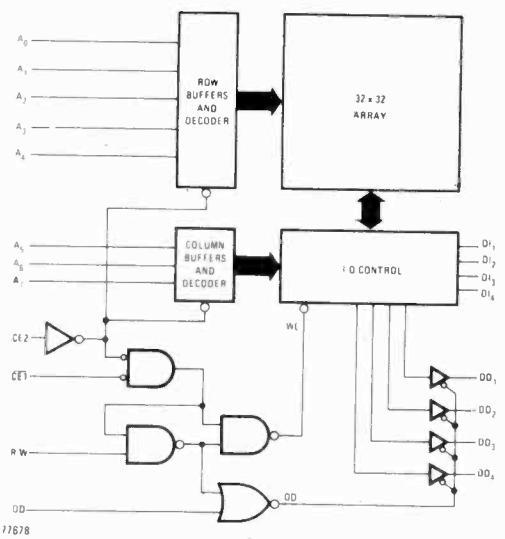
General Description

The AMI S5101 family of 256× 4-bit ultra low power CMOS RAMs offers fully static operation with a single +5 volt power supply.

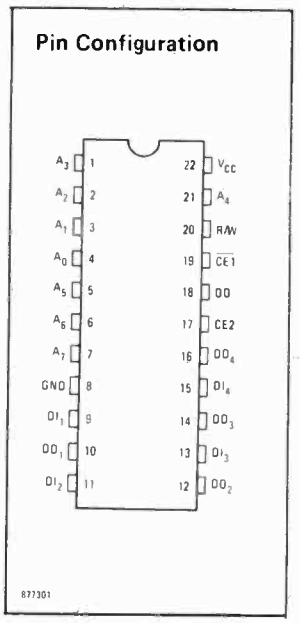
Features

- ☐ Ultra Low Standby Power
- ☐ Data Retention at 2V (L Version)
- ☐ Single +5V Power Supply
- ☐ Completely Static Operation
- ☐ Completely TTL Compatible Inputs
- ☐ Three-State TTL Compatible Outputs

Block Diagram



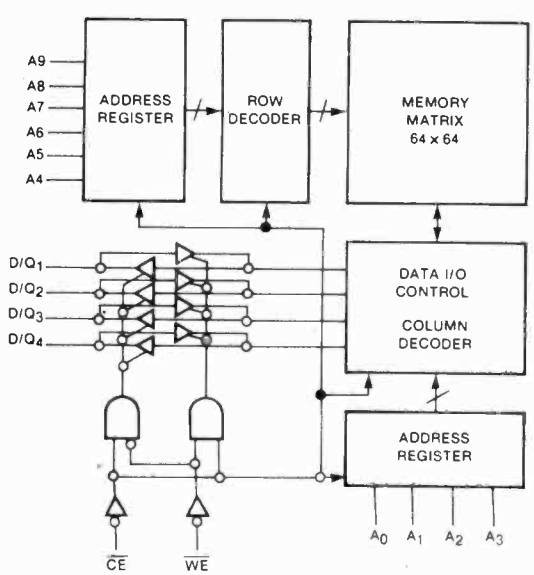
Pin Configuration



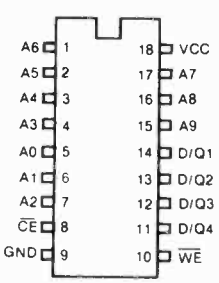
AMI S6514 1024×4 Static RAM

Data courtesy of AMI Microsystems Ltd.

Block Diagram



Pin Connections



Features

- ☐ Low Power Standby—1mW MAX
- ☐ TTL Compatible Inputs/Outputs
- ☐ Three-State Outputs
- ☐ On-Chip Address Registers
- ☐ Data Retention @ 2V
- ☐ Standard 18 pin Package/Pinouts

General Description

The AMI S6514 is a 1024x4 static CMOS RAM offering low power and static operation with a single +5 volt power supply. All inputs and outputs are TTL compatible. The common Data I/O pins allow direct interface with common bus systems.

Battery-backup design is simplified by use of $\overline{CE}$, which when HIGH, allows the other inputs to float.

MITSUBISHI M5L 2716K EPROM

Data courtesy of Mitsubishi Electric (UK) Ltd.

DESCRIPTION

These are ultraviolet-light erasable and electrically re-programmable 16 384-bit (2048-word by 8-bit) EPROMs. They incorporate N-channel silicon-gate MOS technology, and are designed for microprocessor programming applications.

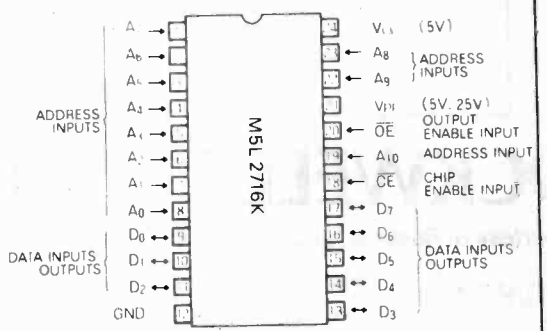
FEATURES

- Fast programming : 100s/16 384 bits (typ)
- Access time M5L 2716K : 450ns (max)
  M5L 2716K-65 : 650ns (max)
- Static circuits are used throughout
- Inputs and outputs TTL-compatible in read and program modes
- Single 5V power supply for read mode
  (25V power supply required for program)
- Low power dissipation: Operating : 525mW (max)
  Standby : 132mW (max)
- Single-location programming
  (requires one 50ms pulse/address)

HANDLING PRECAUTIONS

1. Sunlight and fluorescent light may contain ultraviolet light sufficient to erase the programmed information. For any operation in the read mode, the transparent window should be covered with opaque tape.
2. High voltages are used when programming, and the conditions under which it is performed must be carefully controlled to prevent the application of excessively high voltages. Specifically, the voltage applied to V_{PP} should be kept below 26V including overshoot. Special precautions should be taken at the time of power-on.
3. Before erasing, clean the surface of the transparent lid to remove completely oily impurities or paste, which may impede irradiation and affect the erasing characteristics.

PIN CONFIGURATION (TOP VIEW)



Outline 24K10

FUNCTION

Read

Set the $\overline{CE}$ and OE terminals to the read mode (low-level). Low-level input to $\overline{CE}$ and $\overline{OE}$ and address signals to the address inputs ($A_0 \sim A_{10}$) make the data contents of the designated address location available at the data inputs/outputs ($D_0 \sim D_7$). When the $\overline{CE}$ or $\overline{OE}$ signal is high, data inputs/outputs ($D_0 \sim D_7$) are in a floating state.

When the $\overline{CE}$ signal is high, the device is in the standby mode or power-down mode.

Programming

The chip enters the programming mode when 25V is supplied to the V_{PP} power supply input and $\overline{OE}$ is at high-level. A location is designated by address signals $A_0 \sim A_{10}$, and the data to be programmed must be applied at 8 bits in parallel to the data inputs $D_0 \sim D_7$. A program pulse to the $\overline{CE}$ at this state will effect programming. Only one programming pulse is required, but its width must satisfy the condition $45ms \leq t_{w(CE)} \leq 55ms$.

Erase

Erase is effected by exposure to ultraviolet light with a wavelength of 2537Å at an intensity of approximately 15Ws/cm^2.

Switching Characteristics (Ta = 25 ± 5 °C, V_{CC} = 5 V ± 5 %, V_{PP} = 25 ± 1 V, unless otherwise noted)

Symbol	Parameter		Test conditions	Limits			Unit
				Min	Typ	Max	
$t_{V(OE)}$	Data valid time after output enable			0		120	ns
$t_{a(OE)}$	Output enable access time	M5L 2716K				150	ns
		M5L 2716K-65				300	ns

Timing Requirements (Ta = 25 ± 5 °C, V_{CC} = 5 V ± 5%, V_{PP} = 25 ± 1 V unless otherwise noted)

Symbol	Parameter	Test conditions	Limits			Unit
			Min	Typ	Max	
$t_{su(A-CE)}$	Address setup time before chip enable		2			μs
$t_{su(OE-CE)}$	Output enable setup time before chip enable		2			μs
$t_{su(DQ-CE)}$	Data input setup time before chip enable		2			μs
$t_{h(CE-A)}$	Address hold time after chip enable		2			μs
$t_{h(CE-OE)}$	Output enable hold time after chip enable		2			μs
$t_{h(CE-DQ)}$	Data input hold time after chip enable		2			μs
$t_{w(CE)}$	Chip enable pulse width		45	50	55	ms
$t_{r(CE)}$	Chip enable pulse rise time		5			ns
$t_{f(CE)}$	Chip enable pulse fall time		5			ns

ROCKWELL R87C32 EPROM

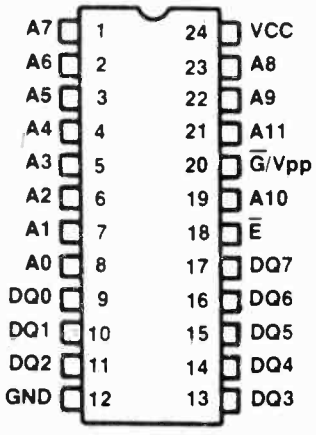
Data courtesy of Rockwell International Electronic Devices.

DESCRIPTION

The Rockwell R87C32 is a 4K × 8 (32,768 bits) ultraviolet (UV) light erasable programmable read-only-memory (EPROM). It is manufactured using CMOS technology for low power dissipation in both active and standby operating modes. Single 5V operation allows simple circuit design in runtime environments.

