Study of Various Motherboards

Thesis

Submitted to the Saurashtra University, Rajkot
For
The Degree of Doctor of Philosophy
(Science)
(Electronics)

By
Bimal H. Vyas
Hardware Support Specialist
Alpha and VAX Server Division
Hewlett Packard
Canada

Research Supervisor
Dr. H. N. Pandya
Associate Professor,
Department of Electronics
Saurashtra University,
Rajkot-360005
India

January 2007

Affectionately dedicated to
My parents
Dr. H. F. Vyas
Dr. U. H. Vyas

Statement under O. Ph. D. of Saurashtra University

The content of this thesis is my own work carried out under the supervision of Dr. H. N. Pandya and leads to some contribution in Electronics supported by necessary references.

(B. H. Vyas)
Hardware Support Specialist
Alpha and VAX Server Division
Hewlett Packard
Canada

This is to certify that the present work submitted by Mr. Bimal H. Vyas for the Ph.D. degree of Saurashtra University, Rajkot, Gujarat, India has been the result of about 5 years of work under my supervision and is a valuable contribution in the field of Electronics.

Dr. H. N. Pandya,
Associate Professor,
Department of Electronics,
Saurashtra University,
Rajkot- 360 005

Acknowledgement

I sincerely thank to my research advisor, Dr. Haresh N. Pandya , Head, Department of Electronics, Saurashtra University, Rajkot. He is not only a great scientist with deep vision but also and most importantly a kind person. His trust and scientific excitement inspired me in the most important moments of making right decisions and I am glad to work with him. His guidance and mentorship make my work easy and valuable. Also I got moral support from his wife Mrs. Kiran Pandya and lots of love from his little daughter Miss Shweta Pandya through out my research.

I want to thank all great people of the Department of Electronics. Without their co-operation and kindness this work would not have been successfully completed.

Special thanks go to Mr. Maulin Nanavati who is a great helpful and trustful friend. My words will be short to say thanks to him for being the best co-operative friend. He was always with me in good and bad time during the study.

I am very much thankful to my uncle Arvind Vyas, Aunty Jaishree Vyas, cousins Rushi and Ravi Vyas for moral support through out my study.

My colleagues Mr. Andnd Bhaskar, Dr. Darshan Vyas, and Mr. Harshad Chirutkar made my work running smoothly. My senior, Dr. Kapil Bhatt, from the Department of Electronics, Saurashtra University, Rajkot supported me in my research work. I want to thank him for all their help, support, interest and valuable hints. Also would like to thanks Mr. Rajendrakumar Khar, Mr. Jaipalsinh Zala, Mr. Hemanshu Vachaani and all other friends.

There was a lovely friend Neha Vyas, who provided an additional and joyful dimension to the life mission.

After my migration to Canada, my research is also blessed by Paul Brown, Claudia Caredona Tellisa and Pierre Severney.

I feel a deep sense of gratitude for my father Dr. H. F Vyas and my mother Dr. Usha H. Vyas, who provided me an opportunity to gain a higher level education. I am grateful for my sisters, brother and brother-in-law. Love of my cute nephew and niece was always with me.

CHAPTER 1

INTRODUCTION

1.1 Objective of present work

At present the society of the whole world is enjoying the fruits of scientific inventions and technological advancement. The very fast growing technology is the computer technology. As in the present age the personal computers and their applications are entering in the routine life, it is quite interesting and inevitable to know and learn about personal computers.

Basically, the essential part of most personal computers is the motherboard. In many PCs the motherboard may be the enter system. The circuitry located on motherboard defines the computer, its capabilities, limitations, and personality. The Motherboard also offers us an excellent introduction to the technologies that underline modern computers. So, we decided to study of various motherboard and basic technology used in earlier motherboards.

Motherboard is the largest circuit board inside the personal computer. A typical motherboard provides attachment points for microprocessors (CPU), memory (RAM & ROM) and various supporting chips for the CPU. To easily extend circuit of the motherboard, it contains many expansion slots on which we can connect smaller circuit boards like, graphic cord, sound card and other application specific cards.

The first product to bear the PC designation was the IBM personal computer introduced in August 1981. The first motherboards to be used in PCs were those of IBM's original personal computer. So we decided to study the earlier motherboards, which gives basic knowledge of different logic blocks. It provided 8 bit XT slots, which always ran at same speed as in XT board CPU support is Intel 8088, Memory support is 640 kilobytes in the form of 16 pin DIP sockets

Motherboards varied a lot more from one model to another in those grand old days. The user made heavy use of PC-XT motherboards as personal computer. Of course, the heavy demand for efficient computers gave rise to still advanced microprocessors like 80816, 80286, 80386... up to at present Pentium IV etc. simultaneously supporting motherboards also were designed for CPU.

But to learn basics of motherboard in depth we have decided to use in house facility efficiently and then designed single board based computer. No doubt, with the discrete support circuitry, designing a motherboard was a true engineering challenge becomes it required.

When advanced microprocessor were designed, it required more and more supporting chips, which make motherboard more costly and complicated. As semiconductor firms become more experienced and fabrication technology permitted smaller design rules and denser packaging, however all the diverse support functions inside a motherboard were integrated into few VLSI components individually termed Application specific integrated circuits (ASICs) collectively called a chipset. This integrated approach made the motherboard less expensive. This chipset enhances the performance of advance microprocessor deep understanding of electronic function.

To study various motherboards I took one motherboard from each stage of evolution. Major new features of 286 is ISA Bus, coprocessor interrupt, bi-directional address bus, intelligent keyboard interface, software reset shadow Ram for system BIOS and Video Bios etc.. In 386 features introduced are EISA bus, cache memory, SIMM RAM, MIPS counter etc.. When 486 systems came, a local bus called VESA Bus was introduced to support high-speed transfer to display adaptor. Other than bus it introduced

software speed selection, programmable DRAM wait states, PS/2 mouse support and power management. Soon Intel introduced PCI bus, which becomes standard for Pentium systems. Front side Bus has become the system bus due to the dual bus architecture of Pentium pro. And subsequent processors in Pentium family (Pentium IV) have drastically changed the system bus speed to 400 MHz. New version of P-IV offers 533 MHz speed.
In practice, many people in this technological field face the fault in motherboards. But it is very difficult to detect the problem and troubleshoot in above motherboards.

Trouble shooting in a motherboard is an intelligent & sophisticated process. To learn trouble shooting in depth, we have selected a trainer, which has discrete support circuitry motherboard (PC-XT motherboards). It gives deep understanding of electronic function of motherboard. To observe various signal status of some important probe sections in motherboard, it provides various points on motherboard. To debug various problems, of motherboards, it consists of various fault analysis test points. Such test points help to troubleshoot motherboard. With the help of above method we can easily get experience to work and prepare our selves to solve problems in advanced motherboards.

Now a day many companies manufacture motherboard like Intel, Mercury, Asus, Gigabyte, etc. Even if though they use the same microprocessor they offer different circuit design. Due to that reason, it gives different performance for same microprocessor (CPU). Mainly this difference is due to different Chipset used in motherboard.
Major motherboard manufacturers design and build motherboard on the basis of requirement of general new product to entice a general user. But they also can manufacture special system for particular use at the lowest possible cost. For that reason

we have decided to test various motherboards on fixed platform. To test we have decided used benchmark software and run useful other general soft wares. Fixed platform we have used is as follows.

1) CPU-Intel Pentium IV processor

2) Memory 128MB RAM

3) Hard disk - 40GB Seagate

But, as soon as we have started study it seems to be very interesting for other processors too. So we have also done this benchmark on many different systems and laptop as well.

This study of various motherboards will be helpful to the both the user. This will offer rather detail information on motherboards to the students, teachers and professionals.

1.2 Literature survey

As the subject we have been chosen is very wide. There is no particular book is exactly matches to our research so, we have started getting all information about Microprocessor from 8085 to Pentium. Then studied various applications and advance features of microprocessor. Finally collected and studied books related to circuit designing to implement our ideas into the real format.

After study of all literature we have decided to document all different essential information at one place.

Some of the useful books are listed as follows.

[1] Microprocessor architecture Programming and applications by R.S.gaonkar

[2] Microprocessor data handbook, Revised Edition, BPB Publications

[3] 0000 to 8085 Introduction to microprocessors for engineers and scientists by P.K.Ghosh and P.R.Shridhar, second edition, PHI Publications.

[4] Printed Circuit Board by Dr. H. N. Pandya, Published by Gujarat Granth Nirman Board, Ahmedabad,India.

[5] The Indispensable PC Hardware Book By Hans-Peter Messmer , Third Edition, Eddision Wesley Longman Ltd, England

[6] All about Motherboard By Manahar Lotia and Pradeep Nair, Second Revised Edition, BPB Publications, New Delhi, India

[7] An Introduction to the Intel Family of Microprocessors By James L. Antonakos, Third Edition, Pearson Education, India

[8] The Pentium Processor By James L. Antonakos, , Pearson Education, India

[9] Intel Microprocessors 8086/8088, 80186, 80286, 80386, 80486 Pentium, Pentium Pro Processor, Pentium II, Pentium III, and Pentium IV: Architecture, Programming, and Interfacing By Barry B, Brey, Sixth Edition, Pearson Education, India

[10] Microprocessor and programmed Logic By Kenneth L. Short, second edition,

[11] The 80x86 Family: Design, Programming, and Interfacing, By John Uffenbeck, third edition, , Pearson Education, India

[12] PC Quest Magazine October 2004 By Cyber Media New Delhi

List of few motherboard manufacturers is as follows.

ABIT Ltd.

Motherboard manufacturer specializing in motherboards for the over clocking and performance market.

AOpen

A division of Acer corporation that sells primarily to the OEM market. They even make a computer motherboard which has a vacuum tube on it.

ASUSTek

One of the oldest and most respected motherboard manufacturers on the market.

Chaintech

Chaintech is one of the newer entries in the US performance motherboard market.

DFI

Maker of a wide variety of ATX and microATX design PC motherboards.

ECS Elite group

Maker of a wide range of PC motherboards that has recently started selling to the performance market.

EPoX Computer

A relatively new motherboard manufacturer that has many high performance motherboard options.

FIC Inc.

Manufacturer of a number of different motherboards. Note: This site does not view well outside of Internet Explorer.

GigaByte Technology

Manufacturer of a wide variety of motherboards.

Intel

The Company most known for its CPUs also develops a line of motherboards based upon its own chipsets and CPUs.

IWILL

Manufacturer that started out developing SCSI controller cards that has expanded into the PC motherboard market.

MSI Inc.

Developer of a wide range of well prices motherboards which still contain excellent features.

Shuttle Computer Group

Manufacturer of a wide variety of motherboards including the small form factor X PC line.

Soltek USA

One of the newest motherboard manufacturers on the market that recently began produce small form factor products.

Tyan Computer

Manufacturer well known for their multiple processor motherboards.

Important web-sites for motherboard manufacturer and microprocessor manufacturers.

http://www.motherboards.org	www.intel.com
http://www.asus.com/	http://www.advantech.com
http://www.dell.com/	http://www.fica.com/
http://www.hp.com/	http://www.ibm.com/us/
http://www.intel.com/design/motherbd/	http://www.cpu-world.com/
http://www.x86.org	http://www.microprocessor.sscc.ru
http://www.apple.com	http://www.2cpu.com/

CHAPTER 2

BASICS OF MOTHERBOARD

2.1 Introduction

PC-XT Motherboard is an old motherboard by today's advance motherboard, but basic functional blocks remain same in even recent motherboards. To understand this functional design we have considered PC-XT motherboard. This chapter describes all functional blocks of PC-XT motherboard.

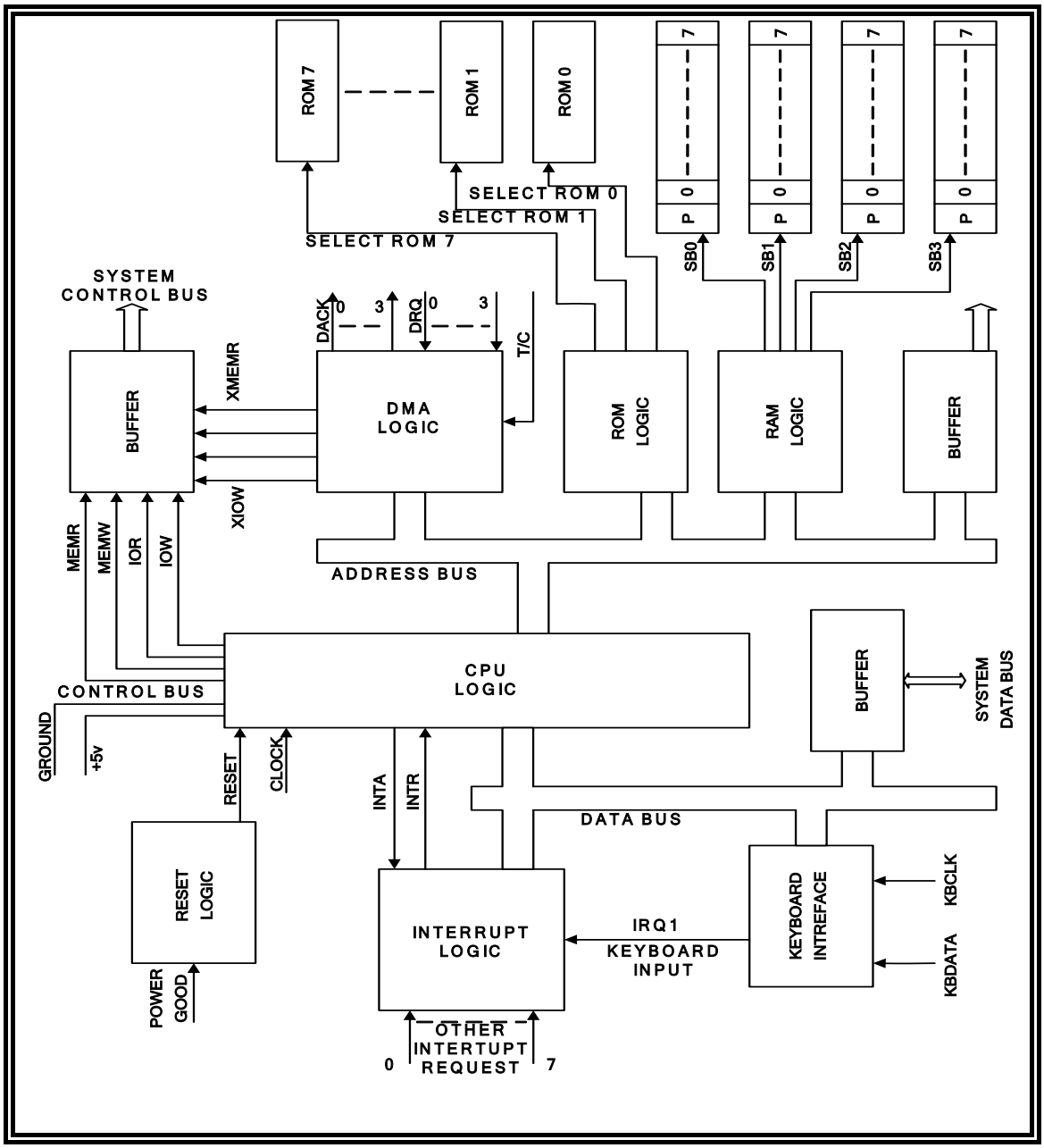


Figure 2.1 Functional block diagram of PC-XT motherboard

Figure 2.1 illustrates functional block diagram of PC-XT motherboard. We divide overall function of motherboard in the following various function.

- Reset Function
- Memory Function
- CPU Function
- Bus Function
- I/O Function
- Interrupt Function

For ease of design the full circuit of motherboard, we focus separate design of functional blocks and logical interconnection of those blocks.

2.2 Logic Blocks of motherboard

2.2.1 List of logic blocks

To learn functional logic in more details we divide basic function in various functional blocks

- Reset Functional Block

- Memory Function

  This logic divided into six internal blocks depends on various memory functions

  - RAM functional block
  - RAM decode functional block
  - RAM parity functional block
  - DRAM functional block
  - ROM functional block
  - ROM logic functional block

- CPU Functional Block

- Co-processor Functional Block

- BUS Function

    - BUS arbitration functional block
    - Control BUS functional block
    - Address BUS functional block
    - XT-BUS functional block
    - Data BUS functional block

- I/O Function

    - I/O codes decode functional block
    - Keyboard interface functional block
    - Serial Port interface functional block
    - Mode switch input functional block
    - Speaker functional block
    - Time of day functional block

- Interrupt Function

    - DMA functional block
    - Wait state functional block
    - NMI functional block
    - Interrupt control functional block

2.2.2 Details of logic blocks

Here, I would like to represent above mentioned functional blocks in a sequence.

RESET FUNCTIONAL BLOCK

I would like to begin my explanation of this motherboard with the reset logic block. The reason is, as soon as power to the system is switched on, **reset logic block** is the

first logic to be performed. As soon as the Reset signal is generated, all the expansion cards are reset to their initial position.

The reset signal is generated

- When power is switched on to the computer
- When front panel Reset button is pressed
- When Power good signal becomes absent
- When Ctrl, Alt, and Delete keys are pressed simultaneously

When Power is Switched on to the PC: -

When power to the computer is switched on, the power supply (SMPS) used in the PC generates a Power Good signal for the motherboard.

This Power Good signal is given out by the SMPS only after all the supplies, i.e. Input AC supply and different DC output supplies have stabilized and are within the tolerable limits.

Power Good signal gives the indication to the motherboard that all the supplies are now suitable for use in the system.

Once this initial Power Good signal is received, the Reset Logic Block generates a RESET signal for the CPU. The complete system, including all the daughter cards, is set to initial logic state. This type of resetting is called the Power-Up resetting.

A low going signal is given to the reset pin RES of the clock generator 1C of the motherboard which, in turn, gives resetting signal to the CPU pin no.21.

This resetting signal is also given to the channel 82 of the I/O slots.

When the microprocessor performs the resetting function, all the flags are cleared, all interrupts are disabled, and all registers are emptied. The Power Good signal is typically +5 volts.

When Reset button is pressed: -

The system can be manually reset to the initial logic state by either pressing the RESET button or by pressing the Ctrl, Alt, and Del keys simultaneously.

When the RESET button is pressed, a low pulse appears at the RES Input of the clock generator pin no. 11, which in turn, gives this information to the CPU.

This makes the complete system to reset to its initial logic state.

This type of resetting is also called Hardware or Manual Resetting.

One of the terminals of Reset switch is connected to the ground, as soon as the front panel reset switch is pressed; it directly connects pin no. 11 of the clock generator with the ground.

When the RESET switch is released the pin no. 11 once again attains high logic state, this generates the required negative pulse at the $\overline{RES}$ input of the clock generator.

When power good signal becomes absent: -

If for any reason, the input AC supply to the SMPS or the output DC supplies given to the motherboard from the SMPS falls low, or becomes too high then the Power good signal becomes low immediately.

A resetting low signal is given to the clock generator, which passes this information to the CPU, and complete system is again reset to its initial logic.

This type of resetting is also called Power-up Resetting.

The system cannot be used until all the supplies have become normal again.

When Ctrl, Alt, and Delete keys are pressed simultaneously: -

When the Ctrl, Alt and Del keys are pressed simultaneously, Power-On-Self-Test (POST) is executed and system is restarted, but the CPU and the various hardware circuits remain unaffected and are not reset.

This type of resetting is called SOFT RESETTING.

RAM FUNCTION:

I would give next priority to the RAM logic because all the data used by the CPU is first written in the RAM memory and all results are also first given to the RAM.

RAM FUNCTIONAL BLOCK

ROM logic deals with the different logic blocks of motherboard RAM memory.

RAM logic is very important part of the motherboard as all the data used by the CPU is first written in the RAM memory and all the results are also first given to the RAM.