FEATURES

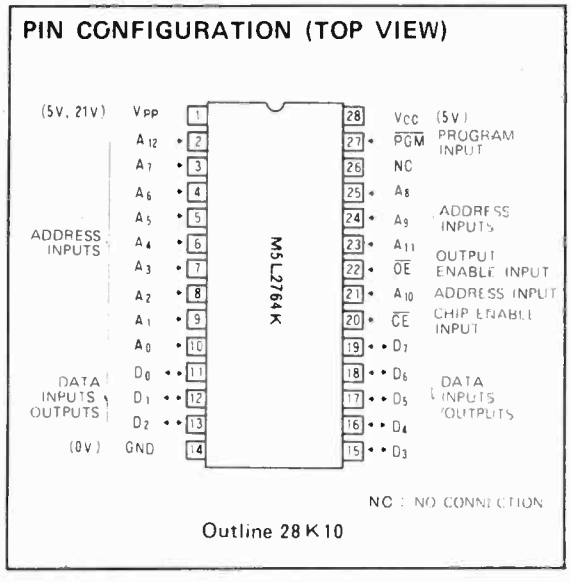
- Inputs and three-state outputs TTL compatible during both read and program mode
- Standard 24-pin dual-in-line package
  - —Pin compatible with INTEL 2732A EPROM
- 4096 × 8 organization
- JEDEC approved pin-out
- Low Power
  - —Active: 132 mW (max)
  - —Standby: 525 μW (max)
- Access time
  - —R87C32-45 450 ns (max)
  - —R87C32-35 350 ns (max)
- Single 5V power supply
- Static operation
  - —no clocks required



R87C32 Pin Configuration

MITSUBISHI M5L 2764K EPROM

Data courtesy of Mitsubishi Electric (UK) Ltd.



Outline 28K10

DESCRIPTION

The Mitsubishi M5L2764K is a high-speed 65536-bit ultraviolet erasable and electrically reprogrammable read only memory. It is suitable for microprocessor programming applications where rapid turn-around is required. The M5L2764K is fabricated by N-channel double poly-silicon gate technology and is available in a 28-pin DIL package with a transparent lid.

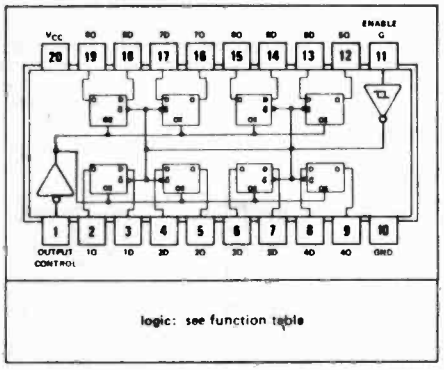
FEATURES

- 8192 Word x 8-bit Organization
- Access Time
  - M5L2764K-2 200 ns (Max)
  - M5L2764K 250 ns (Max)
  - M5L2764K-3 300 ns (Max)
- Two Line Control $\overline{OE}$, $\overline{CE}$
- Low Power Current (I_{CC}) Active . . . . . 150 mA (Max)
  - Standby . . . 35 mA (Max)
- Single 5V Power Supply
- 3-State Output Buffer
- Input and Output TTL-Compatible in Read and Program Mode
- Standard 28-pin DIL Package
- Single Location Programming with One 50 ms Pulse
- Interchangeable with INTEL 2764

TEXAS 74LS373 Latch

Data courtesy of Texas Instruments Ltd.

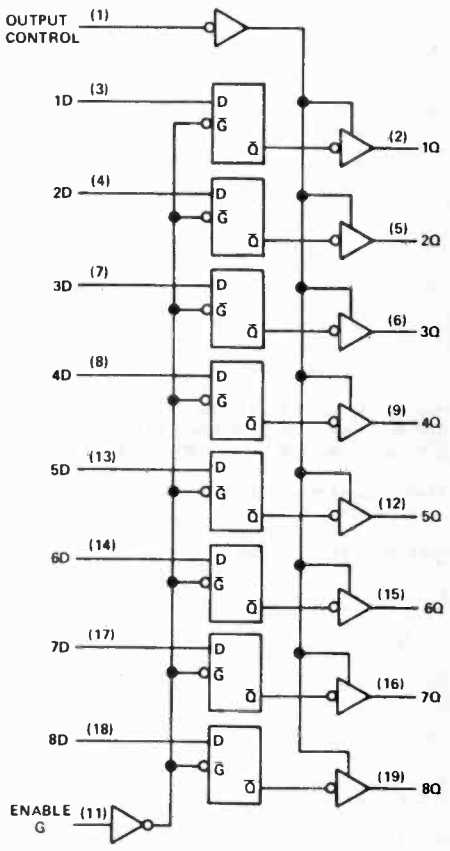
'LS373, 'S373
TRANSPARENT LATCHES



'LS373, 'S373
FUNCTION TABLE

OUTPUT CONTROL	ENABLE G	D	OUTPUT
L	H	H	H
L	H	L	L
L	L	X	Q_0
H	X	X	Z

- Choice of 8 Latches or 8 D-Type Flip-Flops In a Single Package
- 3-State Bus-Driving Outputs
- Full Parallel-Access for Loading
- Buffered Control Inputs



NATIONAL SEMICONDUCTOR INS 8073 CPU/MICROINTERPRETER

The 8073 series is a development of the SC/MPZ. It is a family of single or two chip processors. The 8073 has TINY BASIC in an internal ROM, which allows easy use for people accustomed to personal computers.

Data courtesy National Semiconductors

General Description

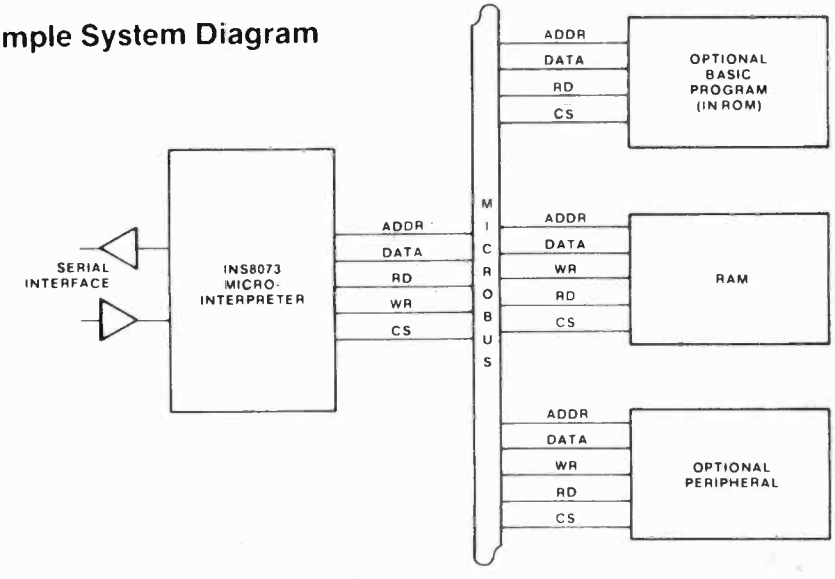
The INS8073, National's Tiny BASIC Microinterpreter, operates to provide a high level, easy to use language for performing control and computation functions in the user's system. The device is a self-contained CPU and Tiny BASIC interpreter on a single chip.

Designed for use in control applications, the Microinterpreter enables the user to write and debug programs on-line. The Microinterpreter executes source code directly, thus avoiding the need to translate the source code into machine language. The advantage of this approach is easier source code manipulation (because the source is always available), and instant revision of the program when errors are detected.

The NSC Tiny BASIC interpreter (resident in the INS8073) executes the user's source programs from read/write memory or ROM PROM automatically. The program statements are interpreted and executed line-by-line.

The INS8073 is a programmed version of the INS8072, a 2.5K x 8 mask-programmable ROM and CPU. It provides complete TTL compatibility and uses a single 5V power supply. Faster and more sophisticated computing power is available through multiprocessing.

Simple System Diagram



INS8073 CPU/Microinterpreter

Features

- Reduces Software Effort in Microcomputer Applications
- Provides Easy Source Code Manipulation and Instant Program Revision
- Allows Immediate Mode Execution of Program Statements to Assist Program Checkout
- Provides Quick Sketches of Control Algorithms
- Facilitates User Hardware Checkout (Faster Than Assembly Language)

INTRODUCTION

National Semiconductor's INS8073 is a member of the INS8070 Family with an on-chip NSC Tiny BASIC interpreter. This interpreter resides in the chip's internal ROM and offers the user the ability to program in a high level language in an absolute minimum system. The only other parts needed to program in NSC Tiny BASIC from a terminal are some RAM and RS-232 or TTY level translators. NSC Tiny BASIC is the revised version of NIBL/National's Industrial BASIC Language.

For pinout details of the INS8073, refer to the INS8070 Data Sheet, or the 70-Series User's Manual. For a further description of the NSC Tiny BASIC language, refer to the NSC Tiny BASIC User's Manual.

COMMAND SUMMARY

NEW expr: Establishes a new start-of-program address equal to the value of 'expr'. NSC Tiny BASIC then executes its initialization sequence. If the value of 'expr' points to a ROM address, the NSC Tiny BASIC program which begins at this address will be automatically executed. Program memory is not altered by this command.

NEW: Sets the end-of-program pointer equal to the start-of-program pointer so that a new program may be entered. If a program already exists at the start-of-program address, it will be lost.

RUN: Runs the current program.

CONT: Continues execution of the current program from the point where execution was suspended (via a STOP, console interrupt, or reset).

LIST [expr]: Lists the current program (optionally starting at the line number specified by [expr]).

STATEMENT SUMMARY

REM anything: Remark (no operation).

CLEAR: Initializes all variables to 0, disables interrupts, and resets all stacks (GOSUB, FOR-NEXT, DO-UNTIL).

[LET] var = expr: Assigns expression value to variable.