RAM memory on the motherboard is composed of four banks, each containing nine RAM chips.

On a XT motherboard the maximum usable RAM is limited to 640K by the Microsoft DOS environment.

These four memory banks are numbered -Bank O, Bank 1, Bank 2, and Bank 3.

Of these two RAM banks are formed by using the 4164 RAM chips while the other two banks are formed by using the 41256 RAM chips.

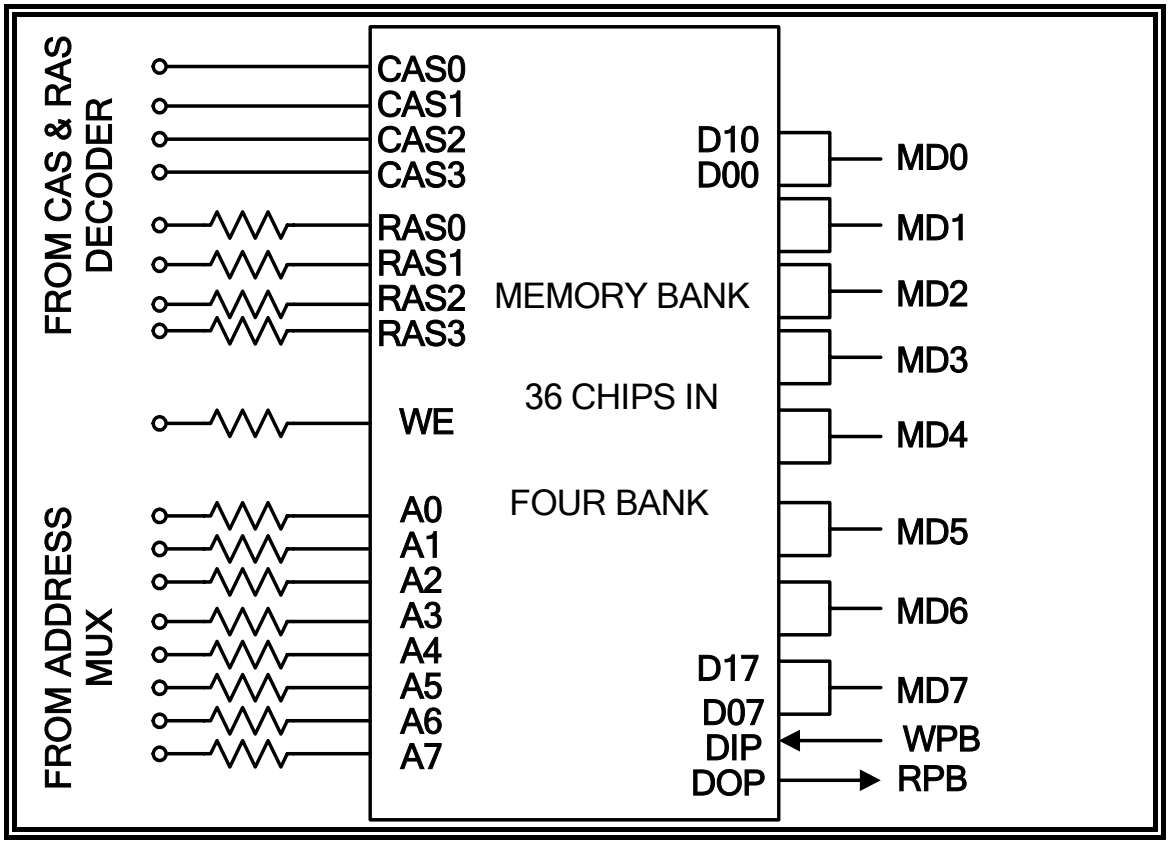


Figure 2.2 Basic RAM function logic

Figure 2.2 shows basic ram function logic. Various types of memory banks used on different versions of 8-Bit motherboards are given below:

- Four banks of 16KB DRAM each - total 64KB
- Four banks of 64KB DRAM each - total 256KB
- Two banks of 256KB DRAM each, and Two banks of 64KB DRAM each - total 640KB

Most of the present 8-Bit motherboards are using either the 256KB or 640KB RAM modules (memory banks).

Selection of the particular RAM bank is done by the RAM Decode Logic Block.

RAM DECODE FUNCTIONAL BLOCK :

Figure 2.3 Illustrates RAM decode functional block. It is the job of RAM Decode Logic Block to selects the RAM bank containing the memory location corresponding to the 20-Bit address currently on the address bus.

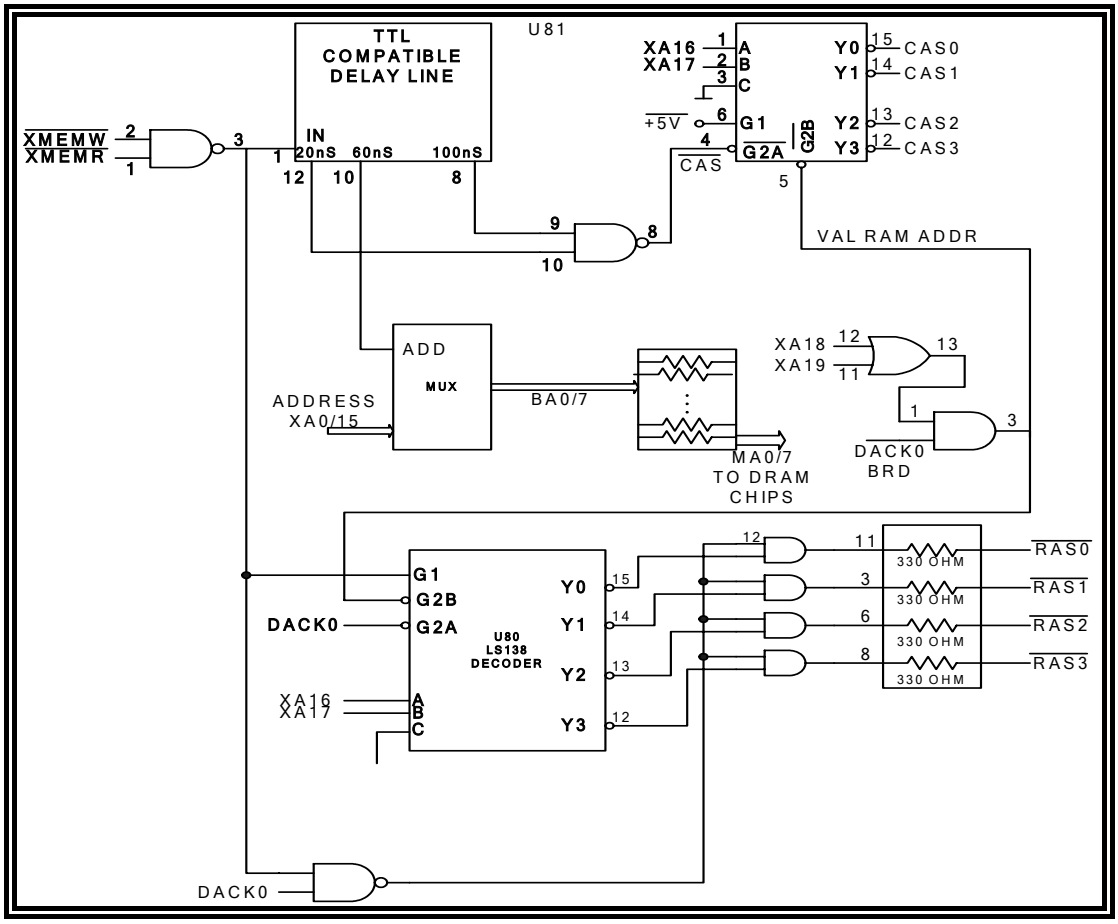


Figure 2.3 RAM decode functional block

Four RAM banks, each containing nine DRAM chips, are generally used on a PC-XT motherboard. All the memory cells in these chips are arranged as rows and columns.

The 20-Bit address to each memory location of the DRAM chips is used for the column address as well as for the row address. Lower Bits of the address represent the row address while higher address Bits represent the column address.

Internal address decoder blocks are built within these memory chips.

The $\overline{CAS}$ and $\overline{RAS}$ decoders and the address multiplexers form the RAM decode logic block.

The Row Address Bits and the Column Address Bits are given to the same input address lines of DRAM chips through an address multiplexer.

A TTL compatible delay line is also formed in the RAM Decoder Logic Block.

To select a particular RAM bank, the $\overline{MEMW}$ and $\overline{MEMR}$ signals are made low (active) by the control logic block. These are the two inputs to the NAND gate.

$\overline{RAS}$ output of the NAND gate is connected to the input of the delay line and also to one of the inputs of the $\overline{CAS}$ and $\overline{RAS}$ decoders.

Second input to the decoders i.e. $\overline{CAS}$ is given from one of the two outputs of the delay line.

When the $\overline{MEMW}$ and $\overline{MEMR}$ signals go low, the RAS input terminal of the decoders goes high.

Now $\overline{RASO}$, $\overline{RAS1}$, $\overline{RAS2}$, and $\overline{RAS3}$ outputs of the decoders, which are connected with the $\overline{RASO}$, $\overline{RAS1}$, $\overline{RAS2}$, and $\overline{RAS3}$ input terminals of the DRAMs are made active low.

During this time, the $\overline{CASO}$, $\overline{CAS1}$, $\overline{CAS2}$, and $\overline{CAS3}$ outputs of the decoders remain high. These outputs are connected with the $\overline{CASO}$, $\overline{CAS1}$, $\overline{CAS2}$, and $\overline{CAS3}$ input terminals of the DRAM chips.

Next, the $\overline{ADDR\ SEL}$ output signal of the delay line is made low. This is connected to $\overline{ADDR\ SEL}$ input terminal of the address multiplexer.

Address Bits XAO to XA15 are already present at the input address lines of this multiplexer. As the $\overline{ADD\ SEL}$ terminal of this multiplexer is low, it loads the row address Bits on the address input lines of the DRAM chips.

This row address is kept latched in the DRAM chips for as long as necessary. Now, the $\overline{RAS}$ input to the decoders is made low.

$\overline{CAS}$ input to the decoders is made high by the delay line and the $\overline{CASO}$, $\overline{CAS1}$, $\overline{CAS2}$, and $\overline{CAS3}$ input terminals to the DRAM chips go low while the $\overline{RASO}$, $\overline{RAS1}$, $\overline{RAS2}$, and $\overline{RAS3}$ terminals are made high.

The $\overline{ADD\ SEL}$ terminal of the address multiplexer is made high.

Now, the column address Bits get loaded in the DRAMs and the desired memory location is selected.

Since the RAM banks are on the MAO-MA7 line, all the address Bits are loaded on this bus.

DRAM REFRESH FUNCTIONAL BLOCK:

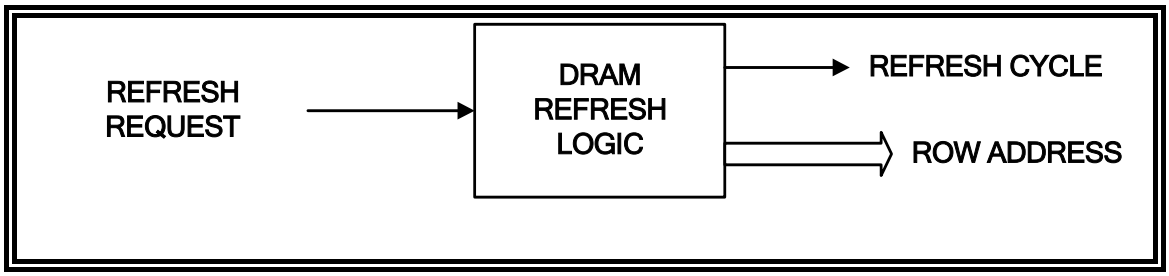


Figure 2.4 Functional block to refresh DRAM Dynamic Memory

Figure 2.4 is functional block to refresh DRAM Dynamic Memory (DRAM). Two basic types of RAM chips used with the motherboard are

- SRAM (Static RAM) and
- DRAM (Dynamic RAM).

Cell of the DRAM memory chip is formed of a single capacitive cell (capacitor). The two logic states of this cell are "charged" (logic- 1), and "discharged" (logic-0).

The capacitors lose its stored charge due to its internal leakage through the dielectric over some time. Thus, if this capacitor is not recharged periodically, the data stored in it will get lost.

This periodical refreshing of DRAM memory is done by the memory refreshing pulse. This refreshing pulse is required every 15 microseconds.

ROM FUNCTIONAL BLOCK

When the power is switched on to the computer system, the Reset Logic becomes active.

Once the system is initialized by the RESET operation, the ROM and RAM logic take over the subsequent actions.

The ROM holds all the instructions necessary for proper startup of the system and to makes it ready for use.

After the RESET operation, the CPU reads the instructions stored in the ROM chips.

The initial instructions stored in the ROM chip immediately perform the Power-On-Self-Test (POST) and initializes all motherboard logic.

Next function performed by the instructions stored in the ROM is to copy the DOS (Disk Operating System) program from the floppy disk or the hard disk into the RAM memory of the motherboard.

DOS must be loaded into the RAM every time the system is powered-on.

This operation is commonly referred to as BOOTING of the system.

The Basic Input/Output System (BIOS) program also remains stored in ROM chips. All the devices interfaced in the system are recognized, sensed, or identified by the CPU using these ROM-BIOS instructions.

Some motherboards also contain an additional ROM chips with a program called the BASIC Interpreter. An interpreter translates the instructions of a high level language into codes, which are executable by the machine.

BASIC (Beginners All Purpose Symbolic Instruction Code) is a high level language, which is not much in use nowadays. Hence, it is not essential that the motherboard contain this interpreter.

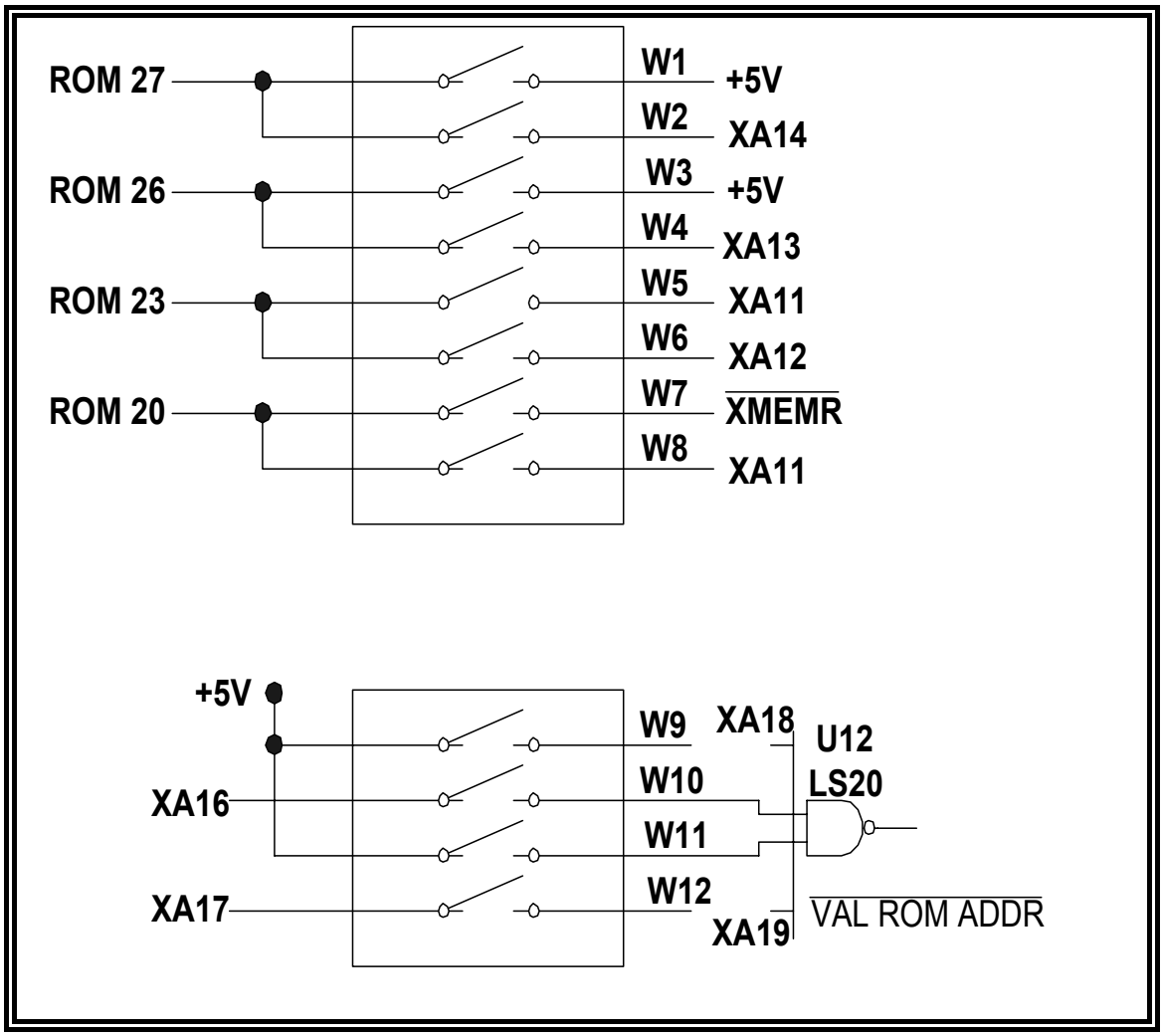


Figure 2.5 Functional block of ROM LOGIC BLOCK

Figure 2.5 shows Functional block of ROM logic block. Some of the popularly used ROM chips on the motherboard of a personal computer are - 2764, 27128, and 27256. All of these ROM chips are 28 pin chips requiring an operating supply voltage of +5 volts.

The ROM memory capacity for a PC motherboard should ideally be 64KB. But even 40KB ROM space is sufficient for executing most of the essential POST, BIOS, and the BASIC interpreter routines.

Even 8KB ROM space is sufficient to hold the POST and the BIOS routines.

Sometimes the Programmable ROM (PROM) 27128 is used on PC motherboards offering a choice of two BIOS options. A DIP switch connected on the motherboard is used for making the selection between these two versions of the BIOS.

No Refreshing Pulses are required in the ROM decode logic since ROM does not lost its content with time, ROM is a Nonvolatile Memory.

Selection of the particular ROM chip from the various chips used on the motherboard is made by the ROM Decode Logic Block.

Many sockets are provided for installing ROM chips on the PC-XT motherboards. ROM Decode Logic Block selects the required ROM chip on the basis of the ROM memory location which is to be read.

ROM DECODE FUNCTIONAL BLOCK

Many ROM chips are used on the motherboard. Data is stored in the numerous memory locations of these ROM chips.

Whenever the CPU executes a program, the memory location containing the relevant data must be accessed.

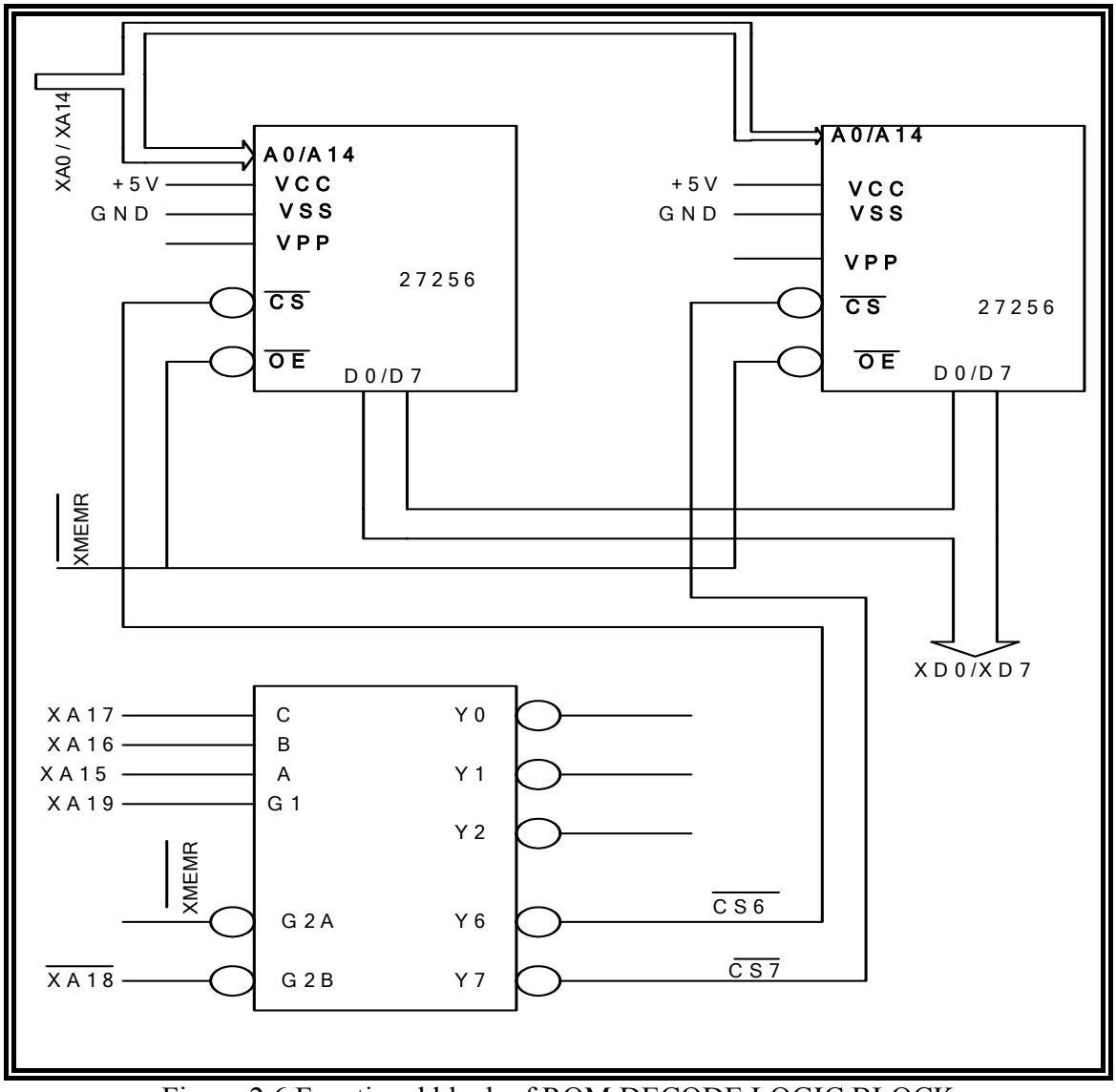


Figure 2.6 Functional block of ROM DECODE LOGIC BLOCK

Figure 2.6 shows Functional block of ROM DECODE LOGIC BLOCK. The ROM Decoder Logic Block performs the task of selecting the particular ROM chip that holds the requisite memory location.

This ROM Decode Logic Block generates ROM Chip Selection control signal.

ROM decoder 1C (1C LS138/U23), used in the ROM Decode Logic Block, is also installed either on the motherboard directly or in a socket provided for it on the motherboard. When some ROM chip is to be selected then this ROM Decode Logic checks the four Most Significant Bits (MSBs) of the 20 address Bits on the address bus and selects the particular ROM chip containing this address.

The ROM logic then generates the appropriate Chip Select signals. Thus, the ROM chip containing the required memory location gets selected.

Different types of ROM chips are used as per the motherboard configuration. Hence, correct jumper settings should be made to match the ROM type so that the ROM Decoder Logic Block generates correct ROM Chip Select signals.

Sometimes the ROM decoder chips are mounted on the motherboard in different positions marked as A, B, and C on the motherboard.

CPU FUNCTIONAL BLOCK

Figure 2.7 illustrates CPU functional block. This is the largest logic block on the motherboard. All major operations related to the CPU are carried out in this logic block.

The operations related to the various support chips on the motherboard are also included in the CPU logic block.

CO-PROCESSOR FUNCTIONAL BLOCK

This optional logic block is installed on the motherboard to increase the efficiency of the CPU. Coprocessor Chip 8087 is used in this logic block. Important operations of this block are

- Execution of Floating Point Instructions
- Assisting the CPU in execution of the instructions of the main program
- Maintaining a parallel instruction queue for the CPU

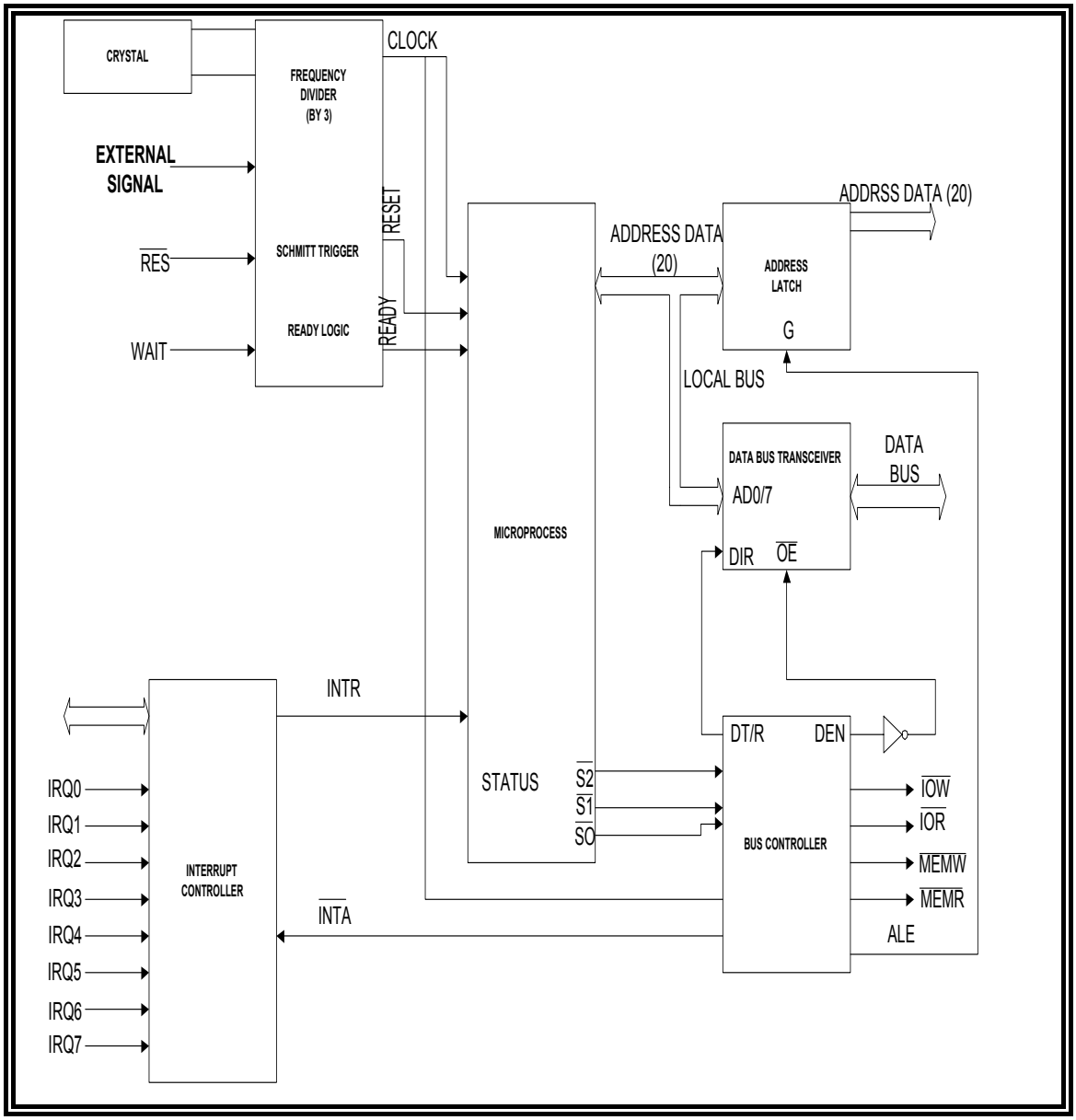


Figure 2.7 CPU functional block

DMA FUNCTIONAL BLOCK:

As shown in the figure 2.8 DMA functional blocks. The DMA (Direct Memory Access) Logic block includes the operations of the DMA controller and its supporting chips.

This block performs the function of managing data transfer between the various I/O controllers, Floppy Disc, Hard Disk, and the Memory devices.

It receives the data transfer requests from these devices and generates bus control request to the Bus Arbitration Logic (BAL) block. The BAL block is discussed later in this section.

DMA logic block includes the DMA Controller, Address Buffer, and Address Latch.

Various channels formed in the DMA controller are

- Channel O, Channel 1, Channel 2, and Channel 3.
- Channel O: "Dynamic memory refresh cycle":

  No data transfer takes place in this channel and only the address Bits and control signals are provided on this channel.

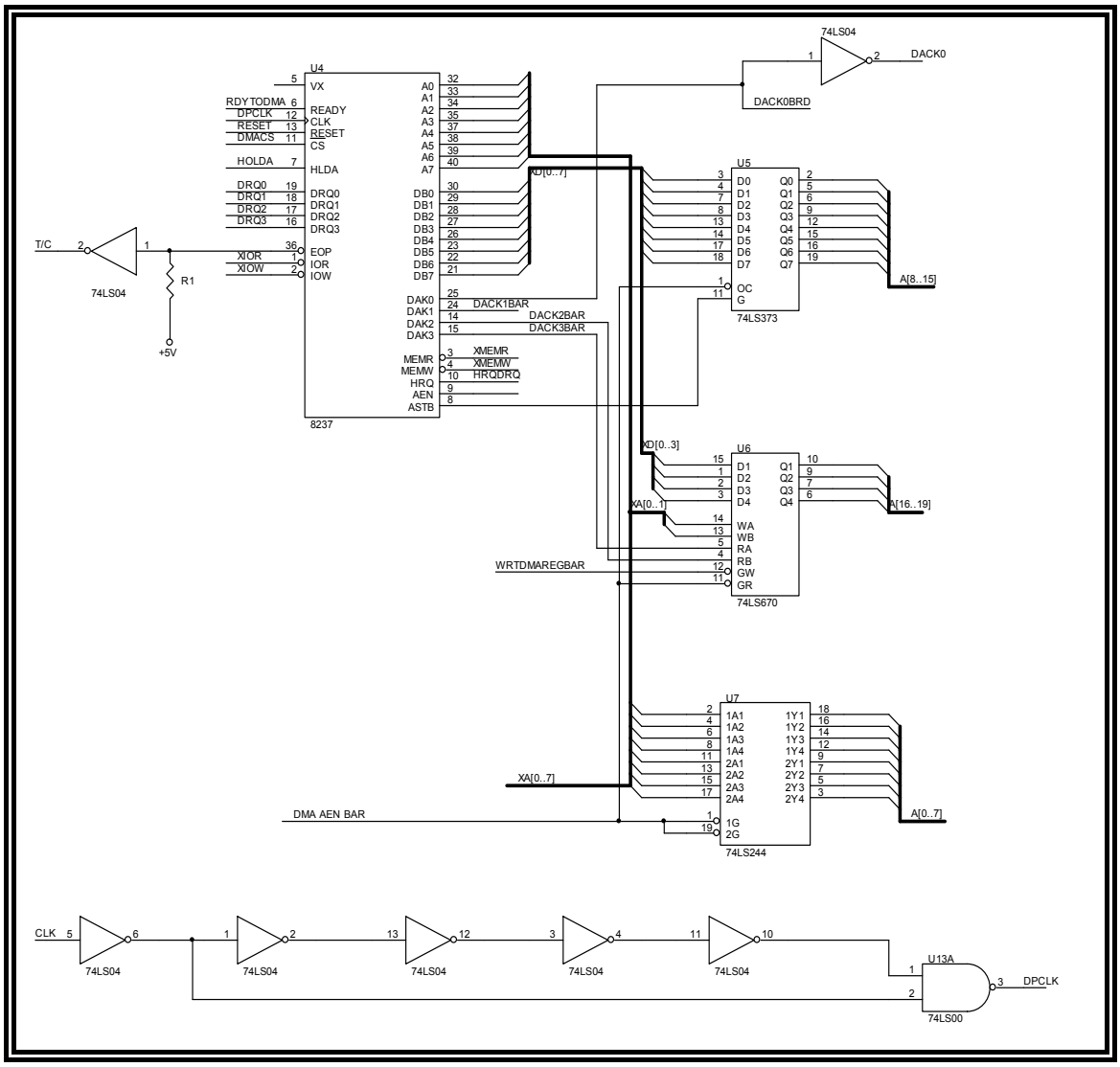


Figure 2.8 DMA functional block

- Channel 1: Not used
- Channel 2: Data is transferred between the floppy drives.
- Channel 3: Data is transferred between the hard disk drives on this channel.

DMA controller which is a 16-Bit chip, is capable of transferring 64KB of data in one I/O cycle.

DMA logic continuously checks the DREQ (DRQ) line during every clock cycle (when it is idle). In this condition, the DMA controller also checks the Chip Select signal to determine whether the CPU wants to access its internal registers or not.

When the DREQ signal is sensed active, the DMA logic block generates Hold Requests (HRQ) for the CPU.

DMA logic now manages the data transfer upon receiving an acceptance from the CPU in the form of corresponding HLDA signal.

Data is transferred by the DMA logic block in four manners:

Single transfer: Only a single byte of data is transferred and soon afterwards the bus control is returned to the CPU.

If another byte has to be transferred then a fresh HRQ must be made and DMA has to wait for the corresponding HLDA signal.

Block Transfer: The DMA returns the control of the bus to the CPU only after complete data transfer has taken place.

Demand Transfer: DMA continues to manage the data transfer for as long as the requesting I/O device maintains the DREQ request active.

As soon as the DREQ is removed, DMA transfers the bus control back to the CPU.

Cascade Transfer: In case, when more than four DMA channels are required on the motherboard, this cascade mode can be used to install more than one DMA controllers in the DMA logic.

WAIT STATE FUNCTIONAL BLOCK

Figure 2.9 illustrates the wait state function for CPU and DMA controller. Other than the CPU and the DMA controller, the memory and various interfacing peripheral devices also require use of the system bus.

This Wait State Logic block generates the different Wait State signals for the CPU and the DMA controller. When the CPU or the DMA controller receives these Wait State signals, the system bus gets transferred to some other device.

For example, when the Hard Disc controller generates the DRQ3 signal to the DMA controller then the Wait State Logic (WSL) block senses this signal and sends a Wait signal to the CPU.

Now, the control of the system bus is transferred to the requesting device i.e. to the Hard Disk Controller.

On the other hand, if the CPU has to use the system bus then the WSL block gives the Wait signal to the DMA controller.

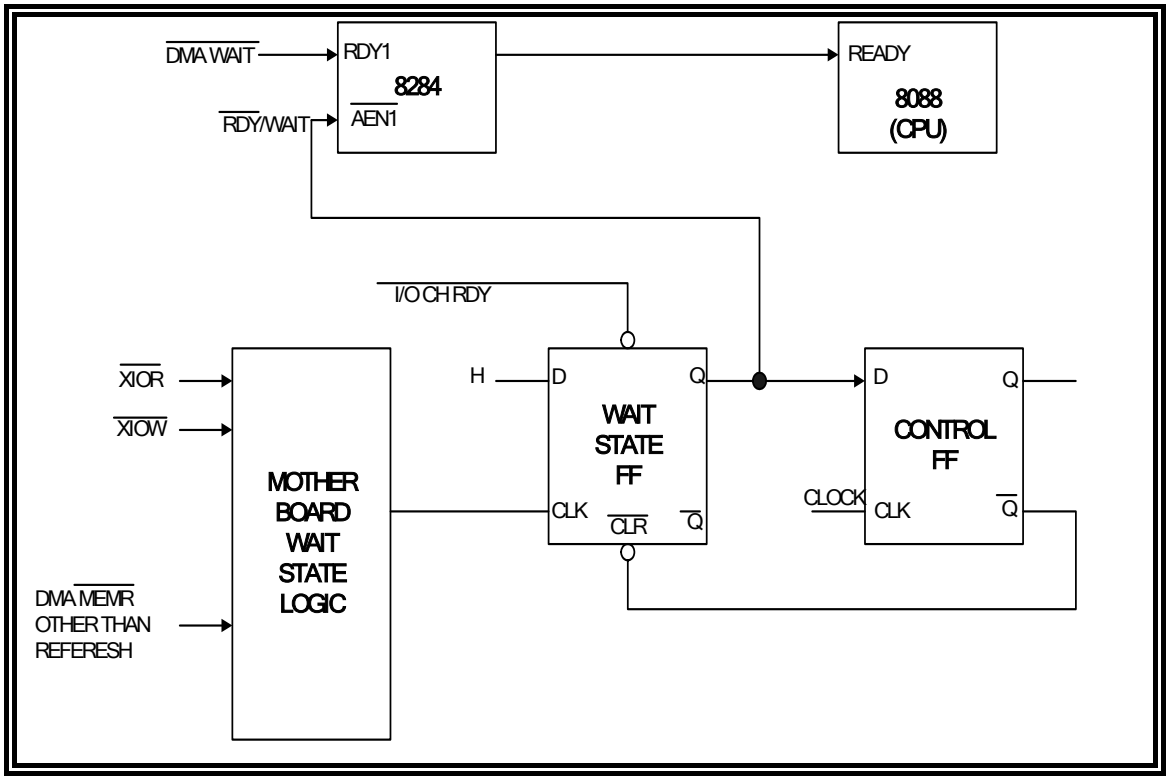


Figure 2.9 Wait state function for CPU and DMA controller

The WSL block receives five signals and generates two Wait State signals. The incoming signals are $\overline{\text{DMA WAIT}}$, $\overline{\text{XIOR}}$, $\overline{\text{XIOW}}$, $\overline{\text{XMEMR}}$, and I/O CH READY.

The DMA WAIT signal is generated from the Bus Arbitration Logic (BAL) block while the XIOR, XIOW, and XMEMR signals are received from the Control Bus Logic block.

The only signal which is received from the external memory device, or interfacing device installed in the I/O slots (channel A10) is the I/O CH RDY.

$\overline{\text{XIOR}}$, $\overline{\text{XIOW}}$, and $\overline{\text{XMEMR}}$ are the control signals generated either by the DMA controller or by the bus controller. Prefix "X" indicates that these signals are buffered.

The two output signals generated by Wait State Logic block are $\overline{\text{RDY}}$/WAIT, and READY To DMA.

These signals cause a Wait State in the CPU or the DMA controller respectively.

When the DMA WAIT, connected to the input RDY1 of the clock generator, is made low then the clock generator ensures that the READY input of the microprocessor is also made low.

This makes the microprocessor enters the Wait State.

Similarly, when the $\overline{\text{RDY}}$/WAIT signal at the $\overline{\text{AEN1}}$ input of the clock generator is made high then also the CPU enters the wait state.

The $\overline{\text{DMA WAIT}}$ signal is directly given to the RDY1 input of the clock generator from the BAL block.

A low $\overline{\text{DMA WAIT}}$ transfers the bus to the DMA Controller and the CPU is kept in the wait state until the DMA controller release the control of the bus.

$\overline{\text{RDY}}$/WAIT signal is made high when any one of the input signals - I/O CH RDY, $\overline{\text{XIOR}}$, $\overline{\text{XIOW}}$, or $\overline{\text{XMEMR}}$, goes low.

When anyone of these signals goes low, the flip-flop in WSL block is set and output Q i.e. $\overline{\text{RDY}}$/WAIT of this flip-flop goes high.