[LET] STAT = expr: Sets the STATUS word equal to the least significant byte of 'expr'. When the STATUS word is used to enable interrupts at the hardware level, interrupt processing will be deferred for one statment.

[LET] @factor = expr: Sets the memory location pointed to by 'factor' equal to the least significant byte of 'expr'.

[LET] $factor = "string": Assigns a string in RAM starting at the address 'factor'. Strings are terminated by a carriage return.

[LET] $factor = $factor: Memory to memory assignment (copy).

PRINT expr: Prints the value of 'expr'.

PRINT "string": Prints the string.

PRINT $factor: Prints the string starting at the memory address 'factor'.

IF expr [THEN] statements: Remainder of the program line is executed if 'expr' is true (non-zero).

FOR var = expr TO expr [STEP expr]: FOR loop initiation. Loops may be nested to four levels.

NEXT var: FOR loop termination.

DO: DO loop initiation. DO loops may be nested to eight levels.

UNTIL expr: DO loop termination.

GO TO expr: Transfer control to statement number 'expr'.

GO SUB expr: Call subroutine at statement number 'expr'. Subroutines may be nested to eight levels.

RETURN: Return from subroutine.

INPUT var: Read value from console into variable.

INPUT $factor: Read string from console into memory beginning at address 'factor'.

LINK expr: Links to an assembly language subroutine which begins at address 'expr'. A "RET" instruction in this routine will cause continuation of the NSC Tiny BASIC program.

ON 1 or 2 expr: Interrupt processing definition. When interrupt number 1 or 2 occurs, NSC Tiny BASIC will execute a GOSUB beginning at line number 'expr'. If 'expr' is zero, the corresponding interrupt is disabled at the software level.

DELAY expr: Delay for 'expr' time units (nominally milliseconds, 1-1040). DELAY 0 gives the maximum delay of 1040 milliseconds.

STOP: Terminate program execution. A message is printed and the Microinterpreter returns to COMMAND mode.

OPERATOR SUMMARY

Arithmetic operators:	addition	+
	subtraction	-
	multiplication	*
	division	/
Relational operators:	less than	<
	greater than	>
	equal to	=
	not equal to	<>
	less than or equal to	<=
	greater than or equal to	>=
Logical operators:	logical AND	AND
	logical OR	OR
	logical NOT	NOT

FUNCTION SUMMARY

@factor: The memory/peripheral address for memory-I/O read/write operations.

STAT: STATUS register.

TOP: Top-Of-Program address (first available memory address after end-of-program byte).

INC (x), DEC (x): Increment or Decrement a memory location (non-interruptable for multiprocessing).

MOD (x,y): Modulus function (remainder of x/y).

RND (x,y): Random number generator (in interval x,y).

RAM SEARCH FOLLOWING POWER-ON OR RESET

At power-on, or when the INS8073 is reset via the NRST pin, NSC Tiny BASIC automatically performs a nondestructive memory search. The search is conducted to determine the address range of the external RAM which is present. External RAM may be located anywhere in memory above the first 4K bytes (X'1000 or higher). The first 256 bytes of external RAM are used to store the Microinterpreter's variables, stacks and buffers. The remainder of the RAM may be used to store programs entered by the user. I/O devices may be memory mapped as long as they are not contiguous with RAM.

PROGRAM MEMORY FORMAT

NSC Tiny BASIC programs are stored in ASCII characters. This feature simplifies the task of checking memory resident programs. Any nulls, line feeds, or blanks between program lines are ignored by the Microinterpreter. Source programs are terminated by any byte which is not one of the ASCII characters recognized by the Microinterpreter. (The byte X'7F is usually used as the end-of-program indicator.)

BAUD RATE SELECTION

The Microinterpreter has built-in I/O routines to serially interface with a serial RS-232 terminal or TTY.

When the INS8073 is initialized, the desired baud rate is automatically selected by reading the contents of memory location X'FD00.

General Description

The INS8070-Series of microprocessors (hereinafter referred to as the 70-Series Family) is intended for use in systems requiring the economy of a single chip, the flexibility of multiprocessing bus architecture and the power of 16-bit arithmetic operations.

The 70-Series family comprises bus-oriented microprocessors with on-board ROM and RAM and built-in multi-processing logic. Low-cost systems containing data may occupy the same bus and still run at full speed, as their external bus requirements are confined to the occasional access of data. Such systems include terminals, intelligent peripherals and multiprocessing systems (e.g., the solution of polynomials in real-time).

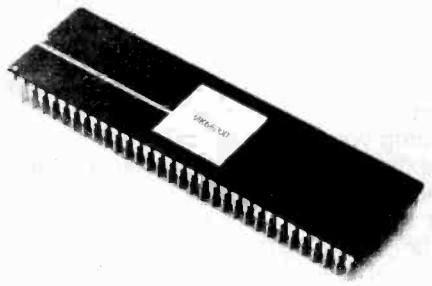
Features

- On-Board ROM
- On-Board RAM (Executable)
- 8-Bit Data Bus
- 16-Bit Address Bus (64K Addressing Capability)
- Comprehensive Set of 8 and 16-Bit Arithmetic, Logic and Stack Manipulation Instructions
- Hardware 16 x 16 Bit Multiply (37 μs) and Divide (42 μs)
- Built-in Multiprocessing and DMA Logic
- Interfaces with Memory and Standard INS8080 Peripherals at Any Clock Speed
- On-Chip Clock Generation
- Single Instruction Character Search and Single Instruction ASCII to Decimal Conversion
- Full Hardware and Software Development Systems Available on STARPLEX™ and ISE™
- Single 5-Volt Supply
- MICROBUS™ Compatible

Mostek MK68000 Advanced CPU

Data courtesy of Mostek UK Ltd.

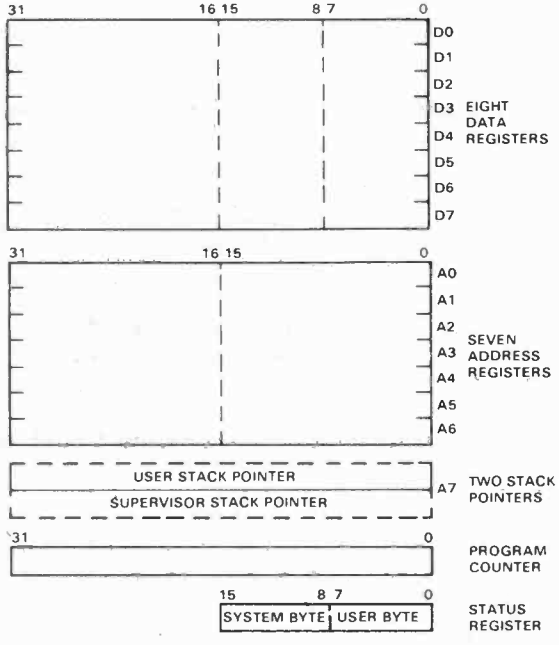
Advances in semiconductor technology have provided the capability to place on a single silicon chip a microprocessor at least an order of magnitude higher in performance and circuit complexity than has been previously available. The MK68000 is the first of a family of such VLSI microprocessors from Mostek. It combines state-of-the-art technology and advanced circuit design techniques with computer sciences to achieve an architecturally advanced 16-bit microprocessor.



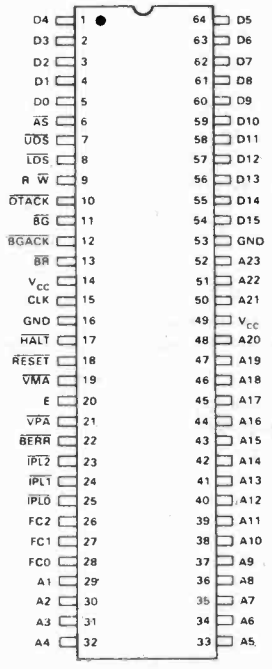
The resources available to the MK68000 user consist of the following:

- 32-Bit Data and Address Registers
- 16 Megabyte Direct Addressing Range
- 56 Powerful Instruction Types
- Operations on Five Main Data Types
- Memory Mapped I/O
- 14 Addressing Modes

PROGRAMMING MODEL



PIN ASSIGNMENT



As shown in the programming model, the MK68000 offers seventeen 32-bit registers in addition to the 32-bit program counter and a 16-bit status register. The first eight registers (D0-D7) are used as data registers for byte (8-bit), word (16-bit), and long word (32-bit) data operations. The second set of seven registers (A0-A6) and the system stack pointer may be used as software stack pointers and base address registers. In addition, these registers may be used for word and long word address operations. All 17 registers may be used as index registers.

A 23-bit address bus provides a memory addressing range of greater than 16 megabytes. This large range of addressing capability, coupled with a memory management unit, allows large, modular programs to be developed and operated without resorting to cumbersome and time consuming software bookkeeping and paging techniques.

Special emphasis has been given to the instruction set's support of structured high-level languages to facilitate ease of programming. Each instruction, with few exceptions, operates on bytes, words, and long words and most instructions can use any of the 14 addressing modes. Combining instruction types, data types, and addressing modes, over 1000 useful instructions are provided. These instructions include signed and unsigned multiply and divide, "quick" arithmetic operations, BCD arithmetic and expanded operations (through traps).

CHIP CHASING

How and where to fish for the elusive microprocessor.

General suppliers of electronic components stock many more CPU support chips – and also CPUs – that we haven't included in Digest: peripheral drivers, sound generators, line drivers (quad and dual), character generators, disk controllers, etc.

If you are designing your own systems, you may want to find out if a chip exists to do a particular job. This presupposes that you know a certain amount about microprocessors and support ICs. Asking your components supplier, leafing through a large catalogue, or (getting really serious) referring to a directory such as IC Masters (see Bibliography) may put you on the right track.