This flip-flop is controlled by yet another flip-flop, called the Control flip-flop. When the main flip-flop is set then the control flip-flop resets the main flip-flop after one clock cycle and the CPU comes out of the Wait State.

The I/O CH RDY input is received from the expansion cords. If this signal is low for more than one clock cycle, the main flip-flop continues to be in the set state until the I/O CH RDY signal is made high again.

The control flip-flop resets the WS flip-flop at the end of the clock cycle.

The $\overline{\text{Q}}$ outputs of both flip-flops are used for generating the READY To DMA signal. This signal is connected to the READY input of the DMA controller.

A low READY To DMA signal introduces wait state in the DMA controller for one clock cycle.

BUS ARBITRATION FUNCTIONAL BLOCK:

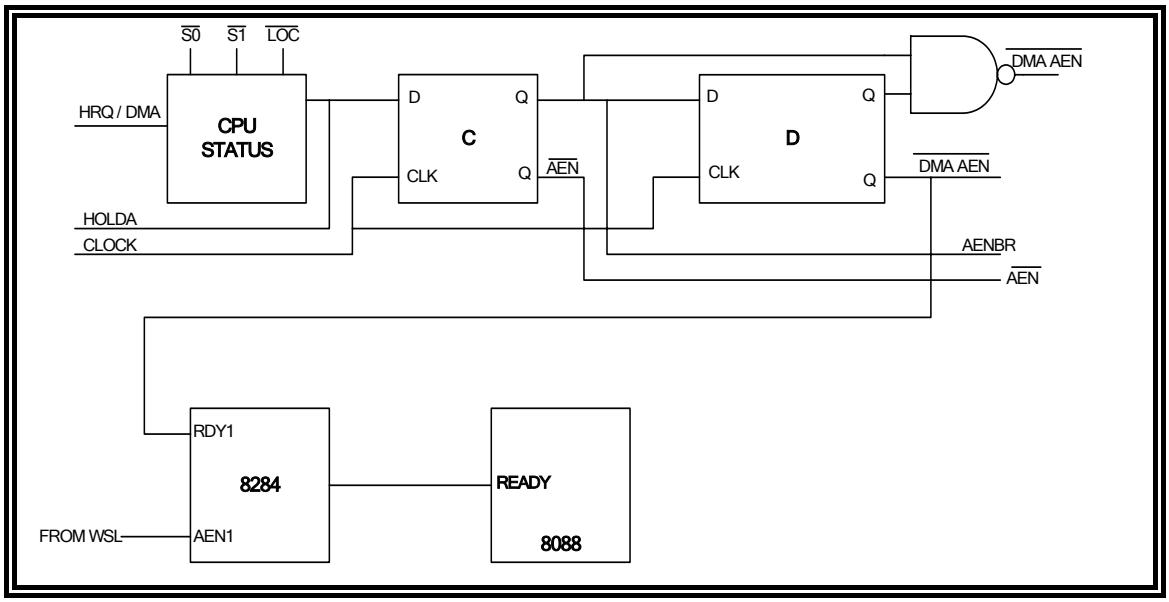


Figure 2.10 Bus arbitrations functional block

Figure 2.10 illustrates bus arbitrations functional block. The Bus Arbitration Logic block manages the system bus. This block decides when the CPU should be given control of the bus and when the DMA controller should use the bus.

All the memory devices and the I/O devices generate their respective bus requests in the form of DRQ signals.

The DMA controller receives these bus request signals. As soon as the DMA controller receives any DRQ signal, it immediately generates a HRQ DMA signal for the BAL block.

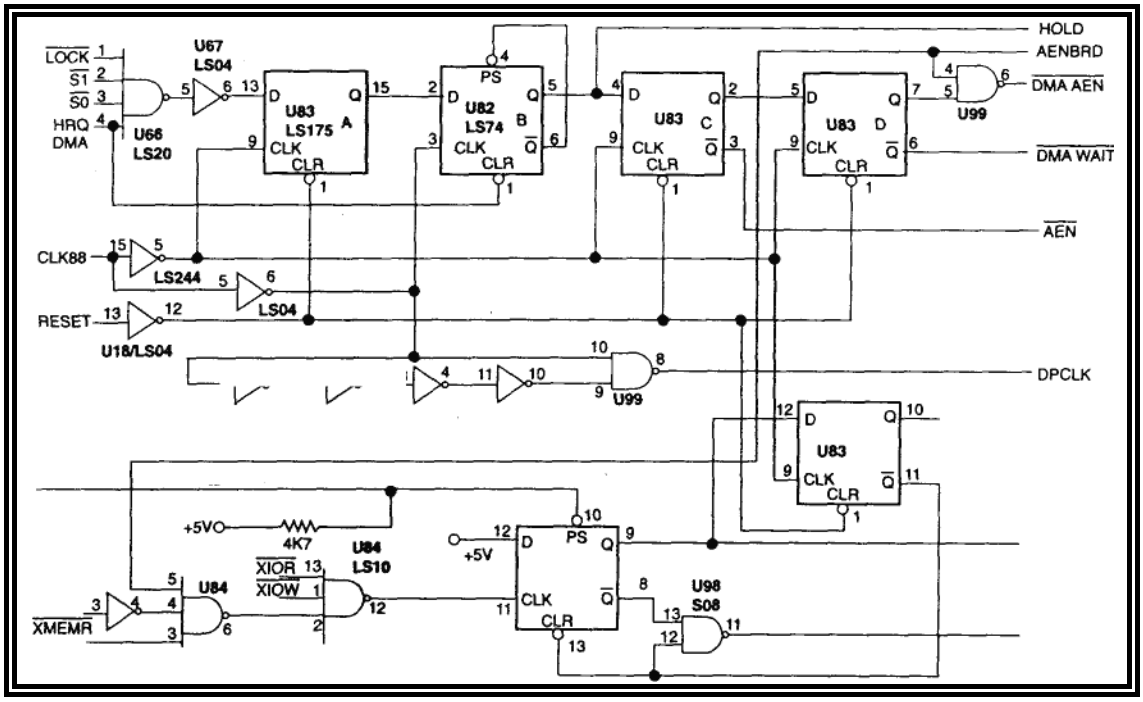


Figure 2.11 Bus arbitration functional block with wait state

Figure 2.11 shows the functional block diagram of bus arbitration block with wait state functional block. The BAL block continuously monitors the conditions under which the system bus is being used by the CPU through the $\overline{SO}$ and $\overline{S1}$ status signals it receives from the microprocessor.

In case, the CPU is actively using the bus and does not want the bus to be transferred to any other device, it pulls the $\overline{LOCK}$ signal to low state. This signal is also monitored by the BAL block.

When the BAL block detects that the microprocessor has free bus cycles, it generates HOLDA (HOLD Acknowledgment) signal to inform the DMA controller that its HRQ DMA signal has been acknowledged.

Now the BAL block drives the $\overline{\text{DMA WAIT}}$ signal low. This ensures that the CPU remains in the wait state while the DMA controller is using the bus.

BAL block also makes the AEN BRD signal high in response to a low $\overline{\text{AEN}}$ signal. These two signals are given to the $\overline{\text{AEN}}$ and CEN Inputs of the bus controller respectively.

This disables outputs of the bus controller. In this manner, the BAL ensures that no control signals from the bus controller reach the control bus while the DMA controller is using the bus.

The address bus is divided in two groups on the motherboard. One of these, the 4 I/O Slot address bus (A bus) is used with the expansion cards installed in the I/O slots. Second is the System Address Bus (XA bus) which is used for transferring the 1 address Bits between the various logic blocks of the motherboard.

When the DMA controller is using the bus, it becomes necessary to isolate the XA bus from the CPU.

To achieve this, the BAL block generates a high AEN BRD signal to the bus \ address logic block. But if the CPU is using the bus and DMA controller is in the wait state, the BAL block generates a low $\overline{\text{DMA AEN}}$ signal to the address bus logic block. This isolates the A bus from the DMA controller.

In a similar manner, when the bus is in use by the DMA controller, a high AEN BRD signal is generated from the BAL block to the control bus logic block. This isolates the data bus from the CPU. Thus, all the three buses are isolated from the CPU while the DMA controller is in control of the bus.

These buses are isolated from the DMA controller during the time when CPU is using these. The control of the system bus remains with the DMA controller for as long as the HRQ DMA signal remains active.

When the DMA controller has no more requests for the bus from the memory devices or I/O devices, it deactivates the HRQ DMA line. The BAL block reverts all signals to the previous state and transfers the bus to the CPU.

The BAL block receives DACKO BRD signal whenever a memory refresh cycle is to be completed. The BAL, block disables the wait state during this period because if the memory were not refreshed, all data would be lost.

CONTROL BUS FUNCTIONAL BLOCK

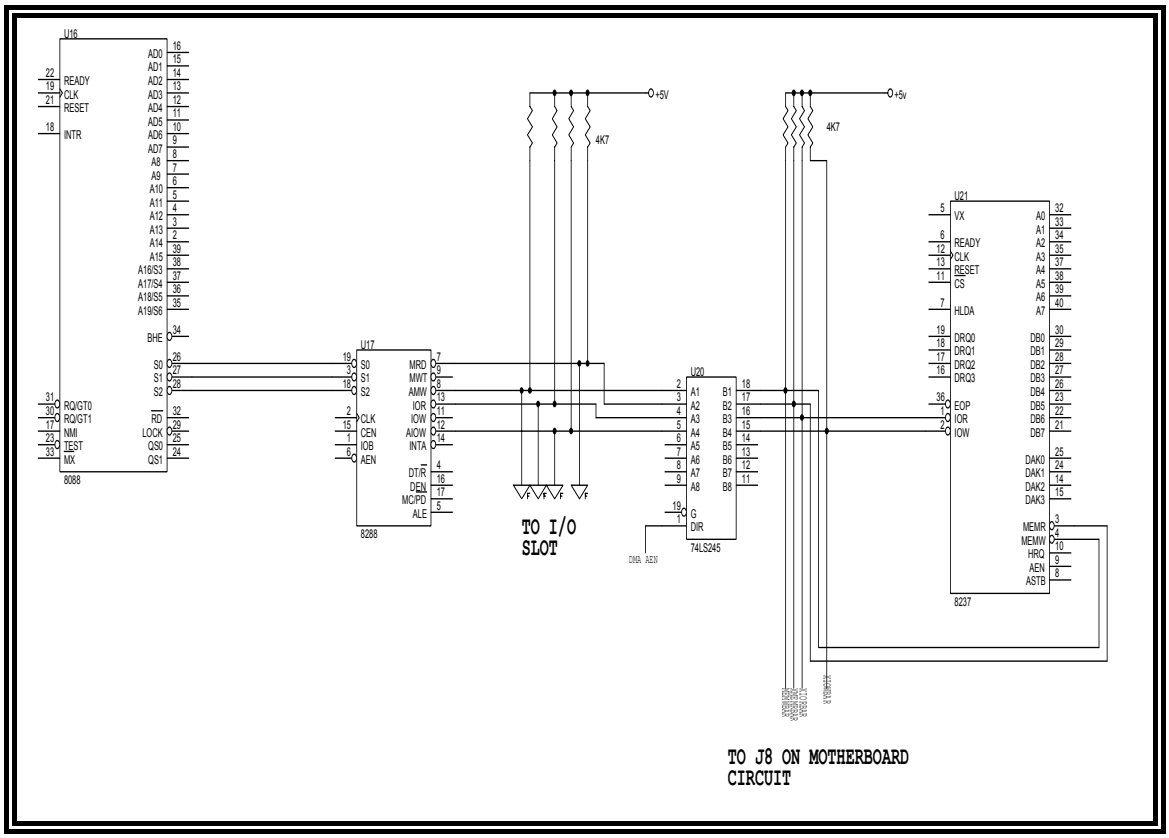


Figure 2.12 Functional block of control bus

Figure 2.12 illustrate functional block of control bus. This control bus logic block generates control signals required for interfacing various peripheral devices installed at the I/O ports, memory devices and different supporting chips with the microprocessor.

This logic block includes the bus controller chip, clock generator, and the DMA controller for generating the necessary control signals.

The Control Bus Logic block may be divided in following subgroups:

- Control Signal Logic Block, and
- Bus Command Logic Block

The control signal logic block manages or controls the data and address bus connected with the microprocessor by generating the DTR, $\overline{\text{DEN}}$, and ALE signals.

When the $\overline{\text{DTR}}$ goes high then data flows from the CPU towards the interfaced device but when this signal is low then the flow of data is reversed and data is transferred from the external device towards the CPU.

When the DEN is goes high then the data bus transceiver, which is used as a buffer between the Bus controller and the DMA controller, gets activated. But when the DEN signal goes low then this buffer, becomes disabled.

The transceiver, when enabled, passes the control signals generated by the bus controller or the DMA controller to the I/O slots from one side and also to the various logic blocks of the motherboard from its other side.

Bus Command Logic block generates the $\overline{\text{MEMW}}$, $\overline{\text{MEMR}}$, $\overline{\text{IOW}}$, $\overline{\text{IOR}}$, and $\overline{\text{INTA}}$ control bus signals. These four bus control signals are buffered by the transceiver and given to the I/O slots from its one side and to the various motherboard logic blocks from its other side.

Transceiver LS245 is used on PC-XT motherboard to perform this function.

Its pin $\overline{\text{DIR}}$ is used to select the control signals generated by the bus controller or by the DMA controller.

When the input signal $\overline{\text{(DMA AEN)}}$ is high, the transceiver selects the control signals generated by the bus controller. When this input is low, the control signals generated by the DMA controller are selected by the transceiver.

Control signals are generated from bus command logic block which are used to bring the CPU in a wait state.

The CPU bus status is recognized by this bus command logic block according to the status of $\overline{\text{SO}}$, $\overline{\text{S1}}$, and $\overline{\text{S2}}$.

When the status signals $\overline{\text{SO}}$ and $\overline{\text{Sl}}$ are high while $\overline{\text{S2}}$ is low, microprocessor enters the wait state but when $\overline{\text{SO}}$, $\overline{\text{S1}}$, and $\overline{\text{S2}}$ are all high, it indicates the end of bus cycle will follow.

These status signals are generated from the microprocessor.

When the bus is being used by the microprocessor, the different control signals are generated by the bus controller, not by the DMA controller.

Other than the four control signals generated from the Control Bus Logic block, three other signals - AEN BRD, AEN, and DMA AEN, also arrive in this logic block from the bus arbitration logic group.

These signals are generally referred to as supervisory signals.

When the CPU is using the bus, the AEN BRD signal goes low while the $\overline{\text{AEN}}$ and $\overline{\text{DMA AEN}}$ signals become high. The low AEN BRD signal and high $\overline{\text{AEN}}$ signal enable the bus controller.

When the bus is with the DMA controller the $\overline{AEN}$ signal becomes low while the $\overline{AEN\ BRD}$ signal goes high. This disables the bus controller. At the same time, $\overline{DMA\ AEN}$ signal becomes low and the transceiver selects the control signals generated by the DMA controller

ADDRESS BUS FUNCTIONAL BLOCK:

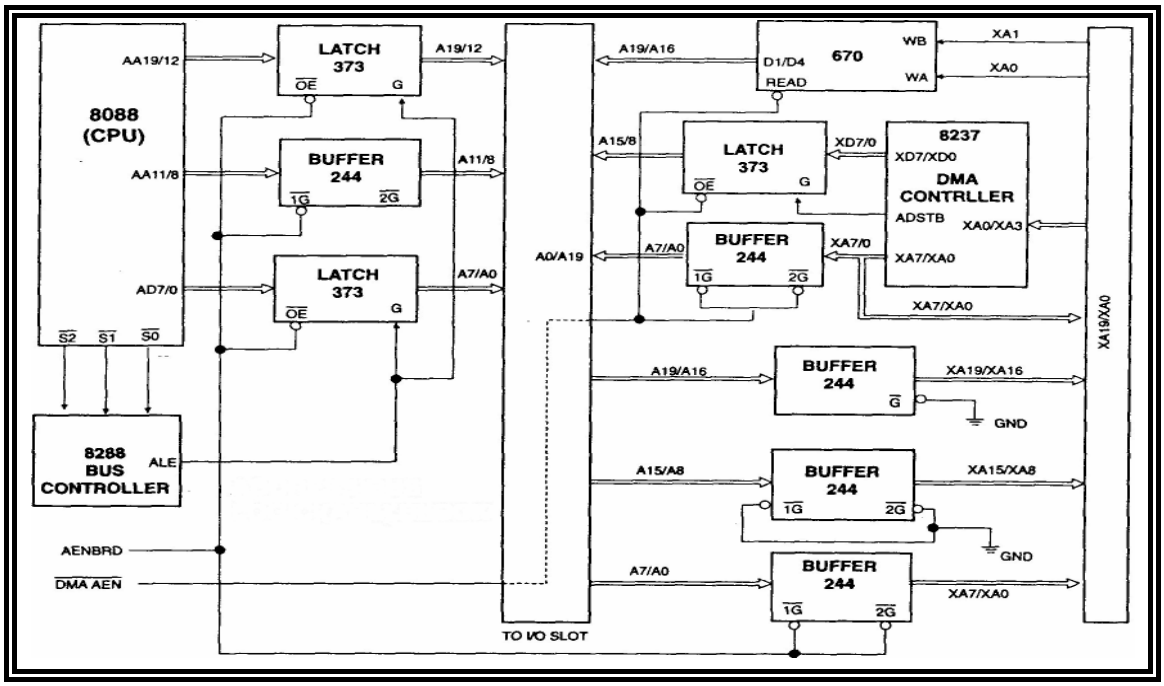


Figure 2.13 (A) Address Bus Functional Block

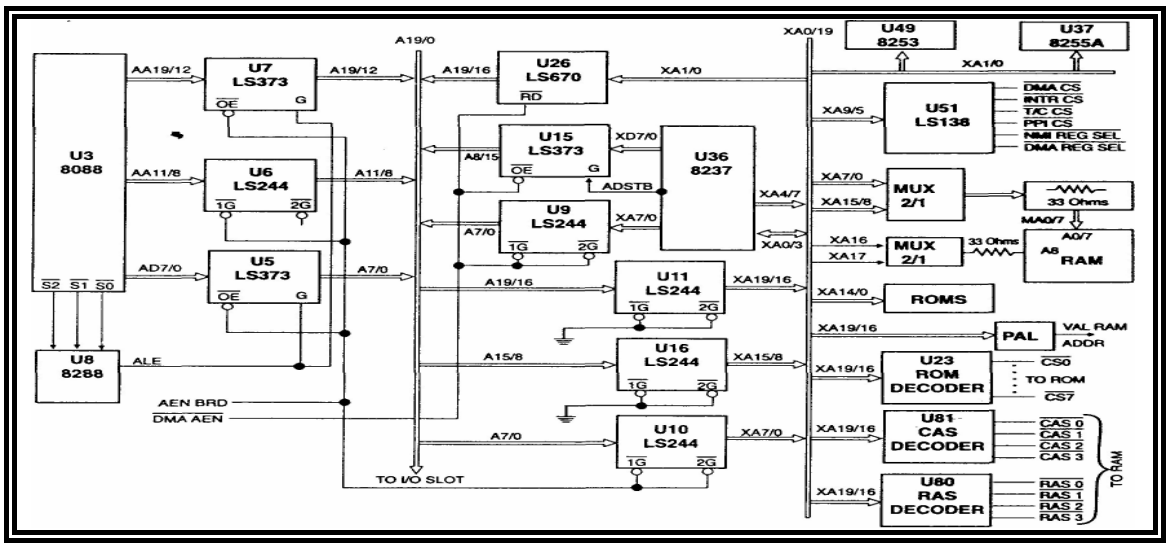


Figure 2.13 (B) XA Bus Logic Address Bus Distribution

Fig. 2.13 illustrates address bus functional block. Address bus of the motherboard is formed either by the CPU logic block or by the DMA logic block. This is a 20-Bit bus

and it is used for transferring the address Bits between the various logic blocks and the I/O slots.

The address bus is used on the motherboard as

- I/O Slot Address Bus A-Bus
- XABus

A-Bus transfers the address Bits between the I/O slots and the various logic blocks of the motherboard.

On the other hand, the XA bus transfers the address Bits amongst the different logic blocks of the motherboard. Thus, the XA bus does not interact with the devices interfaced at the I/O slots.

All the 20 address lines of the address bus i.e. AO-A19 or XAO-XA19 are routed through latch ICs or buffer ICs.

This process is essential in the motherboard circuitry because when the bus control is with the DMA controller, the address bits of the CPU must be prevented from reaching the A-Bus. Also, when the CPU is using the bus, the address Bits of the DMA controller must not reach the A-Bus.

The AEN BRD and the $\overline{\text{DMA AEN}}$ signals received from the buffer, BAL block control these buffers and latches.

The 20-Bits of the A-bus from the CPU are grouped as ADO-AD7, AA8-AA11, and AA12-AA19. Latch ICs are used for the Bits ADO-AD7 and AA12-AA19. A Buffer 1C is used for the Bits AA8-AA11.

The Bits AA8-AA11 are not latched but only d or tri-stated while the remaining 16-Bits are routed through the latch ICs.

When the bus control is with the DMA controller, the AEN BRD signal connected to the tri-state logic control terminal $\overline{\text{OE}}$ of these latch ICs is made high by the BAL block.

Now the address Bits ADO-AD7 and AA12-AA19 remain latched as these are not available at the output, i.e. the outputs of the latches are neither low nor high.

This undetermined state of the outputs is normally known as "Tri-State".

Terminal G of the latch IC is given ALE signal from the bus controller logic group.

$\overline{OE}$	G	Input	Output
High	Ineffective	Tri-stated Low	-
Low	Low	Ineffective	No change
Low	High	Low	Low
Low	High	High	High

When the AEN BRD signal is high, output will always be tri-stated and the address Bits at the input will be isolated From the A bus. Any change can be made in the output only when AEN BRD signal is made low, and the ALE signal is high.
When AEN BRD is low i.e. when CPU is using the bus, and ALE is also low, the address Bit, which is already present at the output, will continue to stay there and it will not change in state even if the CPU changes the address Bit at the corresponding input.
The address Bit at the input will be transferred to the A Bus only when the ALE signal is high and AEN BRD is low.
AEN BRD signal is also given to $\overline{1G}$ pin of the buffer 1C which passes the buffered Bits AA8-AA11 on the A Bus, or otherwise tri-states these Bits.
When AEN BRD signal is high, the address Bits are tri-stated, if the AEN BED signal is low, the address Bits from the CPU are buffered and given on the A Bus.
Thus, all the 20-address Bits are made available to the A Bus when the bus is allocated to the CPU by the BAL block. All address Bits arriving from the CPU on the A Bus are also given to the XA Bus.
Bits AO-A7, A8-A15, and A16-A19 are routed through three buffer to reach the XA bus when the CPU is using the system bus. AEN BRD signal is given to the enable pin $\overline{1G}$ and $\overline{2G}$ of the first buffer (AO-A7).
These address/data Bits from the CPU can be loaded on the XA bus only when the bus control is with the CPU.
Other two buffers remain permanently enabled and the address Bits A8-A15 on the A-Bus, are always present on the XA bus.
Four MSBs of address bus (A16-A19 or XA16-XA19) are used for addressing the RAM memory.

When the bus control is granted to the DMA controller by the BAL block, the DMA controller loads the eight LSBs of the address bus AO-A7 on the A Bus through a buffer 1C.

$\overline{\text{DMA AEN}}$ signal from the BAL block is given to both $\overline{1G}$ and $\overline{2G}$ enable pins of this buffer. These lines are also connected directly to the XA bus.

The eight Bits A8-A15 are loaded on the A Bus through a latch IC. An AD STB pulse is used by the DMA controller for latching these eight address Bits. DMA controller gives this latching pulse to the G terminal of the latch.

$\overline{\text{DMA AEN}}$ signal is given to the $\overline{\text{OE}}$ terminal of this latch from the BAL block. When the bus is being used by the CPU then the BAL block drives the $\overline{\text{DMA AEN}}$ signal high and the address Bits A8-A15 are tri-stated.

The Page Register IC is used for the address Bits A16-A19.

These Bits are used for addressing the RAM memory. The DMA controller does not load these Bits. These four address Bits are generated by the CPU and get loaded in the page register.

$\overline{\text{DMA AEN}}$ signal is given from the BAL block to the $\overline{\text{READ}}$ terminal of the page register. When this signal is low, the page register enters the Read Mode but when this signal is high the page registers enters the Write Mode.

When the program in CPU has generated the four address Bits for the RAM, the BAL block makes the AEN BRD and the $\overline{\text{DMA AEN}}$ signal low.

Now the DMA controller enters the wait state and the RAM memory address is loaded in the registers of the page register.

The specific register in which this address must be loaded is determined by the Bits XA1 and XAO, supplied to the WA and WB terminals of the page register respectively from the DMA controller.

As soon as the bus is transferred back to the DMA controller, the page register enters the Write Mode and loads the MSBs A16-A19 on the A Bus.

As the RAM expansion card on the PC motherboard is installed in the 1/0 slots, this is not necessary to load these four MSBs (A16-A19) on to the XA bus.

DATA BUS FUNCTIONAL BLOCKS:

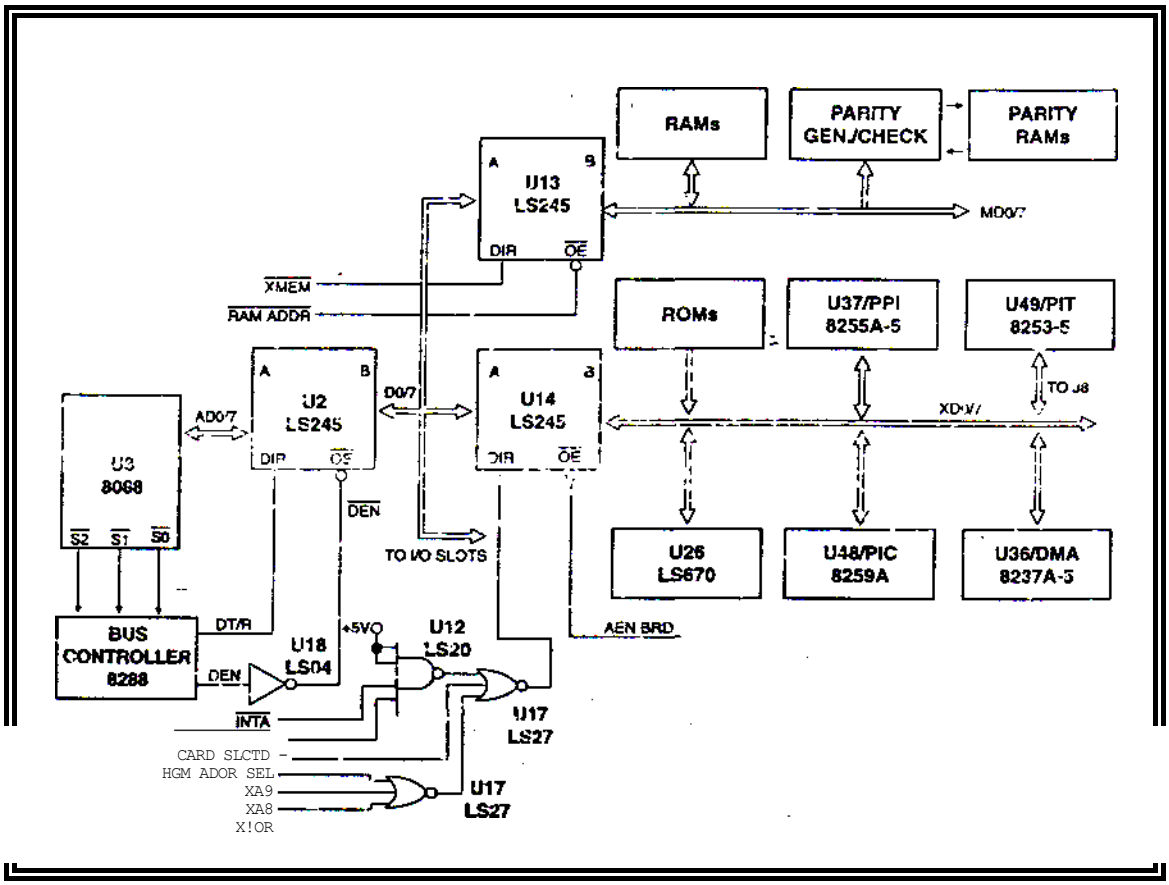


Figure 2.14 Data Bus Functional Block

Fig. 2.14 illustrates data bus functional block. When DT/R signal is made low by the bus controller, the transceiver passes data bits DO-D7 from the system bus towards the local bus ADO-AD7 of the microprocessor.

The transceiver is enabled by a low signal at its OE terminal during this time.

If the CPU wants to start a write cycle, the bus controller senses this and makes the DT/R signal high while keeping the OE terminal of the transceiver low.

Now the Bits ADO-AD7 are transferred from the local bus to the system bus.

If the bus control is with the DMA controller, the OE terminal of this transceiver is made high.

System bus data Bits DO-D7 are directly given to the I/O slots but isolated from the MDO-MD7 bus by another transceiver.

XMEMR signal is given to its DIR terminal from the bus control logic group while a valid RAM ADD SEL signal is given to the OE terminal from the RAM decoder circuit, Decoder PROM, or from PAL.

Any one of these three methods may be used in the XA bus logic group,. depending on the design adopted for the motherboard.

The RAM ADD SEL is kept low if a -valid address has been loaded but if the address on the address bus is not valid, this signal is held high.

As soon as XMEMR is made low the data Bits DO-D7 are transferred on to the MDO-MD7 bus. If the XMEMR signal is made high, while the RAM ADD SEL is low, the Bits MDO-MD7 get transferred on the DO-D7 bus.

The third transceiver is used for transferring the data between the DG-D7 bus and XDO-XD7 bus.

AEN BRD signal comes at its OE terminal from the BAL group. From now on, this transceiver will be enabled only when the CPU is using the bus i.e. when the AEN BRD signal is low.

When the DMA is having control of the system bus, this transceiver is disabled by a high AEN BRD signal from the BAL block, this isolates the XDO-XD7 bus from DQ-D7 bus.

Flow of data is controlled by the output signal given to the DIR terminal or trip third transceiver through a NOR gate.

First input of the NOR gate that controls the direction of data transfer is given from the output of four-input NAND gate.

INTA and CAR1DSLCTD signals form the two inputs to this NAND gate while the other two inputs ore permanently kept high [+5V).

INTA is normally held high but it is mode low (active) as soon as the IRQ request signal from some device on the I/O slots has been accepted.

The CARD SLCTD signal from the slot 18 is permanently kept high. Hence, under normal condition output of this NAND gate is low. Second input of the controlling NOR gate is directly connected to the ROM ADD SEL signal from the ROM decoder logic through a two-input NOR gate.

ROM ADD SEL is normally kept low by the normally high (active low) VAL ROM ADDR and XMEMR signals. When the CPU wants to communicate some data with the ROM or with the I/O ports or with the DMA controller then the VAL ROM ADDR and XMEMR signals are made active low and the ROM ADDR SEL signal is driven high.

The last input to the NOR gate is connected with another three-input NOR gate, this controls the direction of the third transceiver.

The three inputs of this NOR gate are the Bits XA.9, XA8, and the XIOR signal.
The address Bits XA8 and XA9 of all the Input/Output ports on the XA bus are always at a low logic level. The XIOR signal (active low) is normally kept high. This makes the output of this NOR gate normally low.
Under normal conditions, the first input to the direction controller NOR gate is low, second input is low, and the third input is also low.
This gives a high output at the DIR terminal.
As the data transfer in the system is a continuous process, the status at the DIR and OE terminal of the transceivers is never permanent. Pulses continuously arrive at these terminals and the data continues to flow from one direction to the other on the data bus.
The INT A signal is made low after the CPU accepts the IRQ signal. This drives the output of the NAND gate high, which results in a low signal at the DIR terminal. As a result, data Bits from XDO-XD7 bus are transferred to the DO-D7 bus for the requesting device at I/O slot.
If the CPU has to access some data from the I/O port on the XD bus, the ROM ADDR SEL signal goes high (because of low MEMR and low VAL ROM ADDR at the inputs of the NOR gate in the ROM decoder logic block).
The data from the XDO-XD7 bus reaches the CPU local bus from the DO-D7 bus.
The expansion card for the J8 (eighth) slot uses the CARD SLCTD signal. This is a special card used in some IBM PC-XT motherboards. As this card is not much in use now, this CARD SLCTD signal is always kept high.

SERIAL PORT FUNCTIONAL BLOCK:

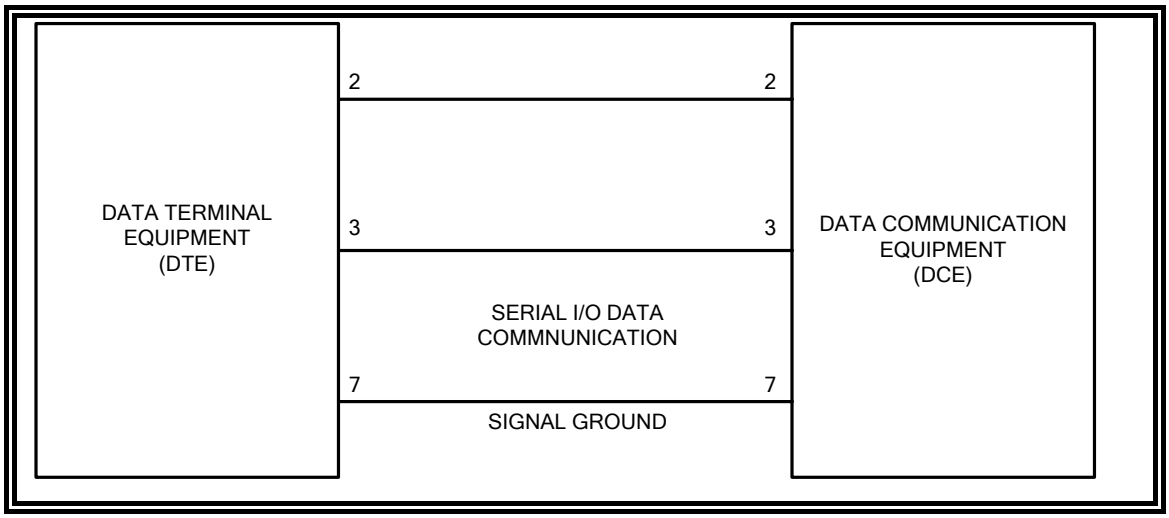


Figure 2.15 serial communication using DTE and DCE

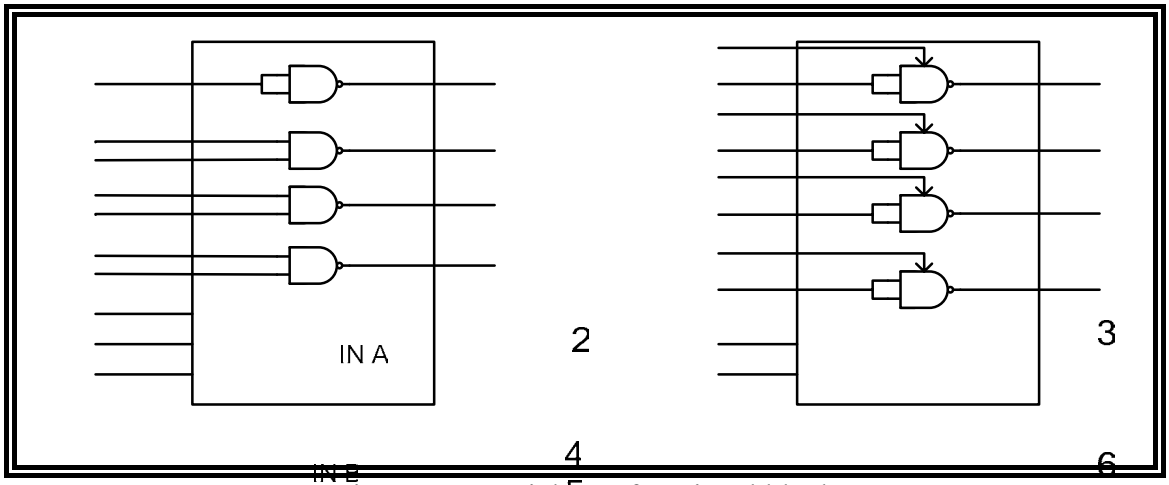


Figure 2.16 serial port functional block

For serial data transmission computer peripherals (Data terminal Equipment) and modem (Data communication equipment), RS 232 data transmission method is to be used. Figure 2.15 shows communication both DTE and DCE. Figure 2.16 illustrates serial port functional block. RS232 data transmission method is used for serial data transmission between the Data Terminal Equipment (DTE) such as the computer printer, mouse, scanner etc. and the Data Communication Equipment (DCE) such as the modem.

The main drawback of RS232C transmission system is that the voltages used for data transmission are not compatible with TTL devices. Also noise develops in the signals if longer interfacing cables are used for data communication.

Four types of signals are transmitted from the Serial Ports

- Data signals, Control signals, Timing signals, and Ground-level signals.

Control signals are used for transmitting data between the microprocessor and the interfaced peripheral devices these are called handshake signals.

During the transmission

- voltage level from +3V to +15 volts is considered low (0), and
- voltage from -3 volts to -15 volts are considered as high (1). Generally +12 volts and -12 volts are used for these two logic levels.

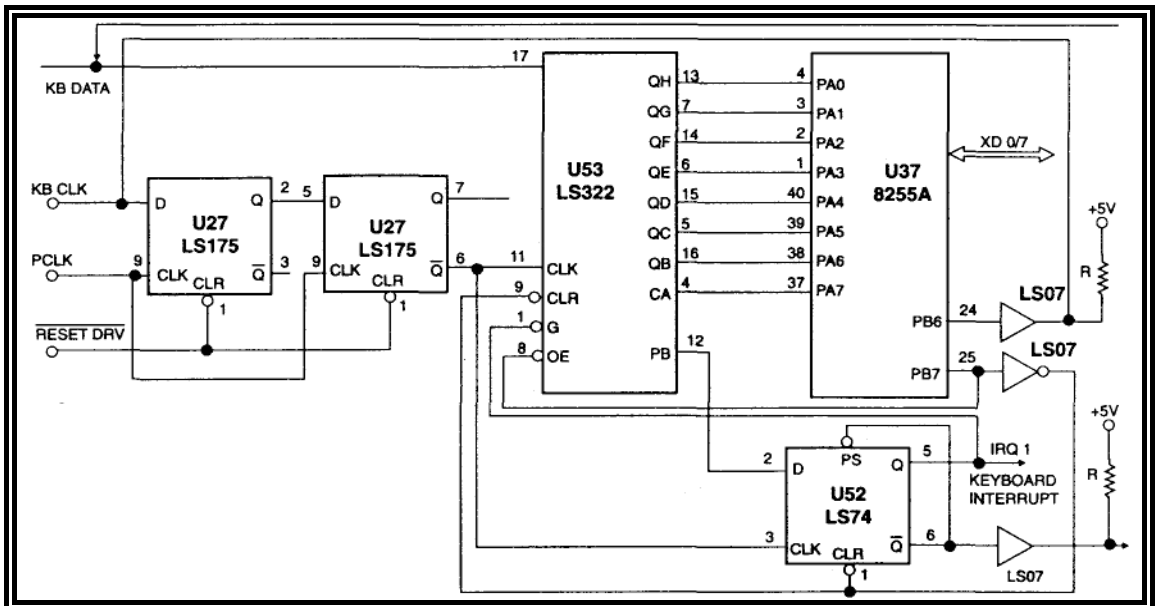


Figure 2.17 Functional Block Diagram of Keyboard Interface

Keyboard is the most common input device for a computer system. Whenever a key is pressed on the keyboard, a scan code related to that particular key is generated by the keyboard.