If, on the other hand, you are working with a familiar chip, or working to someone else's project design, finding somewhere to buy the devices is not normally a problem – they are commonly available, and if you have no local component supplier that stocks them, then there are plenty of mail order suppliers advertising in *Electronics Today* or *Hobby Electronics*.

Ideally, you should get a full data sheet with the chip when you buy it. Problems arise when a microprocessor (or other IC) is obtained by luck, cannibalisation or other second-hand means, or if a device is stored up for future use and the data sheet goes missing – which not infrequently happens. The constructor is then stuck until the missing information can be supplied.

ICs usually have identifying marks on the flat top: the manufacturer's name and/or logo, and a part number, consisting of letters and numbers. The letters at the beginning of the part number refer either to the manufacturer (Hitachi use 'HD' for instance) or to an industry standard, such as 'LM' for 'Linear Microcircuit'. Other figures may be a part number exclusive to a particular manufacturer, but a part of the figure – usually four digits – is very often a standard number by which the device is familiarly known: the figures '6800', for instance, will appear in the part number of most devices of this type. This is not perfectly consistent, though: some sources give their ICs numbers slightly different from similar ICs from another source. Hence the need for cross-references and specific data, especially with something as complex as microprocessors. "Made In Korea" only means that the IC is assembled in Korea – the 'chips' themselves are usually made in the USA and Europe.

Aware of this, some manufacturers' data books provide information on equivalents: Intersil's microprocessor data book, for instance contains an 'Alternative Source Index', with the Intersil equivalents to ICs from other sources, while the Farnell catalogue lists its ICs under the manufacturers' letter codes in its contents list.

Once you know what you have and (you hope) who the manufacturer is, you can turn your attention to obtaining a replacement data sheet. If the original supplier cannot help, the best place to contact is a franchised distributor for the manufacturing company. The distributors are there to make sure the goods reach the customers, and are geared to dealing with small orders and individual queries, where the manufacturing companies are not. Even contacting the wrong distributor may put you on the road for the right one.

In the last resort, contacting the Public Relations department of a manufacturing company may turn up a photocopy of the data sheet (they don't generally keep stocks handy) or a list of distributors. Manufacturers are geared to supplying orders for a few tens of thousands-off – requests for one chip or one data sheet tend to get lost in the system. We have included a list of manufacturers and selected distributors below, with a few comments where appropriate.

Distributors

This is a selected list of distributors who handle microprocessor ICs. Some of them distribute certain manufacturers' devices, as note, but may handle ICs from other sources as well.

Abacus Electronics, Kennet House, Pembroke Rd, Berks. Tel: Reading 33311.
National Semiconductor, SGS.

Access Electronic Components Ltd., Austin House, Bridge Street, Hitchin, Herts. Tel: Hitchin 31221.
Texas.

Alpha Electronic Components Ltd., 66 Wilbury Way, Hitchin, Herts. Tel: Hitchin 57244.
National Semiconductor.

Alteck Microcomponents Ltd., 22 Market Place, Wokingham, Berks. Tel: Wokingham 791579.
Mitsubishi.

Anzac Components Ltd., Burnham Lane, Slough, Berks. Tel: Slough 4701.
Hitachi.

Axiom Electronics Ltd., Unit F, Turnpike Rd., Cressex Industrial Estate, High Wycombe, Bucks. Tel: High Wycombe 442181.
Motorola.

B A Electronics Ltd., Millbrook Road, Yate, Bristol BS17 5NX. Te: (0454) 315824.
Texas.

Bytech Ltd., Unit 57, London Rd., Earley, Reading, Berks. Tel: Reading (0734) 61031.
Intel.

Celdis Ltd., 37-39 Loverrock Rd., Reading, Berks. Tel: Reading 585171.
Motorola, Mostek.

Continued/

ADDRESSES

Comway Microsystems Ltd., Market St., UK-Bracknell, Berks. Tel: Bracknell 413127.
Intel.

Crellon Electronics Ltd., 380 Bath Rd., Slough, Berks. Tel: Burnham 4434.
Motorola, SGS, Zilog.

Decade Ltd., 100 School Rd., Tilehurst, Reading, Berks. Tel: Reading 450144.
Intel.

Dialogue Distribution Ltd., Watchmore Rd., Camberley, Surrey. Tel: Camberley 682001.
Hitachi.

DTV Group, 10-12 Ernest Avenue, West Norwood, London SE27. Tel: 01-670 6166.
National Semiconductor.

Electronic Manufacturing Co. Ltd., 3B Avonburg Industrial Estate, Long Mile Rd., Dublin 12. Tel: 0001 519668.
National Semiconductor.

ESI Components Ltd., Brighton Hill Parade, Brighton Hill, Basingstoke, Hants. Tel: Basingstoke 57166.
Mitsubishi.

Farnell Electronic Components Ltd., Canal Rd., Leeds LS12 2TU. Tel: Leeds 636311.
Hitachi, National Semiconductor.

Hawke Electronics Ltd., Amotex House, 45 Hanworth Rd., Sunbury-on-Thames, Middx. Tel: 01-979 7799.
Source of technical advice on Motorola components and systems, Texas.

Hill Electronics NI Ltd., 290 Antrim Rd., Belfast, Northern Ireland.
Mostek.

Hi-Tek Distribution, Trafalgar Way, Cambridge. Tel: (0954) 81996.
National Semiconductor, Zilog.

Impulse Electronics Ltd., Croudace House, Caterham, Surrey. Tel: Caterham 40325.
Hitachi.

ITT MULTIcomponents, Edinburgh Way, Harlow, Essex. Tel: Harlow 442971.
Hitachi.

Jermyn Distribution Ltd., Vestry Estate, Otford Rd., Sevenoaks, Kent. Tel: Sevenoaks 450144.
Motorola, National Semiconductor, Intel.

A M Lock Distribution, Neville St., Middleton Rd., Oldham, Lancs. Tel: 061 652 0431.
Motorola, Mostek.

Macro Marketing Ltd., Burnham Lane, Slough, Berks. Tel: Burnham 4422.
Motorola, National Semiconductor, Zilog, Texas.

MDL Components Ltd., Fleets Industrial Estate, Willis Way, Poole, Dorset. Tel: Poole 681331.
Mitsubishi.

MEDL, East Lane Rd., North Wembley, Middx. Tel: 01-904 9307.
Intel.

Pelco Electronics Ltd., Spring Gardens, London Rd., Romford, Essex. Tel: Romford 61911.
Rockwell, SGS.

Quarndon Electronics Ltd., Slack Lane, Derby DE3 3ED. Tel: Derby 32651.
Texas.

The Radio Resistor Co., St. Martin's Industrial Estate, Cambridge Rd., Bedford MK42 0LF. Tel: Bedford 47188.
Motorola.

Rapid Recall Ltd., Rapid House, Denmark St., High Wycombe, Bucks. Tel: High Wycombe (0494) 26271.
Intel.

STC Electronics, Edinburgh Way, Harlow, Essex. Tel: Harlow 26777.
Motorola, National Semiconductor, SGS, Texas.

VSI Electronics Ltd., Roydenbury Industrial Park, Horsecroft Road, Harlow, Essex. Tel: Harlow 29666.
Texas, AMI.

Manufacturers featured in this issue:

AMI Microsystems Ltd, 108a Commercial Rd, Swindon, Wilts. Tel: Swindon 31345.
Deal through distributors.
6800 family.

Fairchild Camera & Instrument (UK) Ltd, 230 High St, Potters Bar, Herts EN6 5BU. Tel: Potters Bar 51111.
Deal through distributors.
6800 family.

Hitachi Electronic Components (UK) Ltd, 221-225 Station Rd, Harrow, Middlesex HA1 2XL. Tel: 01-861 1414.
Can deal with genuine technical enquiries direct; otherwise deal through distributors.
6800 family.

Intel Corporation (UK) Ltd, Pipers Way, Swindon, Wilts SN3 1RJ. Tel: Swindon 488388.
Deal through distributors.
8080 family.

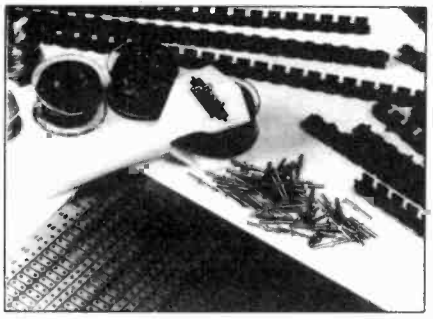
Mitsubishi Electric (UK) Ltd, London Office, 18th Floor, Centre Point, 103 New Oxford St, London WC1. Tel: 01-379 7160.
Deal through distributors.
8080 family.

Road Runner

WE OFFER A RANGE OF ELECTRONIC WIRING AIDS & HARDWARE TO SUIT BOTH THE ENTHUSIAST AND PROFESSIONAL ENGINEER.