This scan code generated by the keyboard is received by the Keyboard interface Logic block on the motherboard.

Sometimes an additional line is also used to carry the RESET signal from the motherboard to the keyboard.

Two shielded wires are used to carry the +5 volt and ground supply to the keyboard.

The serial data Bits are converted into eight Bit scan code by the keyboard interface logic block.

Later, an interrupt is raised for the interrupt logic block.

The keyboard interface logic block of the PC motherboard comprises of Shift Register, interrupt Generator, and PPI.

The shift register is used for storing the serial scan code Bits in the eight Bit scan code format, the interrupt generator raises the interrupt for the interrupt logic block.

Outputs of the shift register are connected to the port-A of the PPI. As soon as the CPU reads the scan code at the port-A, the shift register is cleared for the next scan code.

MODE SWITCH INPUT FUNCTIONAL BLOCK:

DIP switches are used to indicates the hardware configuration to the ROM BIOS at the time of system startup.

DIP switches are set to indicate whether the POST should be performed periodically, whether the coprocessor is installed, the memory capacity of the installed RAM bank, the type of the installed display card, and the number of installed floppy drives etc.

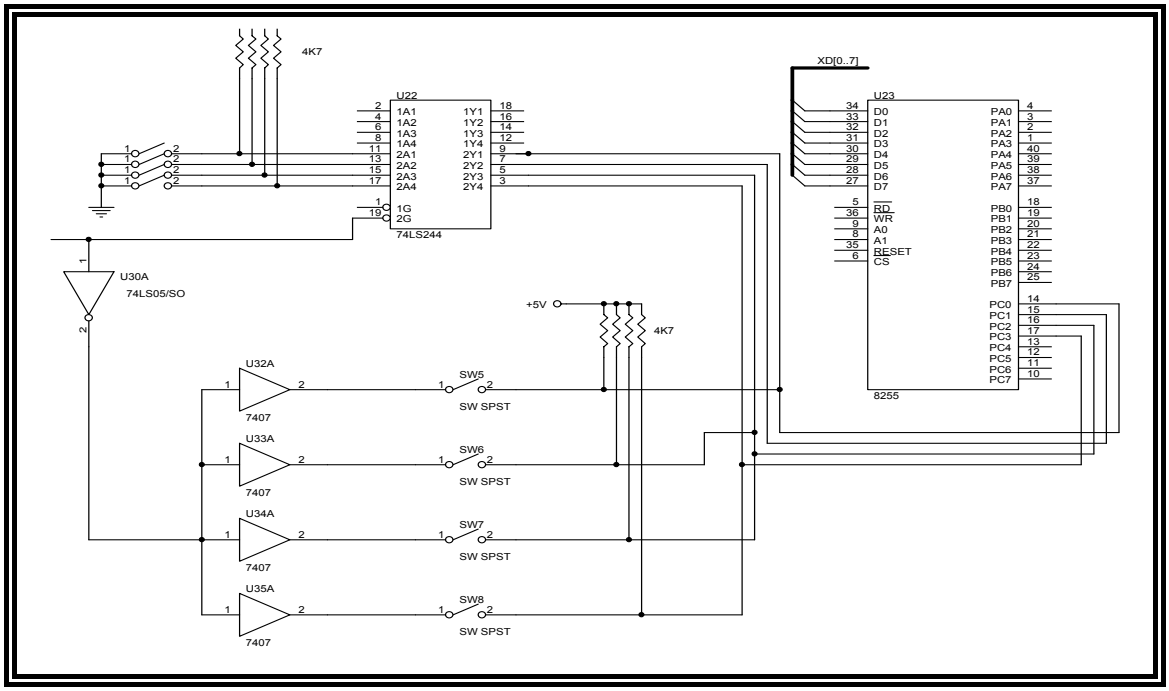


Figure 2.18 Functional diagram of mode switch input this block

Figure 2.18 shows functional diagram of mode switch input this block is formed by 8 DIP switches, 8255 two buffers and in inverter. Mode Switch Input Logic Block is formed by eight DIP switches, a programmable peripheral interface, two buffers and an inverter.

First four switches are routed through a buffer enabled by a low input at its 1G terminal. The four inputs to this buffer are connected to the positive +5 volt line through 4.7K pull-up resistors.

Each terminal of the four switches is directly connected to the ground level supply line. When the buffer is enabled and if any of the first four switches are closed, the corresponding outputs go low.

All the buffer outputs remain high when all of these four switches are open. The buffer is enabled if 1G is low. If the controlling gate 1G of the buffer is made high, the buffer outputs are tri-stated.

Signal from the line PB3 is also given to the four inputs of another buffer 1C through an inverter. All the Four corresponding outputs of this buffer are directly connected to each terminal of the last four DIP switches.

Outgoing terminals of the DIP switches are pulled high through four 4.7K pull-up resistors and are directly connected with the four output terminals of the first buffer. Finally the outputs are given to the four input terminals PCO, PCI, PC2, and PC3 of the PPI. These inputs form the port-C.

If the line PB3 is low, the first buffer will be enabled while the outputs of the last four switches will remain high. In this situation, the inputs to the port-C can be made low only when any of the first four switches are closed.

If the line PBS is high, the outputs of the first buffer will be tri-stated and the input to the second buffer will be low. If one of the last four switches is closed, the corresponding input to the port-C of the PPI will become low.

The PPI loads the input signals on the XDO-XD7 when the CPU issues an INT instruction as per the software.

SPEAKER FUNCTIONAL BLOCK

Speaker logic block is formed by the Timer-2 in the PC motherboard and controlled by software. A speaker driver circuit is also used to generate sound (tones) from the speaker.

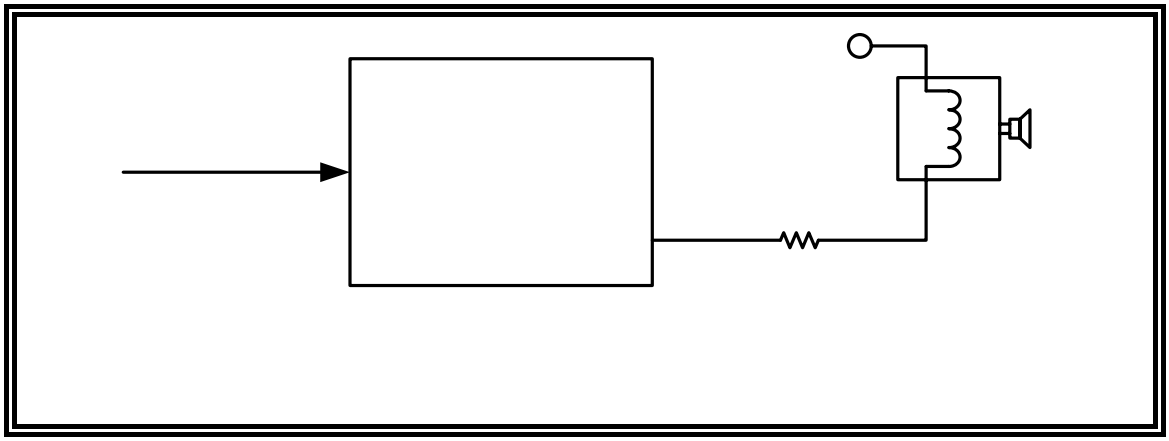


Figure 2.19 Speaker functional block

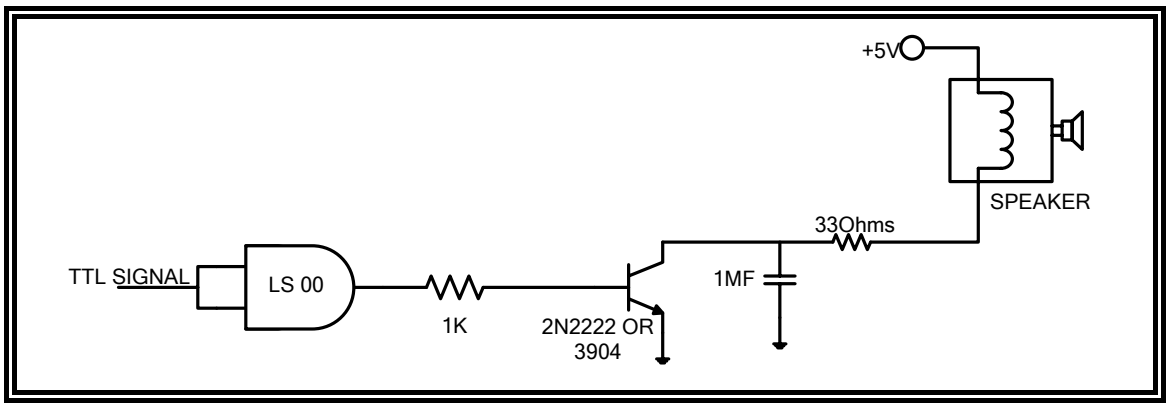


Figure 2.20 Speaker driving functional with power transistor

Figure 2.19 is the block diagram of the speaker functional on the motherboard. Whenever sound is to be generated from the speaker, software instructs timer-2 to generate a square waveform, which is given to the speaker driver circuit.

The output from the timer-2 can also be modulated by using gates to generate different tones by the software.

The speaker driver circuit is required because the TTL chips do not possess output current that is strong enough to directly drive the speaker.

The current required to drive the normal speaker in a PC is about 200mA to 300mA. Many different methods are used to achieve this driving current, such as use of relay driver ICs or use of power transistors etc. Figure 2.20 shows speaker driving functional through power transistor.

2.3 Software – requirement to communicate hardware:-

In order for any computer to function, it must have software to run on it. All that a processor--or any hardware for that matter--knows how to do is to follow instructions. The software is that collection of instructions, as described in this part of the Introduction. All regular programs that you run on your PC are stored permanently on your hard disk, and are loaded into your system memory (RAM) when you need to use them. From the memory, the processor can access the instructions coded into the program and run them, which lets you do your work.

When you first turn on your PC, the processor is "raring to go", but it needs some instructions to execute. However, since you just turned on the machine, your system memory is empty; there are no programs to run. To make sure that the BIOS program is always available to the processor, even when it is first turned on, it is "hard-wired" into a read-only-memory (ROM) chip that is placed on your motherboard.

A uniform standard was created between the makers of processors and the makers of BIOS programs, so that the processor would always look in the same place in memory to find the start of the BIOS program. The processor gets its first instructions from this location, and the BIOS program begins executing. The BIOS program then begins the system boot sequence which calls other programs, gets your operating system loaded, and your PC up and running.

The BIOS program is always located in a special reserved memory area, the upper 64K of the first megabyte of system memory (addresses F000h to FFFFh). Some BIOSes use more than this 64K area.

The BIOS provides various software routines (subprograms) that can be called by higher-level software such as DOS, Windows, or their applications, to perform different tasks. Virtually every task that involves accessing the system hardware has traditionally been controlled using one or more of the BIOS programs (although many newer operating systems now bypass the BIOS for improved performance). This includes actions like reading and writing from the hard disk, processing information received from devices, etc.

BIOS services are accessed using software interrupts, which are similar to the hardware interrupts except that they are generated inside the processor by programs instead of being generated outside the processor by hardware devices. One thing that this use of interrupts does is to allow access to the BIOS without knowing where in memory each routine is located.

Normally, to call a software routine you need to know its address. With interrupts, a table called an *interrupt vector table* is used that bypasses this problem. When the system is started up, the BIOS puts addresses into this table that represent where its routines are located for each interrupt it responds to. Then, when DOS or an application wants to use a BIOS routine, it generates a software interrupt. The system processes the interrupt, looks up the value in the table, and jumps to the BIOS routine automatically. DOS itself and application programs can also use this interrupt vector table.

CHAPTER 3

SINGLE BOARD COMPUTER

3. 1 Introduction

After complete the study of PC-XT motherboard, I have decided to do some practical work. So, I have decided to build one motherboard. I found that by using in-house facilities I can build double layered 8088 motherboard. As this would give some exercise for practical training I tried to build an 8088 system given in a book named “An introduction to the Intel family of microprocessor” By James L. Antonakes, Pearson Education Asia. In fact this is the motherboard and many projects do not require the new technology processors like Pentium, Celeron etc. An 8088 motherboard will do a fine job controlling robot arm, calculations, logical solution for simple projects, for example, where as new technology processors would be overkill.

I have used MASM assembler, EPROM programmer to write the code (BIOS software for this motherboard) and burn into the EPROM. I have used EPROM programmer available in the research lab of Department of Electronics.

Particularly in this chapter I want to explain design, build, study of instruction set of 8088, and software understanding to build 8088 motherboard.

Requirements of this motherboard are the same as those of any computer system and consist of four sections

1. CPU Section
2. Clock Generation Section
3. Memory Section
4. Input Output Section

I have used personal computer as communication console for the single board computer. To communicate both systems I wired single board computer and personal computer together. After interfacing my system I have used terminal emulation program on the personal computer. Basically this terminal emulation program will do all that is necessary to provide serial INPUT / OUTPUT between both systems.

After study of this chapter one can design, build and write appropriate software for own computer from scratch.

For this system only fifteen ICs are needed.

List of integrated circuits:

1. 8084A : Clock generator : Quantity=1
2. 74LS07 : Buffer : Quantity=1
3. 74LS04 : Hex Inverter : Quantity=1
4. 8088 : microprocessor : Quantity=1
5. 74LS244 : Octal Buffer : Quantity=1
6. 8282 : Octal Latch : Quantity=1
7. 8286 : Bi-directional bus driver : Quantity=1
8. 8288 : Bus controller : Quantity=1
9. 74LS138 : Three to eight line decoder : Quantity=2
10. 2764 : 8K x 8 EPROM : Quantity=1
11. 6264 : 8K x 8 RAM : Quantity=1
12. 8251 : USART : Quantity=1
13. MC441 : Baud-rate generator : Quantity=1
14. MAX232CPE : TTL to RS232 converter : Quantity=1

3.2 CPU section:

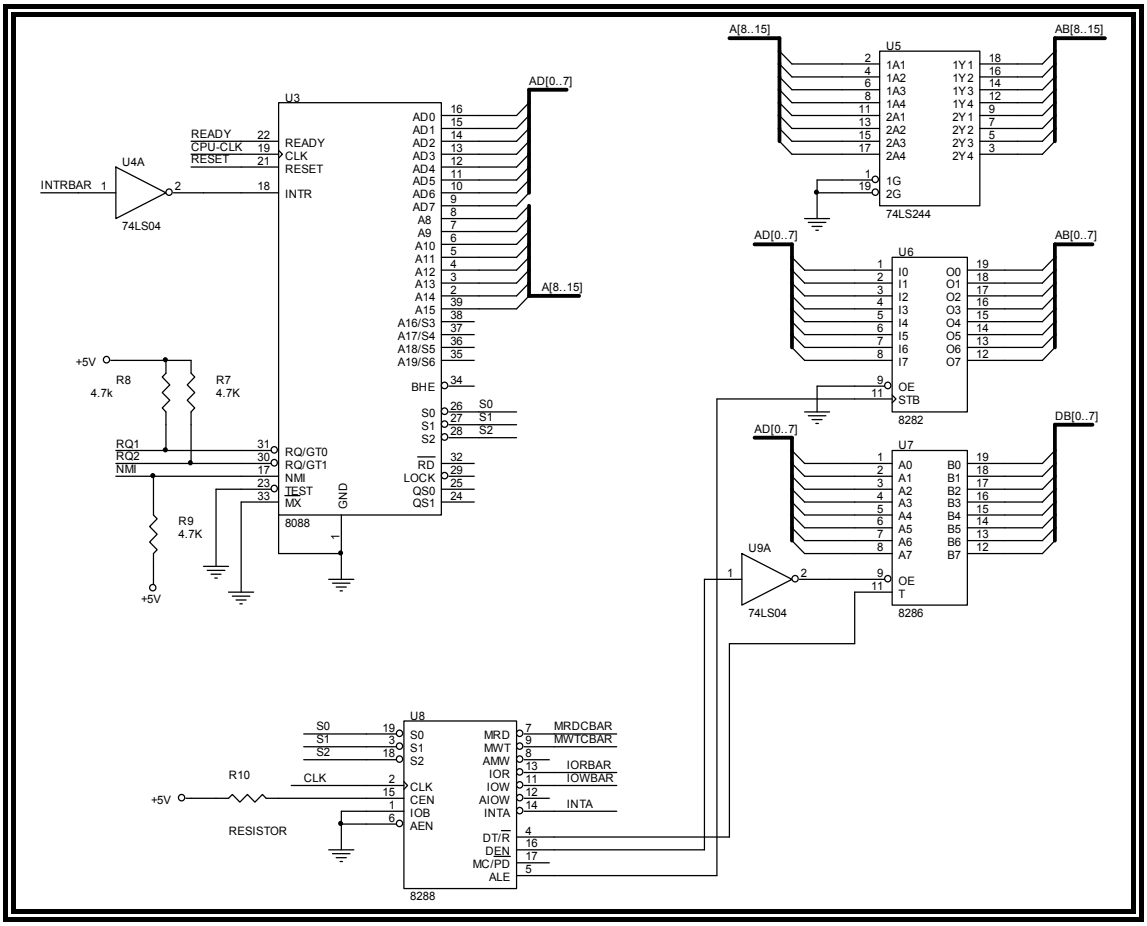


Figure 3.1 shows schematic diagram of CPU section.

I have decided to use microprocessor 8088 in its maximum mode. As shown in the Figure 3.1 pin no. 33 (MN/$\overline{MX}$) of 8088 is connected to ground.

In design of minimal system no device will be connected pin 30 and 31 named as $\overline{RQ}$/$\overline{GT1}$ and $\overline{RQ}$/$\overline{GT2}$ respectively, made high by use of TWO 4.7K pull-up resistors. These pull-up resistors will not prevent us from connecting a DMA device to either input in future expansion. Data lines and address lines of 8088 are used in both the INPUT / OUTPUT and memory section. Our system contains RAM, EPROM, SERIAL INPUT / OUTPUT and MEMORY section. I have decided to buffer the address and data lines for better performance. To drive address lines A8 – A15, IC 74LS 244 OCTAL BUFFER is used. Address lines A0 – A7 are multiplexed with the data lines D0 – D7. I used 8288 OCTAL LATCH, to generate the ALE signal to de-multiplex and drive address lines through A0- A7. These address lines A0 – A15 will allow for 64KB of system memory in design of minimal system. UPPER four address lines A16 – A19 are not used in this system.

We used 8286 bi-directional line driver / receiver to buffer data bus lines in two directions. The direction of data lines is controlled by pin no. 4 (DT/$\overline{R}$) of the 8288.

Pin $\overline{S0}$, $\overline{S1}$, $\overline{S2}$ are connected with 8288 bus controller. Pin no. 18 (INTR) of 8088 made low through inverter IC 74LS04. It shows no devices are connected in this system. Another interrupt pin no17 (NMI) is also edge sensitive. When no interrupting device connected, NMI will remain high state, and no interrupt will be requested.

3.3 Clock generation section:

Figure 3.2 shows clock generator for single board computer. We have used 8284 clock generator. Two of the pins X1 and X2 are directly connected to the crystal of 10 MHz.

The internal clock circuitry of the 8284 then generates the proper clock signal of 1/3 the crystal frequency for 8088 microprocessor. This frequency is fast enough to provide very quick execution of instructions.

We have used economic RAMs with longer excess time of 200ns.