*** QUALITY * LOW PRICES * SAME DAY SERVICE ***

WIRING SYSTEM



Kits:	
Euro Introkit (with Single Eurocard)	17.59
Project Kit (with Hobby Board)	7.99
Mini Kit 'A' (Glue Strips)	7.35
Mini Kit 'B' (Press Strips)	6.83
PCB Repair Kit	5.95
Training Kit (Cutters, Tweezers, etc.,)	9.85
Others:	
Wiring Pencil	3.30
Above with brake	3.99
TCW Pencil	3.25
Blue Bobbins/4	2.30
Green Bobbins/4	2.30
Pink Bobbins/4	2.30
Copper Bobbins/4	2.30
Mixed (one of each)	2.40
Tinned Copper Bobbins/2	1.06
Glue Strips 20	3.40
High Density Strips/20	3.60
Press Fit Strips/20	2.99
Roadrunner H T Iron	6.29
Iron Stand	4.99
Spare Copper Bit	0.70
Spare Long Life Bit	1.40
Spare Element	4.99

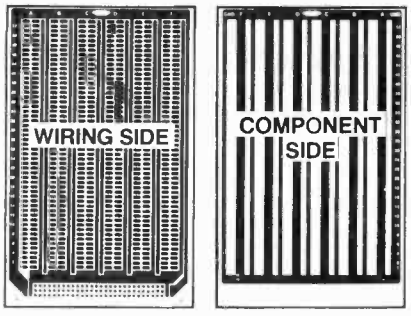
CONTACT US NOW AND ASK ABOUT OUR NEW COMPUTER ACCESSORY PRODUCT RANGE (Disk, cassettes, etc.,)

Save Time & Telephone

We welcome orders using Access or Visa

Access VISA

PROTOTYPING CARDS



Single Eurocards	4.99
Double Eurocard	9.50
Project Card (s/sided)	3.99
S100 Board	14.99
Hobby Board	1.99
Matrix Board (150mm x 115mm)	0.99

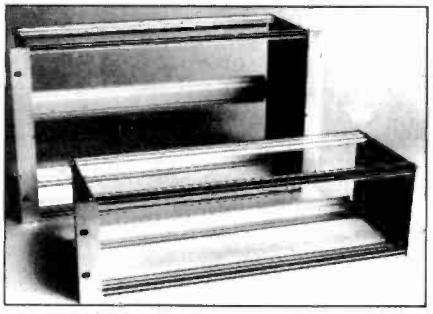
CONNECTORS

DIN 41612 2 part for Eurocard:-	
B Type (2 rows)	
32 way Plug	1.03
32 way Socket (ww)	1.56
64 way Plug	1.61
64 way Socket (ww)	2.43
C TYPE (3 rows)	
64 way Plug	1.61
64 way Socket (ww)	3.29
96 way Plug	2.21
96 way Socket (ww)	3.29
Fixing Screws 20	0.40
Fixing Nuts 20	0.46
S100 Connector	4.25
Hobby Board Connector	2.15

SOCKETS

GOLD		TINNED	
8 way	0.19	8way	0.10
14 way	0.20	14 way	0.13
16 way	0.22	16 way	0.15
18 way	0.26	18 way	0.17
20 way	0.30	20 way	0.20
24 way	0.34	24 way	0.24
28 way	0.44	28 way	0.27
40 way	0.55	40 way	0.38

SUBRACKS



3U 19"] as in	21.35
6U 19"] photo	32.25
Handi-raks	
3U Halfwidth	19.24
3U Quarterwidth	17.14
Acessories	
Guiderails	0.18
Rataining Clip	0.24
Extractor	0.28
Coverplate (top & bottom)	3.57
½ Width	1.92
¼ Width	1.16
Front Panel (3U)	8.75
Front Panel (6U)	14.17
3U panel ½ width	5.37
3U panel ¼ width	4.15
Panel Fixing 6	1.28

TOOLS etc

Weller High Temp Bit	1.71
Xersin solder (½ kg)	8.99
Desolder Braid	1.09
Microshears	3.82
Knife	0.55
Tweezers	1.71
Pin Inserter	2.19
Solder Pins 100	0.65
Spot Face Cutter	1.71
Adhesive	0.55
Wire Schedules 10	0.55
BANANA MULTIMETER	22.60

ORDER INFO

1. All prices are EXCULSIVE of VAT and carriage. Please add 5% to goods nett total to cover carriage, packing and insurance, and then add 15% VAT.

2. For any order less than £5 nett, please add 50p handling charge.

3. Please add 15% to total "exc. VAT" for EXPORT orders to cover carriage, packing and insurance.

Orders to:

Roadrunner Electronic Products Ltd.

UNIT 3, THE HASLEMERE INDUSTRIAL ESTATE,
WEYDOWN ROAD, HASLEMERE, SURREY GU27 1DW.
TEL: 0428 53850. TELEX: 859130 HAMHOU G.

Road Runner ®

The name to remember in electronics.

electronics today international
BOOK SERVICE

How to order: indicate the books required by ticking the boxes and send this page, together with your payment, to: ETI Book Service, Argus Specialist Publications Ltd, 145 Charing Cross Road, London WC2 0EE. Make cheques payable to ETI Book Service. Payment in sterling only please. All prices include P & P. Prices may be subject to change without notice.

BEGINNERS GUIDE

☐ Beginner's Guide to Basic Programming Stephenson	£5.35
☐ Beginner's Guide to Digital Electronics	£5.35
☐ Beginner's Guide to Electronics	£5.35
☐ Beginner's Guide to Integrated Circuits	£5.35
☐ Beginner's Guide to Computers	£5.35
☐ Beginner's Guide to Microprocessors	£5.35

COOKBOOKS

☐ Master IC Cookbook Hallmark	£10.15
☐ Microprocessor Cookbook M. Hordeski	£7.70
☐ IC Op Amp Cookbook Jung	£14.25
☐ PLL Synthesiser Cookbook H. Kinley	£7.70
☐ Active Filter Cookbook Lancaster	£13.40
☐ TV Typewriter Cookbook Lancaster	£11.15
☐ CMOS Cookbook Lancaster	£11.85
☐ TTL Cookbook Lancaster	£10.95
☐ Micro Cookbook Vol. 1 Lancaster	£15.30
☐ BASIC Cookbook K. Tracton	£6.00
☐ MC6809 Cookbook C. Warren	£7.25

ELECTRONICS

☐ Principles of Transistor Circuits Amos	£8.50
☐ Design of Active Filters with experiments Berlin	£11.30
☐ 49 Easy to Build Electronic Projects Brown	£6.00
☐ Electronic Devices & Circuit Theory Boylestad	£13.20
☐ How to build Electronic Kits Capel	£3.55
☐ How to Design and build electronic instrumentation Carr	£9.35
☐ Introduction to Microcomputers Daglecs	£7.20
☐ Electronic Components and Systems Dennis	£15.00
☐ Principles of Electronic Instrumentation De Sa	£11.40
☐ Giant Handbook of Computer Software	£12.95
☐ Giant Handbook of Electronic Circuits	£17.35
☐ Giant Handbook of Electronic Projects	£11.75
☐ Electronic Logic Circuits Gibson	£5.55
☐ Analysis and Design of Analogue Integrated Circuits Gray	£30.25
☐ Basic Electronics Grob	£11.30
☐ Lasers – The Light Fantastic Hallmark	£7.70
☐ Introduction to Digital Electronics & Logic Joynson	£5.25
☐ Electronic Testing and Fault Diagnosis Loveday	£7.85
☐ Electronic Fault Diagnosis Loveday	£6.25
☐ Essential Electronics A-Z Guide Loveday	£7.50
☐ Microelectronics Digital & Analogue circuits and systems Millman	£12.70
☐ 103 Projects for Electronics Experimenters Minis	£8.30
☐ VLSI System Design Muroga	£34.10
☐ Power FETs and their application Oxner	£9.40
☐ Practical Solid State Circuit Design Olesky	£25.00
☐ Master Handbook of IC Circuits Powers	£12.85
☐ Electronic Drafting and Design Raskhodoff	£22.15
☐ VOM – VTVM Handbook Risse	£8.50
☐ Video and Digital Electronic Displays Sherr	£28.85
☐ Understanding Electronic Components Sinclair	£7.50
☐ Electronic Fault Diagnosis Sinclair	£4.50
☐ Physics of Semiconductor Devices Sze	£17.35
☐ Digital Circuits and Microprocessors Taub	£32.00
☐ Active Filter Handbook	£7.60
☐ Designing with TTL Integrated Circuits Texas	£15.20
☐ Transistor Circuit Design Texas	£15.20
☐ Digital Systems: Principles and Applications Tocci	£12.95
☐ Master Handbook of Telephones Traister	£10.00
☐ How to build Metal/Treasure Locators Traister	£6.00
☐ 99 Fun to Make Electronic Projects Tymony	£8.50
☐ 33 Electronic Music Projects you can build Winston	£6.95

COMPUTERS & MICROCOMPUTERS

☐ BASIC Computer Games Ahl	£6.35
☐ From BASIC to PASCAL Anderson	£9.95
☐ Mastering Machine Code on your ZX81 T. Baker	£7.25
☐ UNIX – The Book Banaham	£8.75
☐ Z80 Microcomputer Handbook Barden	£10.95
☐ Microcomputer Maths Barden	£11.90
☐ Digital Computer Fundamentals Barter	£9.90
☐ Visicalc Book. APPLE Edition Bell	£15.55
☐ Visicalc Book. ATARI Edition Bell	£15.55
☐ Introduction to Microprocessors Brunner	£23.00
☐ Programming your APPLE II Computer Bryan	£9.25
☐ Microprocessor Interfacing Carr	£7.70
☐ Microcomputer Interfacing Handbook A/D & D/A Carr	£9.50
☐ Musical Applications of Microprocessors Chamberlain	£28.85
☐ 30 Computer Programs for the Home Owner in BASIC D. Chance	£9.25
☐ Microcomputers Dirkson	£9.30
☐ APPLE Personal Computer for Beginners Dunn	£9.50
☐ Microcomputers/Microcomputers – An Intro Gioone	£11.80
☐ Troubleshooting Microprocessors and Digital Logic Goodman	£9.25
☐ Getting Acquainted with your VIC 20 Hartnell	£8.50
☐ Getting Acquainted with your ZX81 Hartnell	£5.95
☐ Let your BBC Micro Teach you to program Hartnell	£7.90
☐ Programming your ZX Spectrum Hartnell	£8.50
☐ The ZX Spectrum Explored Hartnell	£6.95
☐ How to Design, Build and Program your own working Computer System Haviland	£9.30
☐ BASIC Principles and Practice of Microprocessors Heffer	£7.15
☐ Hints and Tips for the ZX81 Hewson	£5.25
☐ What to do when you get your hand on a Microcomputer Holtzman	£9.95
☐ 34 More Tested Ready to Run Game Programs in BASIC Horn	£7.70
☐ Microcomputer Builders' Bible Johnson	£12.40
☐ Digital Circuits and Microcomputers Johnson	£14.55
☐ PASCAL for Students Kemp	£7.20
☐ The C – Programming Language Kernighan	£18.20
☐ CDBDL Jackson	£9.25
☐ The ZX81 Companion Maunder	£9.50
☐ Guide to Good Programming Practice Meek	£6.40
☐ Principles of Interactive Computer Graphics Newman	£13.95
☐ Theory and Practice of Microprocessors Nicholas	£11.35
☐ Exploring the World of the Personal Computer Nilles	£12.95
☐ Microprocessor Circuits Vol. 1. Fundamentals and Microcontrollers Noll	£9.80
☐ Beginner's Guide to Microprocessors Parr	£5.35
☐ Microcomputer Based Design Peatman	£11.30
☐ Digital Hardware Design Peatman	£9.80
☐ BBC Micro Reavealed Ruston	£9.45
☐ Handbook of Advanced Robotics Safford	£14.45
☐ 1001 Things to do with your own personal computer Sawusch	£8.50
☐ Easy Programming for the ZX Spectrum Stewart	£7.15
☐ Microprocessor Applications Handbook Stout	£34.40
☐ Handbook of Microprocessor Design and Applications Stout	£37.60
☐ Programming the PET/CBM West	£17.80
☐ An Introduction to Microcomputer Technology Williamson	£8.20
☐ Computer Peripherals that you can build Wolfe	£12.40
☐ Microprocessors and Microcomputers for Engineering Students and Technicians Wooland	£7.10

REFERENCE BOOKS

☐ Electronic Engineers' Handbook Fink	£56.45
☐ Electronic Designers' Handbook Giacoletto	£59.55
☐ Illustrated Dictionary of Microcomputer Technology Hordeski	£8.45
☐ Handbook for Electronic Engineering Technicians Kauffman	£27.50
☐ Handbook of Electronic Calculators Kauffman	£35.00
☐ Modern Electronic Circuit Reference Manual Marcus	£44.00
☐ International Transistor Selector Towers	£10.70
☐ International Microprocessor Selector Towers	£16.00
☐ International Digital IC Selector Towers	£10.95
☐ International Op Amp Linear IC Selector Towers	£8.50
☐ Illustrated Dictionary of Electronics Turner	£12.95

VIDEO

☐ Servicing Home Video Cassette Recorders Hobbs	£12.95
☐ Complete Handbook of Videocassette Recorders Kybett	£9.25
☐ Theory and Servicing of Videocassette Recorders McGinty	£12.95
☐ Beginner's Guide to Video Matthewson	£5.35
☐ Video Recording: Theory and Practice Robinson	£14.40
☐ Video Handbook Van Wezel	£21.90
☐ Video Techniques White	£12.95

Please send me the books indicated. I enclose cheque/postal order for £..... Prices include postage and packing
I wish to pay by Access/Barclaycard. Please debit my account.

5	2	2	4													

4	9	2	9													

Signed.....

Name .....

Address .....

.....

.....

Mostek UK Ltd, Masons House, 1 Valley Drive, Kingsbury Rd, London NW9. Tel: 01-204 9322.
Deal through distributors.
Z80 family.

Motorola Ltd, c/o Graham & Gillies, Omega House, 471 Kings Rd, London SW10.
Deal through distributors – not geared up to dealing with queries direct.
6800 family.

National Semiconductor (UK) Ltd, 301 Harpur Centre, Horne Lane, Bedford MK40 1TR. Tel: Bedford 47147.
Deal through distributors.
Z80 family.

Rockwell International, Electronic Devices Division, Heathrow House, Bath Rd, Hounslow, Middlesex. Tel: 01-759 9911.
Deal through distributors.
6500 family.

SGS-ATES (UK) Ltd, Planar House, Walton St, Aylesbury, Bucks. Tel: Aylesbury 5977.
Deal through distributors.
Z80 family.

Texas Instruments Ltd, Manton Lane, Bedford MK41 7PA. Tel: Bedford 211655.
Will accept technical enquiries and deal directly with small purchasers as well as through distributors.
TMS 9900 family.

Zilog (UK) Ltd, Zilog House, 45-53 Moorgridge Rd, Maidenhead SL6 8PL. Tel: Maidenhead 39200.
Deal through distributors.
Z80 family.

Other Manufacturers:

Advanced Micro Devices, Ebury Gate, 23 Lower Belgrave St, London SW1. Tel: 01-730 0855.

Ferranti Ltd, Fields New Rd, Oldham, Lancs. Tel: 061 624 0515.

General Instrument Microelectronics Ltd, Times House, Ruislip, Middx. Tel: Ruislip 35700.

Intersil Data (UK) Ltd, 9th Floor, Basing View, Basingstoke, Hants. Tel: Basingstoke 57361.

RCA, Windmill Rd, Lincoln Way, Sunbury-On-Thames. Tel: Sunbury-on-Thames 85511.

Siliconix Ltd, 30a High St, Thatcham, Newbury, Berks. Tel: Newbury 64846.

Thompson-CSF, Ringway House, Bell Rd, Daneshill, Basingstoke, Berks. Tel: Basingstoke 29155.

BIBLIOGRAPHY

This is a list of books, mostly inexpensive, that can be investigated if you want to learn more about handling microprocessors. They are nearly all paperbacks; you can check prices by writing for the publisher's catalogue, or by asking a bookshop to check in **Books In Print.**

Non-fiction books especially can often be ordered for borrowing through a local library.

We have included a few massive reference tomes which can sometimes be found in libraries, professional electronics workshops, etc. but which cost too much for the amateur to invest in.

We have also included the addresses of several publishers who have a fair selection of electronics and computing books on their catalogues.

The book titles are followed by author, publisher and date of publication in that order, with sometimes a distributor included in the case of American titles.

4- and 8-bit Microprocessor Handbook, Adam Osborne and Gerry Kane, Osborne/McGraw-Hill, 1981.

16-bit Microprocessor Handbook, Adam Osborne and Gerry Kane, Osborne/McGraw-Hill, 1981.

A Microprocessor Primer, R. A. Penfold, Barnard Babani, Ltd., 1980.

A Z80 Workshop Manual, E. A. Parr, Bernard Babani Ltd., 1983.

Beginner's Guide To Digital Techniques, G. T. Rubaroe, Bernard Babani Ltd., 1979.

Beginner's Guide To Microprocessors, E. A. Parr, Newnes Technical Books, 1982.

Beginner's Guide To Microprocessors And Computing, E. F. Scott, Bernard Babani, 1980.

Digital Computer Logic And Electronics, Chris Gane and Alan Unwin, Cambridge Learning Ltd.

Digital Design, compiled and published by Cambridge Learning Ltd.

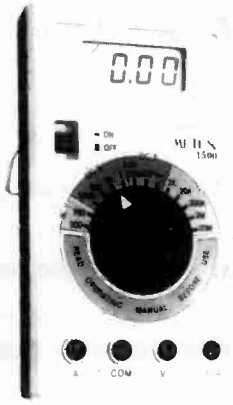
Digital Logic Design, B. Holdsworth, Butterworths, 1982.

Continued/

hi! performance hi! competitive hi!

hi! reliability hi! service hi! performance hi! competitive

Probably the
BEST VALUE FOR MONEY
DMM AT £29.50
inc P+D. VAT extra



with FREE
* TEST LEADS
* BATTERY
* OPERATING MANUAL
* SPARE FUSE

PLUS a FREE
* CARRYING CASE

and
* FULLY GUARANTEED FOR 12 MONTHS

hi! ONE large 'easy to use' rotary switch

hi! 30 protected ranges to 1000V 10 Amps and 20 Meg Ohms

hi! Other features inc. small compact hand held size – basic 0.5% accuracy – high reliability using CMOS circuitry – large clear LCD display – auto zero, polarity, overange and LO Batt. indication – diode test – tilt stand with non skid feet – high surge voltage protection etc. etc.

AVAILABLE from **HOUSE OF INSTRUMENTS** and **LOCAL DISTRIBUTORS**
Just ask for **METEX 3000 FREE DATA**

Easy to Order: Fill in and post the coupon enclosing cheque P.O. Credit Card details or Official Order

VISA ACCESS AMERICAN EXPRESS

In a hurry? Then ring (0799) 24922

Name .....
Address .....
.....
Tel. No. .....
Please send:
.... Metex Type 3000 at £33.93 inc VAT - P+D .....
.... Luxury Test Leads at £6.60 inc VAT - P+D .....
.... Luxury Test Lead Set at £10.60 inc VAT - P+D .....
Total £

Please tick and fill in number of card
Barclaycard Access American Express
Credit Card No. .....
Please allow 14 days for delivery

House of Instruments
Clifton Chambers, 62 High Street
Saffron Walden, Essex CB10 1EE
Telephone (0799) 24922 Telex 818750

hi! House of Instruments Ltd.

hi! reliability hi! service hi! performance hi! competitive hi!

RECRUITING?
SELLING A PRODUCT?
OR A SERVICE?