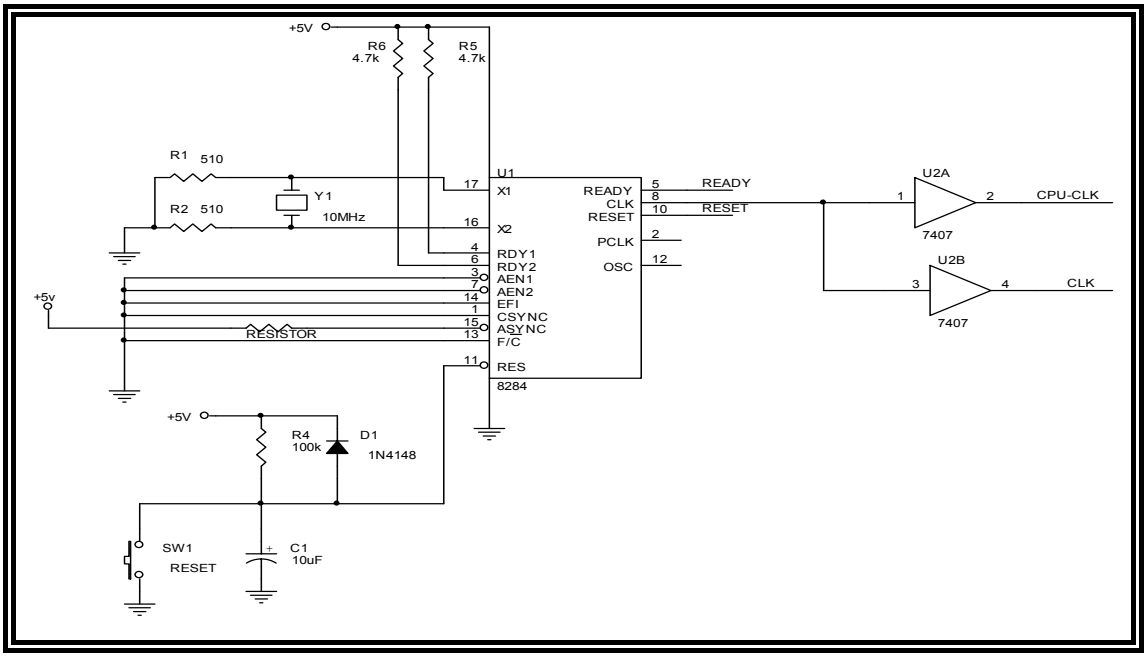


Figure 3.2 Clock generators for single board computer

To buffer the clock signal pin no. 8 of 8284 it is connected to the inputs of 7404 buffer IC, so that external loading on the CLK signal will not affect it's operation. First O/P of 7407 from pin no. 2 is directly applied to CPU clock, while second O/P from pin no. 4 of 7407 is applied to bus controller 8288 and USART 8251. Pin no. 5 and 10, READY, RESET respectively, directly connected to pin no. 22 (READY) and 21 (RESET) of 8088. To provide the power on RESET signal pin no. 11 ($\overline{RES}$) is used. The 8284 has built in RESET circuitry that uses an external R-C network to generate the power on RESET pulse. About 1ms is enough duration to satisfy the hardware RESET requirement of the 8088 and other system devices. To produce RESET pulse of 1ms we have used resistor of 100K ohms and capacitor of 10Uf. A push button is connected across the capacitor to allow manual resets.

3.4 MEMORY SECTION:

We have used partial address decoding because it is the cheapest way to generate address signals for this system. As shown in Figure 3.3 address lines A13 to A15 are used, because we want to break up the 8088's memory space into 8KB blocks of convenient space. We can conveniently make upper three address lines low for the RAM range and high for the EPROM range.

If A13 through A15 are low, the decoder IC 74LS138 will output a zero on pin no. 15 which is connected to the RAM 6264's chip enable input. Address lines A0 through A12 will directly select individual location within the 6464 RAM. Finally as per the memory mapping we will get an address range at 00000H to 01FFFH. We

used this memory mapping because the interrupt vector table must be stored in locations 00000H through 003FFH.

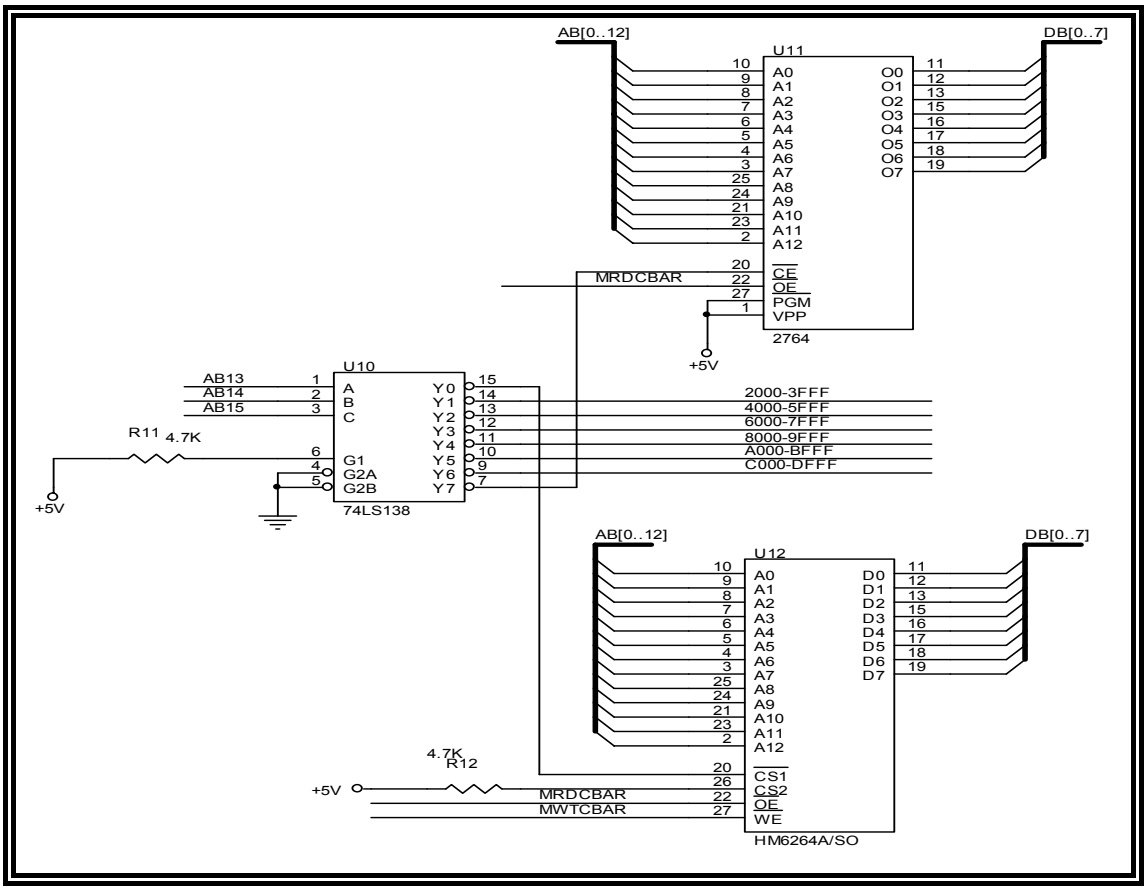


Figure 3.3 Memory circuitry for the single-board computer

If A13 through A15 are high the decoder 74LS138 will output low on pin no. 7, which is connected to the chip enable pin no. 20 of EPROM 2764. Address lines A0 to A12 maps the EPROM in to locations XE000H through XFFFFH.

Table 3.1 illustrates the address range associated with the each output of the decoder 74LS138. The free decode signals can be use for future expansion to map in the desired range. To control the data transfer between processor and memory, two signals $\overline{\text{MRDL}}$ and $\overline{\text{MWTC}}$ are generated by 8288 bus controller of the CPU section.

3.5 Input Output Section:

3.5.1 Serial Section:

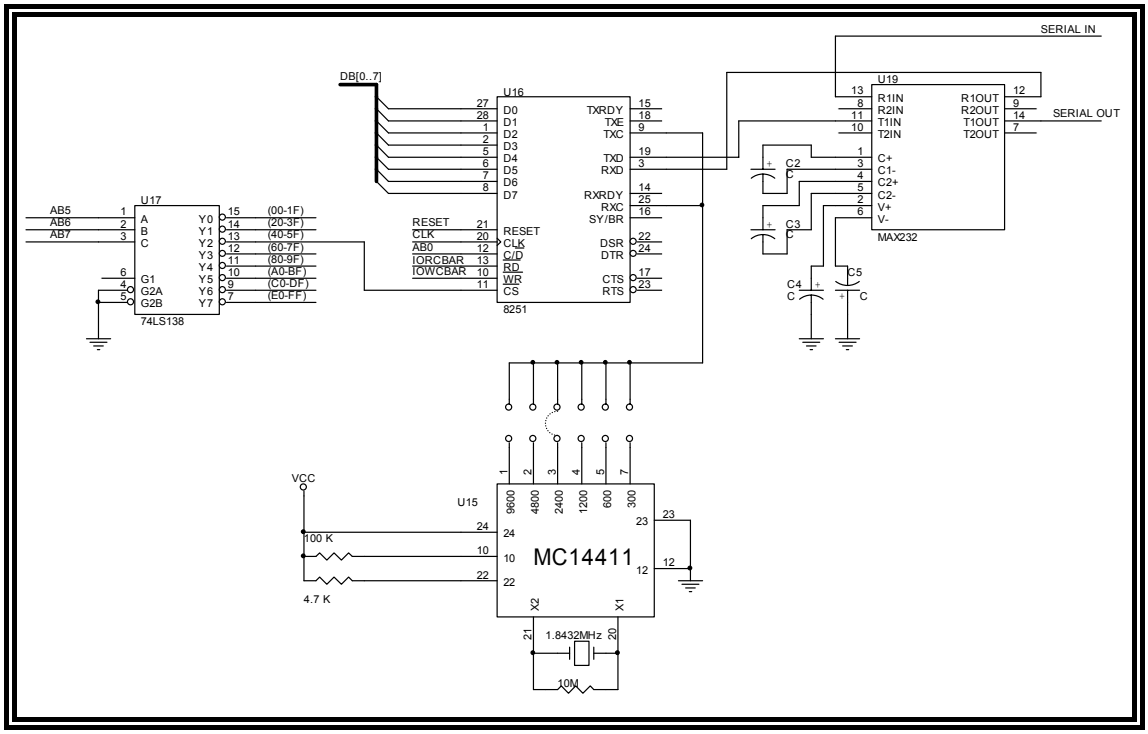


Figure 3.4 Serial I/O circuitry for minimal system

We want to connect this board with the computer's COM1 connector to transmit and receive signals. This section contains 8251 to provide serial communication, MC14411 to generate clock baud rate and MAX232CEP to convert TTL levels of 8251 into RS232 compatible voltage levels. According to design of memory mapping of this board address lines A5 through A7 generate 8 port addresses. Out of this addresses port 40H through 5Fh use to excess 8251 and other INPUT / OUTPUT devices. The remaining addresses can use for future expansion. To control READ and WRITE operations in the 8251, $\overline{\text{IORC}}$, $\overline{\text{IOWC}}$ and A0 of 8088 is connected to 8251. To initialize the 8251 at power up RESET is used. CLK (pin no. 20) is connected with pin no. 4 of 7404 in timing section. This clock is applied for internal activities of 8251.

As we communicate data of this board with PC we have to generate baud rate from 300 to 9600. To generate this standard clock frequency we used 1.8432 MHz crystal, which is connected between pin no 20 and 21 of MC14411. IC MC14411 baud rate generator outputs six different baud rates which are as follows

300, 600, 1200, 2400, 4800, 9600

Baud rate 2400 is to be selected because higher than this will be too fast to read on the screen and slower will take long time to read. For this reason we used jumper settings to select one of the range from 300 to 9600. TxD and RxD signals of 8251 are connected to pin no. 11 and 12 of MAX232CPE respectively. To create swings of ±10V on the output of the MAX232 we used four 22Uf electrolytic capacitors.

3.5.2 Parallel Section:

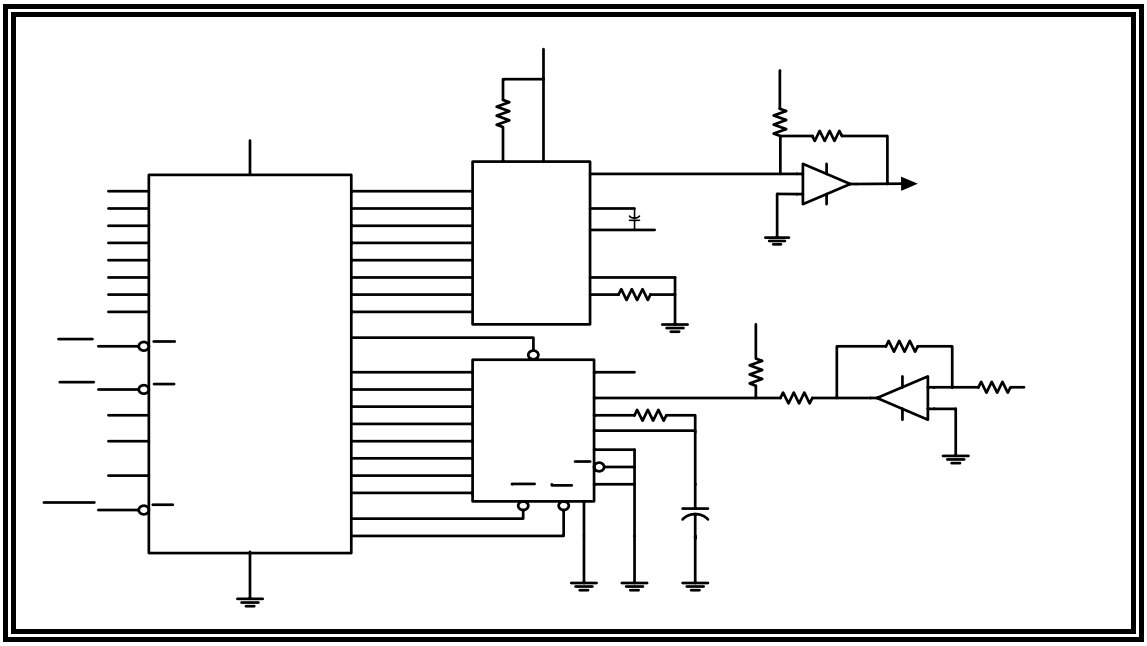


Figure 3.5 Parallel INPUT / OUTPUT circuitry for minimal system

This section is designed for single INPUT / OUTPUT port to provide parallel communication. This section contains 8255 peripheral programmable interface, 1408 DAC, 0804 ADC, and 2714 OP-Amps. As shown in figure 3.5 the data lines DB0 to DB7, INPUT / OUTPUT signals $\overline{IORC}$ and $\overline{IOWC}$, address lines AB0 and AB1 are connected to microprocessor 8088. Address lines AB0 to AB1 are used to select one of the four internal ports of 8255. We have used port addresses 00H to 03H through 74LS138 decoder to select 8255. For that pin no 15 of decoder in serial section is connected to chip select pin no. 6 of 8255. Generally, 8255 provides 24 bits of parallel INPUT / OUTPUT. This section provides analog INPUT / OUTPUT. Digital to analog converter 1408 is connected to port A of the 8255. For analog O/P from 741 we have connected pin no. 4 of 1408 to pin no. 2 of 741. 741 converts current output of the 1408 into a ±2.5V swing.

To accept analog input we have used analog to digital converter IC 0804. 0804 is connected to port B of 8255. The 0 to 5 volt swing needed by 0804, is provided by second 741 by adjusting input voltage range of ±2.5V to the 0 to 5V swing. ADC is controlled by two bits of PORT C (pin 13 and 12 of 8255). This design provides to digitise over 8000 analog samples in one second.