GET MAXIMUM BENEFIT FOR YOUR MONEY

FIND OUT ABOUT OUR WHOLE RANGE OF PUBLICATIONS IN:

COMPUTING
VIDEO
RADIO
ELECTRONICS

Simply telephone

ASP CLASSIFIED
01-437 1002
(We take Access and Barclaycard)

It's easy to complain about advertisements.

The Advertising Standards Authority.
If an advertisement is wrong, we're here to put it right.
A.S.A. Ltd., Brook House, Torrington Place, London WC1E 7HN.

BIBLIOGRAPHY

Don Lancaster's Micro Cookbook: Volume 1: Fundamentals. Don Lancaster, published by Howard W. Sams & Co. Inc., distributed by Prentice Hall International in the UK.

Elements Of Electronics Book 4: Microprocessing Systems And Circuits, F. A. Wilson, Bernard Babani Ltd., 1980.

Interfacing To Microprocessors, J. C. Cluley, Macmillan, 1983.

Interfacing To Microprocessors And Microcomputers, Owen Bishop, Newnes Technical Books, 1982.

Introducing Microprocessors, Ian R. Sinclair, Keith Dickson Publishing Ltd., 1981.

Introduction to 6800/6802 Microprocessor Systems: Hardware, Software, And Experimentation, R. J. Simpson and T. J. Terrell, Newnes Technical Books, 1982.

Microelectric Systems Level 1 and Level 2, Ian Sinclair, Holt-Saunders Ltd., 1 St. Anne's Rd., Eastbourne, East Sussex BN21 3UN. Tel: Eastbourne 638221.

Microprocessor Data Book, S. A. Money, Granada, 1982 (hardback).

Microprocessors For Hobbyists, Ray Coles, Newnes Technical Books, 1979.

Microprocessors: Your Questions Answered, Alec Wood, Newnes Technical Books. 1981.

Practical Computer Experiments, E. A. Parr, Bernard Babani Ltd., 1980.

Practical Design of Digital Circuits, Ian Kampel, Newnes Technical Books, 1983.

Practical Microprocessor Systems, Ian R. Sinclair, Newnes Technical Books, 1981.

Understanding Microprocessors, Don L. Cannon and Gerald Luecke, Texas Instruments Learning Centre Understanding Series.

Z80 Computer Systems, J. Hopton, Sigma Technical Press, 1979.

Books to look for in the library:

An Introduction to Microprocessors: Vol. 0: **The Beginner's Book;** Vol. 2: **Basic Concepts;** Vol. 3: **Some Real Microprocessors;** Vol. 4: **Some Real Support Devices;** by Adam Osborne (Vol 0 by A. Osborne and David Bunnell), published by Osborne Associates Inc., distributed by the McGraw-Hill Book Co. (UK) Ltd., McGraw-Hill House, Shoppenhangers Rd., Maidenhead, Berks. Tel: Maidenhead 23431. Updated regularly.

I C Master (two volumes covering all IC types). Edited by Dave Howell, published by Hearst Business Communications Inc., UTP Division, 645 Stewart Av., Garden City, NY11530, USA, 1982. Updated regularly. Distributed in the UK by Evan Steadman Services Ltd., The Hub, Emson Close, Saffron Walden, Essex. Tel: 0799 26699.

This reference work will take you from the stage where you know what you would like to do, to finding out which devices will do it, and through to who makes them and where to obtain them. It costs £65.00 – too much for the individual but very useful for companies and institutions.

Microprocessor Data Book (Edition 7) by D.A.T.A. Inc., distributed in the UK by London Information (Rowse Muir) Ltd., Index House, Ascot, Berks SL5 7EU. Tel: Ascot 23377. Updated regularly.

Some Publishers' addresses:

Bernard Babani (publishing) Ltd., The Grampians, Shepherd's Bush Rd., London W6 7NF.

Cambridge Learning Ltd., FREEPOST, Unit 93, Rivermill Site, St. Ives, Cambs PE17 4BR.

Keith Dickson Publishing Ltd., 17 Hendon Lane, London N3 1RT.

Newnes Technical Books, Butterworth & Co., (Publishers) Ltd., Borough Green, Sevenoaks, Kent TN15 8PH.

Texas Instruments Ltd., Book Department, PO Box 50, Market Harborough, Leics.

Prentice-Hall International, 66 Wood Lane End, Hemel Hempstead, Herts HP2 4RG.

Sybex Inc.: European catalogue from Sybex, 18 Rue Planchet, 75020 Paris, France.

The editors wish to thank the following for their assistance in compiling this edition of Electronics Digest: AMI Microsystems Ltd., Fairchild Semiconductors Ltd., Graham & Gillies (for Motorola Ltd.), Hawke Electronics Ltd. (for Motorola Ltd.), Hitachi Electronic Components (UK) Ltd., the Intel Corporation (UK) Ltd., Mitsubishi Electric (UK) Ltd., National Semiconductor (UK) Ltd., Patria (for Mostek UK Ltd.), Pelco Electronics Ltd. (for Rockwell products), Rockwell International Electronic Components, SGS-ATES (UK) Ltd., Texas Instruments Ltd., Zilog (UK), Jenny Smith (Monode Ltd.) and the indispensible Phil Walker of ETI.

Bookshelf

THE HE BOOKSHELF is the easiest way to build up your library of electronics books. Order today to have these top titles delivered DIRECT TO YOUR DOOR.

52 PROJECTS USING IC741 £1.25
More than 50 ways of using the ubiquitous IC741.

50 (FET) FIELD EFFECT TRANSISTOR PROJECTS £1.75
Contains a wide range of projects, all based on Field Effect Transistors.

COUNTER DRIVER AND NUMERAL DISPLAY PROJECTS £1.75
Features many varied projects using numeral displays, counter and driver ICs.

RADIO CIRCUITS USING IC's £1.35
All you need to know about using IC's for AM or FM reception.

IC LM3900 PROJECTS £1.35
Provides the ground work for simple and more advanced uses of IC LM3900.

PRACTICAL CONSTRUCTION OF PRE-AMPS, TONE CONTROLS, FILTERS AND ATTENUATORS £1.45
Contains practical designs for use with home constructed equipment or commercial modules.

BEGINNERS GUIDE TO MICROPROCESSORS AND COMPUTING £1.75
Basic introduction to binary arithmetic, micro-processor operation and macine language programming.

IC 555 PROJECTS £1.95
The 555 timer users 'Bible'!

CHOOSING AND USING YOUR HI-FI £1.65
Concise information for all hi-fi enthusiasts.

ELECTRONIC GAMES £1.75
Amusing electronic games which can easily and inexpensively be constructed by the enthusiast.

ELECTRONIC HOUSEHOLD PROJECTS £1.75
Packed with circuits for projects that can be used in or around the home.

A MICROPROCESSOR PRIMER £1.75
A painless approach to understanding computers and computing.

REMOTE CONTROL PROJECTS £1.95
Practical information about many applications of remote control.

ELECTRONIC MUSIC PROJECTS £1.75
Many practical circuits for less complex items of electronic music equipment.

ELECTRONIC TEST EQUIPMENT CONSTRUCTION £1.75
How to build and use a wide range of test equipment for the hobbyist.

POWER SUPPLY PROJECTS £1.75
Packed with circuits for building various kinds of power supply.

SINGLE IC PROJECTS £1.50
Easy to construct projects based around a single IC.

PRACTICAL COMPUTER EXPERIMENTS £1.75
Build and examine a number of basic computer circuit elements.

ELECTRONIC PROJECTS USING SOLAR CELLS £1.95
A range of practical applications using silicon solar cells.

VMOS PROJECTS £1.95
Contains a number of varied and interesting projects using VMOS power FETs.

DIGITAL IC PROJECTS £1.95
Simple and advanced projects based on digital ICs.

INTERNATIONAL TRANSISTOR EQUIVALENTS GUIDE £2.95
Find possible equivalents for a popular user-orientated selection of modern transistors.

HOW TO USE OP-AMPS £2.25
All you need to know about Op-Amps, a source book of circuits and design calculations.

AUDIO PROJECTS £1.95
Clear information on how to build pre-amps, mixers, power amps, tone controls etc.

To receive your books you have only to fill in the form below and send it, together with your payment, to the address stated.

Please allow up to 21 days for delivery. This offer applies only within the U.K. Prices may be subject to change without notice

SEND TO: HE BOOKSHELF, 513 London Road, Thornton Heath, Surrey CR4 6AR.

I am enclosing my Cheque/Postal Order/ for: (delete as necessary)

£..... (Made payable to A.S.P. Ltd)

OR Debit my Access/Barclaycard (delete as necessary)

VISA

Please use BLOCK CAPITALS and include post codes.

Name (Mr/Mrs/Miss) .....
(delete accordingly)

Address .....

.....

.....

Signature ..... **Date** .....

Books Required	Price
Add 75p p&p	0.75
Total Payment £	

MPU GLOSSARY

ACCUMULATOR: The register where arithmetic or logic results are held. Most MPU instructions manipulate or test the accumulator contents.

ACCESS TIME: Time take for specific byte of storage to become available to processor.

ACIA: Asynchronous Communication Interface Adapter. Interface between asynchronous peripheral and an MPU.

ALU: Arithmetic and Logic Unit. The part of the MPU where arithmetic and logic functions are performed.

ASCII: American Standard Code for Information Interchange. Binary code to represent alphanumeric, special and control characters.

ASSEMBLER: Software which converts assembly language statements into machine code and checks for non valid statements or incomplete definitions.

ASSEMBLY LANG: Means of representing programme statements in mnemonics and conveniently handling memory addressing by use of symbolic terms.

ASYNCHRONOUS: Operations that initiate a new operation immediately upon completion of current one — not timed by system clock.

BASIC: Beginner's All Purpose Symbolic Instruction Code: An easy to learn, widely used high level language.

BAUD: Measure of speed of transmission line. Number of times a line changes state per second. Equal to bits per second if each line state represents logic 0 or 1.

BAUDOT CODE: 5-bit code used to encode alphanumeric data.

BCD: Binary Coded Decimal. Means of representing decimal numbers where each figure is replaced by a binary equivalent.

BENCHMARK: A common task for the implementation of which programmes can be written for different MPUs in order to determine the efficiency of the different MPUs in the particular application.

BINARY: The two base number system. The digits are 0 or 1. They are used inside a computer to represent the two states of an electric circuit.

BIT: A single binary digit.

BREAKPOINT: Program address at which execution will be halted to allow debugging or data entry.

BUFFER: Circuit to provide isolation between sensitive parts of a system and the rest of that system.

BUG: A program error that causes the program to malfunction.

BUS: The interconnections in a system that carry parallel binary data. Several bus users are connected to the bus, but generally only one "sender" and one "receiver" are active at any one instant.

BYTE: A group of bits — the most common byte size is eight bits.

CLOCK: The basic timing for a MPU chip.

COMPILER: Software which converts high level language statements into either assembly language statements, or into machine code.

CPU: Central processor unit. The part of a system which performs calculation and data manipulation functions.

CRT: Cathode Ray Tube. Often taken to mean complete output device.

CUTS: Computer Users Tape System. Definition of system for storing data on cassette tape as series of tones to represent binary 1's and 0's.

DEBUG: The process of checking and correcting any program errors either in writing or in actual function.

DIRECT ADDRESSING: An addressing mode where the address of the operand is contained in the instruction.

DMA: Direct Memory Access.

DUPLEX: Transfer of data in two directions simultaneously.

ENVIRONMENT: The conditions of all registers, flags, etc., at any instant in program.

EPROM: Electrically Programmable Read Only Memory. Memory that may be erased (usually by ultra violet light) and reprogrammed electrically.

EXECUTE: To perform a sequence of program steps.

EXECUTION TIME: The time taken to perform an instruction in terms of clock cycles.

FIRMWARE: Instructions or data permanently stored in ROM.

FLAG: A flip flop that may be set or reset under software control.

FLIP-FLOP: Two state device that changes state when clocked.

FLOPPY (DISK): Mass storage which makes use of flexible disks made of a material similar to magnetic tape.

FLOW CHART: A diagram representing the logic of a computer program.

GLITCH: Noise pulse.

HALF DUPLEX: Data transfer in two directions but only one way at a time.

HANDSHAKE: System of data transfer between CPU and peripheral whereby CPU "asks" peripheral if it will accept data and only transfers data if "answer" is yes.

HARD COPY: System output that is printed on paper.

HARDWARE: All the electronic and mechanical components making up a system.

HARD WIRE: Circuits that are comprised of logic gates wired together, the wiring pattern determining the overall logic operation.

HEXADECIMAL: The base 16 number system. Character set is decimal 0 to 9 and letters A to F.

HIGH LEVEL LANGUAGE: Computer language that is easy to use, but which requires compiling into machine code before it can be used by an MPU.

HIGHWAY: As BUS.

IMMEDIATE ADDRESSING: Addressing mode which uses part of the instruction itself as the operand data.

INDEXED ADDRESSING: A form of indirect addressing which uses an Index Register to hold the address of the operand.

INDIRECT ADDRESSING: Addressing mode where the address of the location where the address of the operand may be found is contained in the instruction.

INITIALISE: Set up all registers, flag, etc., to defined conditions.

INSTRUCTION: Bit pattern which must be supplied to an MPU to cause it to perform a particular function.

INSTRUCTION REGISTER: MPU register which is used to hold instructions fetched from memory.

INSTRUCTION SET: The repertoire of instructions that a given MPU can perform.

INTERFACE: Circuit which connects different parts of system together and performs any processing of signals in order to make transfer possible (ie, serial — parallel conversion).

INTERPRETER: An interpreter is a software routine which accepts and executes a high level language program, but unlike a compiler does not produce intermediate machine code listing but converts each instruction as received.

INTERRUPT: A signal to the MPU which will cause it to change from its present task to another.

I/O: Input/Output.

K: Abbreviation for $2^{10} = 1024$.

KANSAS CITY (Format): Definition of a CUTS based cassette interface system.

LANGUAGE: A systemmatic means of communicating with an MPU.

LATCH: Retains previous input state until overwritten.

LIFO: Last In First Out. Used to describe data stack.

LOOPING: Program technique where one section of program (the loop) is performed many times over.

MACHINE LANG: The lowest level of program. The only language an MPU can understand without interpreter.

MASK: Bit pattern used in conjunction with a logic operation to select a particular bit or bits from machine word.

MEMORY: The part of a system which stores data (working data or instruction object code).

MEMORY MAP: Chart showing the memory allocation of a system.

MEMORY MAPPED I/O: A technique of implementing I/O facilities by addressing I/O ports as if they were memory locations.

MICRO CYCLE: Single program step in an MPUs Micro program. The smallest level of machine program step.

MICRO PROCESSOR: A CPU implemented by use of large scale integrated circuits. Frequently implemented on a single chip.

MICRO PROGRAM: Program inside MPU which controls the MPU chip during its basic fetch/execute sequence.

MNEMONIC: A word or phrase which stands for another (longer) phrase and is easier to remember.

MODEM: Modulator/demodulator used to send and receive serial data over an audio link.

NON VOLATILE: Memory which will retain data content after power supply is removed, e.g. ROM.

OBJECT CODE: Bit patterns that are presented to the MPU as instructions and data.

O/C: Open Collector. Means of tieing together O/P's from different devices on the same bus.

OCTAL: Base 8 number system. Character set is decimal 0-7.

OP CODE: Operation Code. A bit pattern which specifies a machine operation in the CPU.

OPERAND: Data used by machine operations.

PARALLEL: Transfer of two or more bits at the same time.

PARITY: Check bit added to data, can be odd or even parity. In odd parity sum of data 1's + parity bit is odd.

PERIPHERAL: Equipment for inputing to or outputting from the system (e.g., teletype, VDU, etc.).

PIA: Peripheral Interface Adapter.

POP: Operation of removing data word from LIFO stack.

PORT: A terminal which the MPU uses to communicate with the outside world.

PROGRAMS: Set of MPU instructions which instruct the MPU to carry out a particular task.

PROGRAM COUNTER: Register which holds the address of next instruction (or data word) of the program being executed.

PROM: Programmable read only memory. Proms are special form of ROM, which can be individually programmed by user.

PUSH: Operation of putting data to LIFO stack.

RAM: Random Acce Memory. Read write memory. Data may be written to or read from any location in this type of memory.

REGISTER: General purpose MPU storage location that will hold one MPU word.

RELATIVE ADDRESSING: Mode of addressing whereby address of operand is formed by combining current program count with a displacement value which is part of the instruction.

ROM: Read Only Memory. Memory device which has its data content established as part of manufacture and cannot be changed.

SCRATCH PAD: Memory that has short access time and is used by system for short term data storage.

SERIAL: Transfer of data one bit at a time.

SIMPLEX: Data transmission in one direction only.

SOFTWARE: Programs stored on any media.

SOURCE CODE: The list of statements that make up a program.

STACK: A last in first out store made up of registers or memory locations used for stack.

STATUS REGISTER: Register that is used to store the condition of the accumulator after an instruction has been performed (e.g., Acc = 0).

SUBROUTINE: A sequence of instructions which perform an often required function, which can be called from any point in the main program.

SYNTAX: The grammar of a programming language.

TRAP (Vector): Pre-defined location in memory which the processor will read as a result of particular condition or operation.

TRI STATE: Description of logic devices whose outputs may be disabled by placing them in a high impedance state.

TTY: Teletype.

TWO'S COMPLEMENT ARITHMETIC: System of performing signed arithmetic with binary numbers.

UART: Universal Asynchronous Receiver Transmitter.

VDU: Visual Display Unit.

VECTOR: Memory address, provided to the processor to direct it to a new area in memory.

VOLATILE: Memory devices that will lose data content if power supply removed (i.e., RAM).

WORD: Parallel collection of binary digits much as byte.

If you missed out on Breadboard '79, '80, '81 and '82 . . . . Now's the time to catch up with

Friday November 25th	10am - 6pm
Saturday November 26th	10am - 6pm
Sunday November 27th	10am - 4pm

Improved venue

We have transferred Breadboard to Cunard International Exhibition Centre, so that we can offer improved facilities to the visitor, including car parking and ease of access by rail, tube and car, all in a modern attractive setting. We have also arranged a reduced hotel/rail fare package to attract enthusiasts from all parts of the country.

Planned features include

1. Full range of lectures planned over 3 days to cover most aspects of electronics and computing.
2. Electronics/Computing Advice Centre — manned by experts.
3. Demonstration of electronic organs and synthesisers.
4. Holography presentation.
5. Practical Demonstration on "How to produce printed circuit boards".
6. Computer Corner — extensive display of computer hardware — "Try Before You Buy".
7. Amateur radio Action Centre.
8. Computer controlled model railway competition.
9. Pick of the Projects — Demonstration of the best from ELECTRONICS TODAY INTERNATIONAL, HOBBY ELECTRONICS and ELECTRONICS DIGEST over the past ten years.
10. Giant T.V. screen video games.
11. Robotic display.

Why not bring the family to the show and enjoy a weekend in London? We have arranged a complete hotel package for our visitors to the exhibition. All inclusive rail tickets also available. Send now for details of what *we*, the organisers, can offer *you*.

HOLIDAY WEEKEND

Write to: Breadboard '83
ASP Exhibitions
145 Charing Cross Road
London WC2H 0EE