3.6 Schematic Diagram: Figure 3.6 to Figure 3.10 shows section wise schematic diagram of single board.

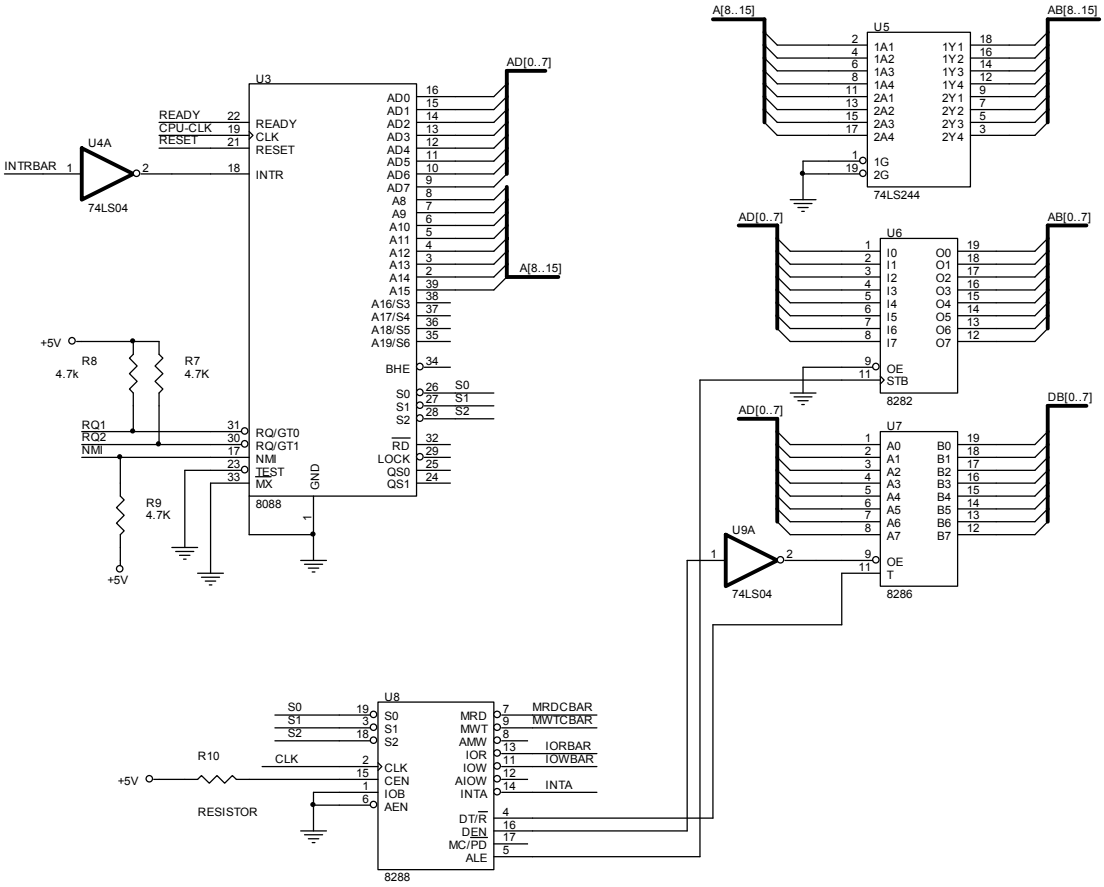


Figure 3.6 CPU SECTION

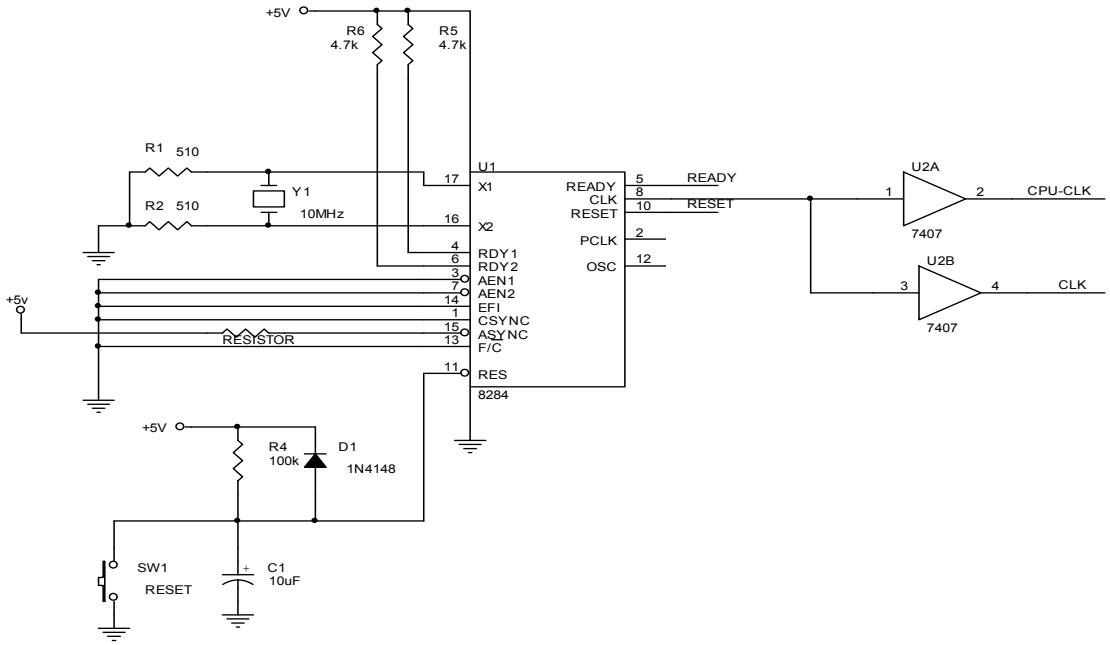


Figure 3.7 CLOCK GENERATOR SECTION

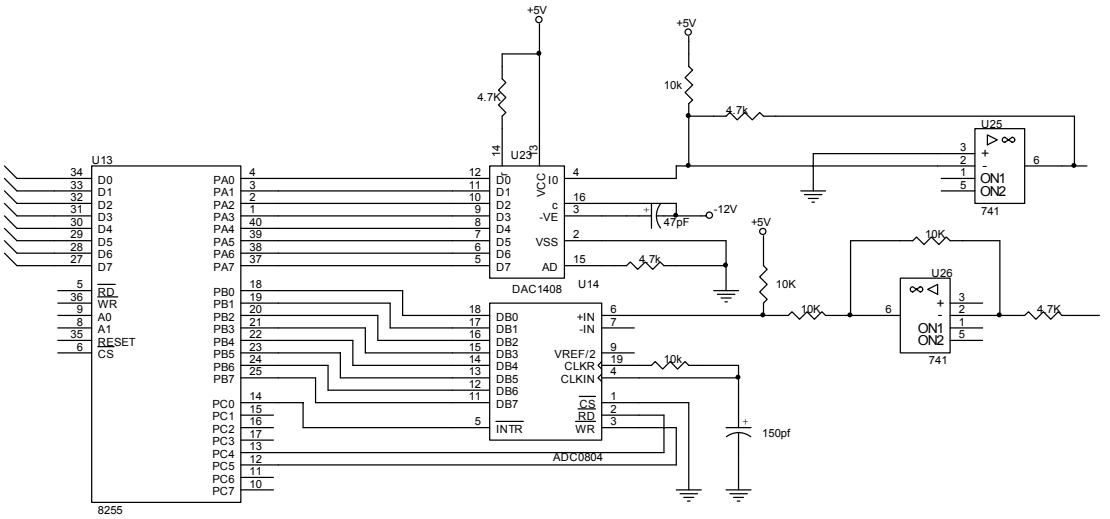


Figure 3.8 PARALLEL I/O SECTION

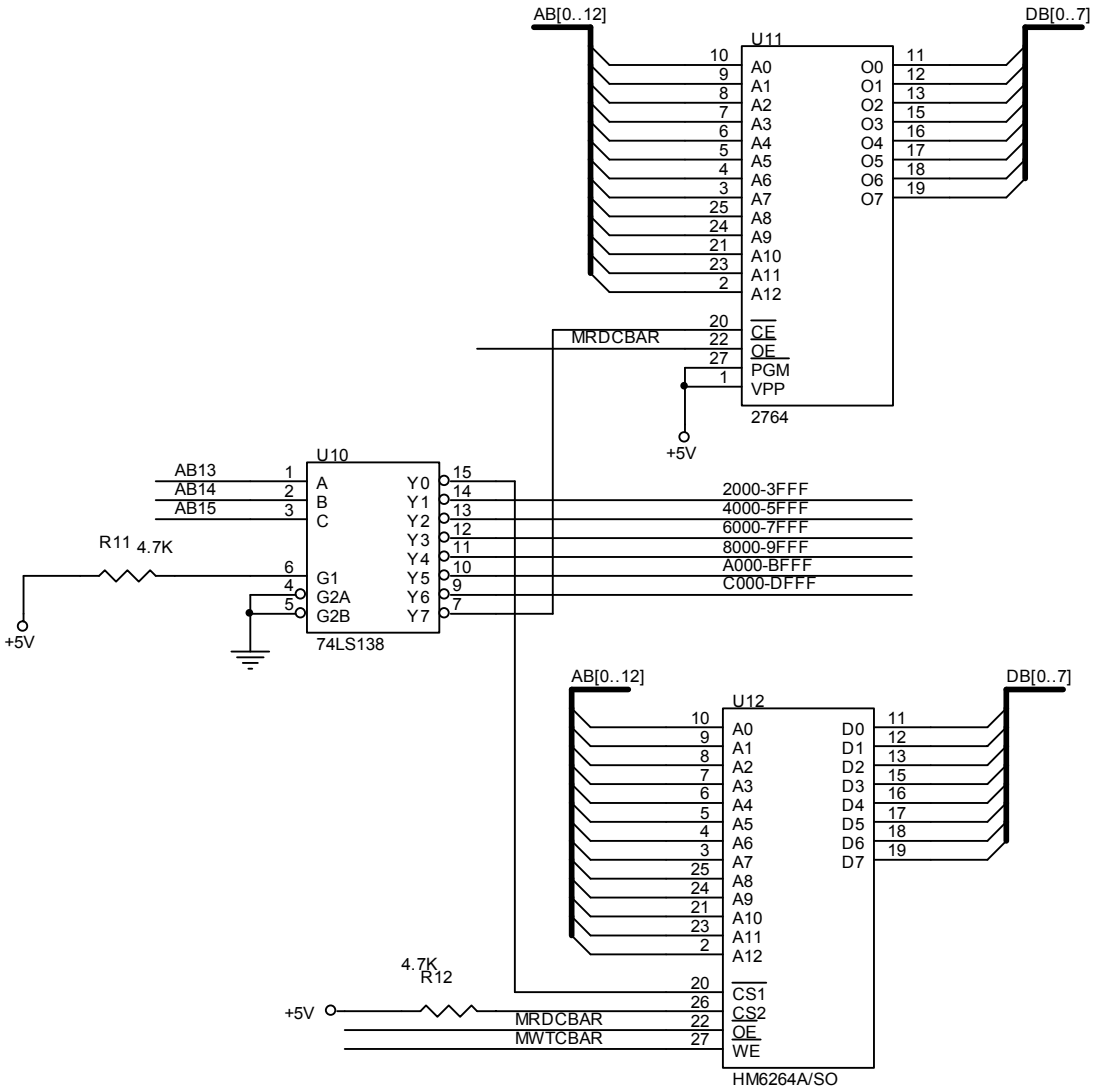


Figure 3.9 MEMORY INTERFACE SECTION

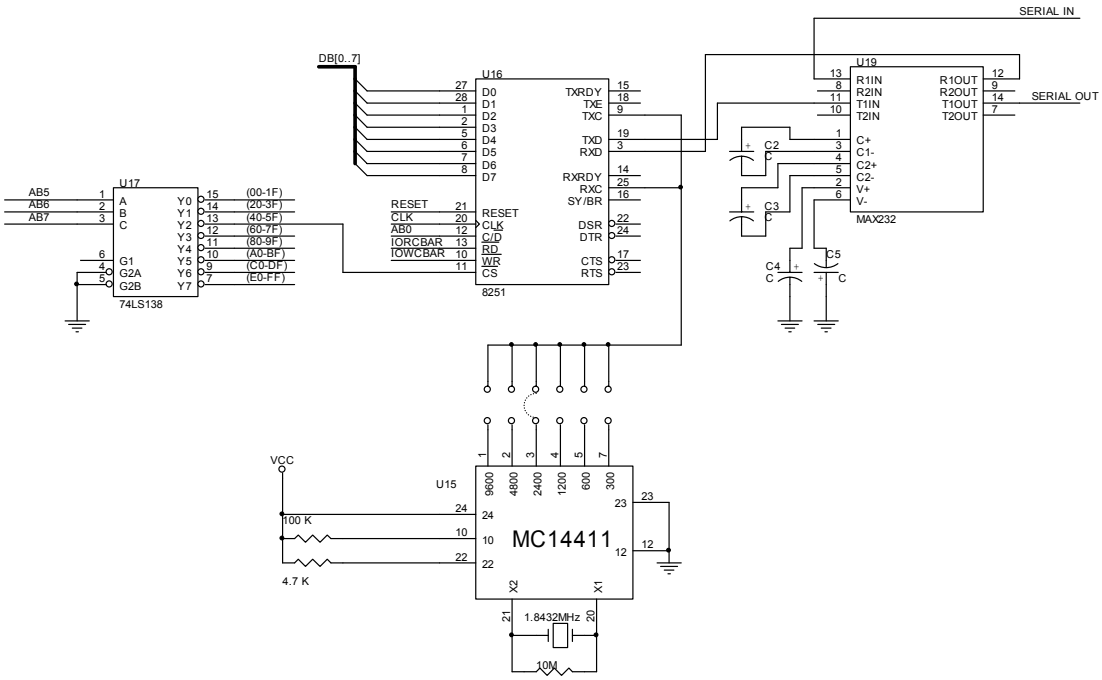
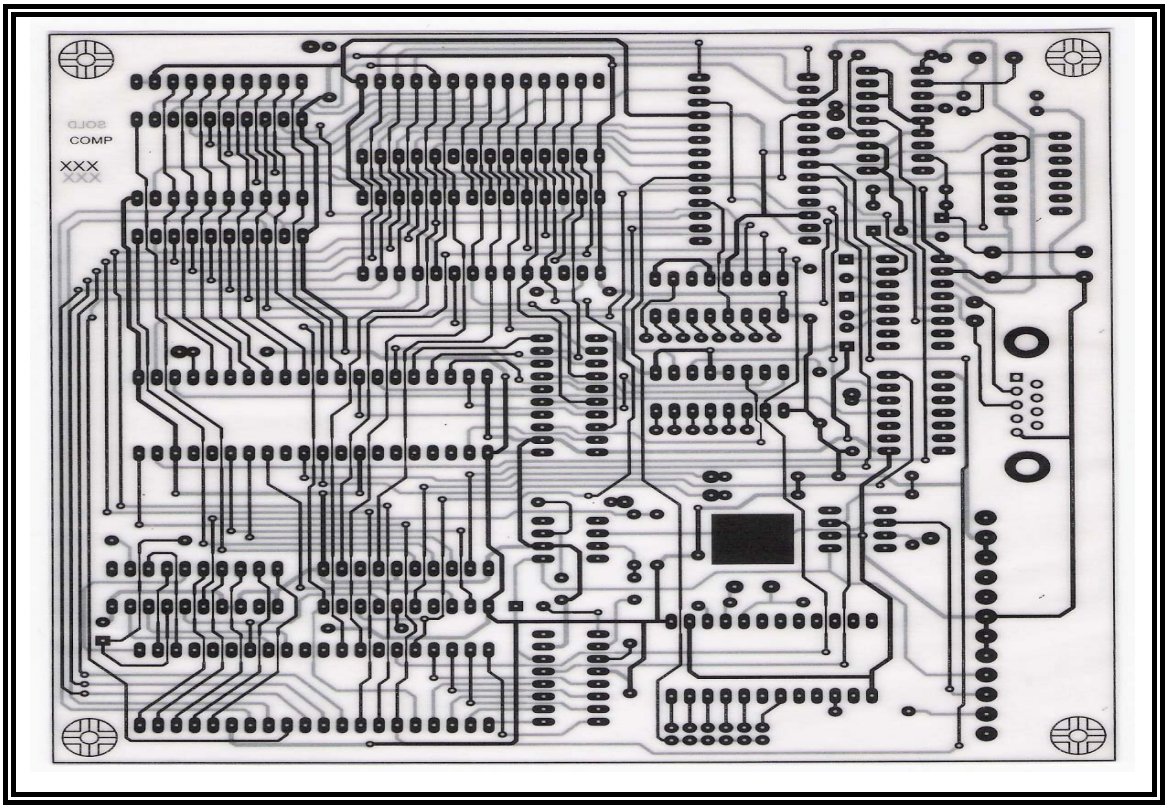
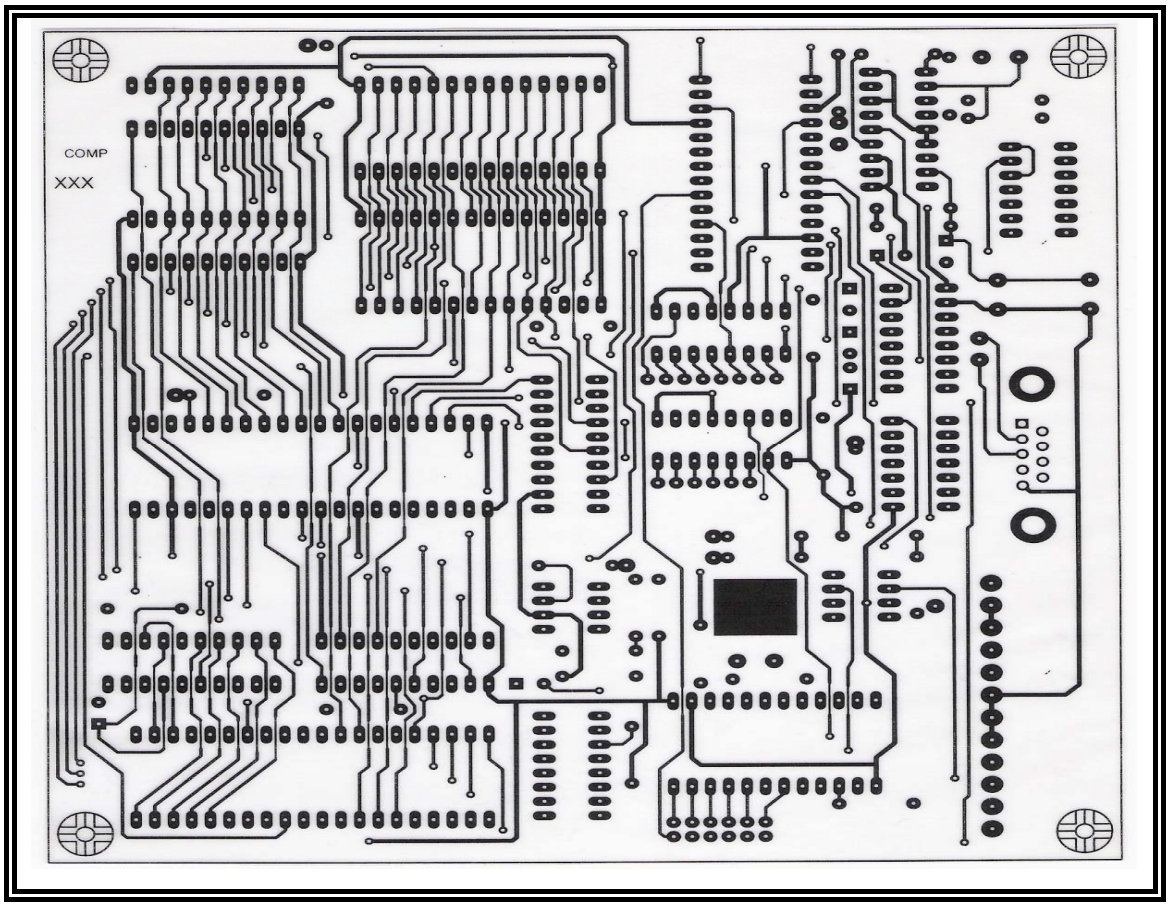
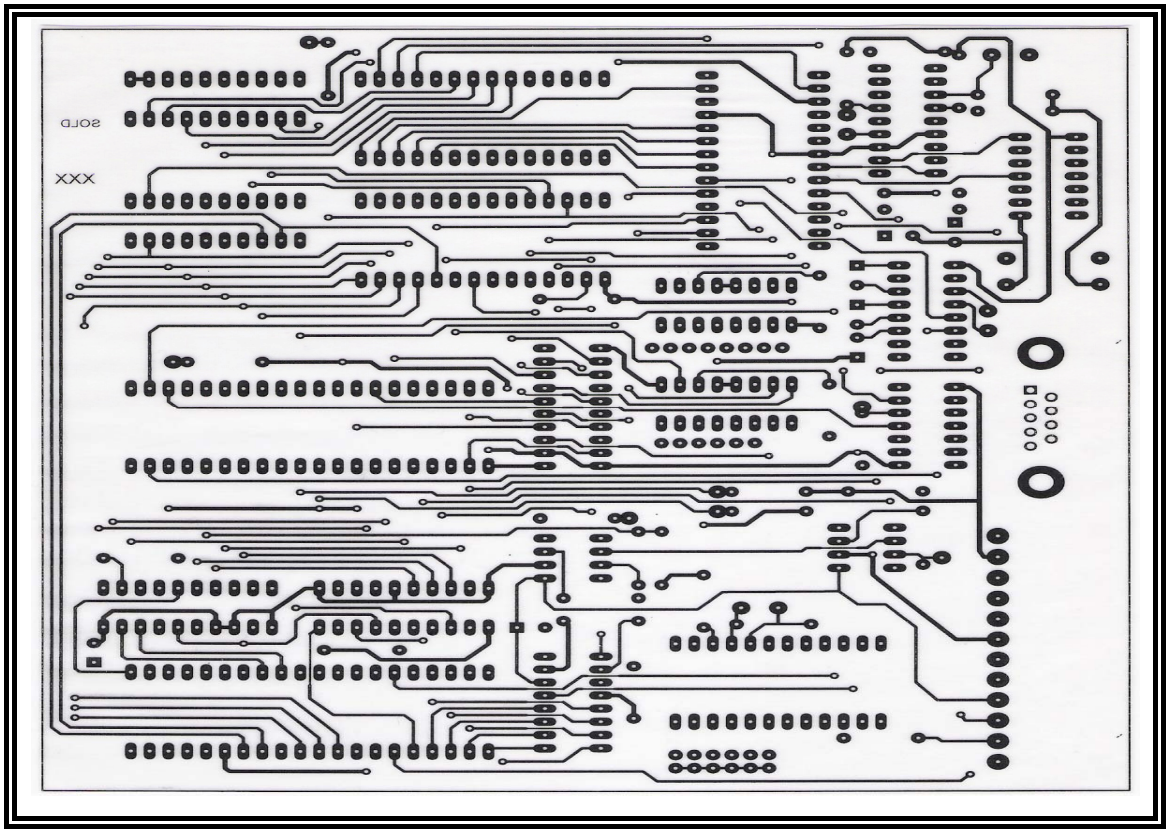


Figure 3.10 SERIAL I/O SECTION

3.7 PCB Layout: Figure 3.11 shows section wise PCB Layout of single board using 8088 processor.




COMP
XXX


SOLD
XXX

3.8 Software Description:

In section 3-2 to 3-7, I have studied, Designed and constructed the hardware part of the 8088 motherboard. But the goal is to use this system for customer programs, the monitor program must be capable of performing every step that is needed to get the new program into the memory, edit, display in the hexadecimal format and execute it. For monitor program I have used in this system has following commands.

1. B: Set break point
2. C: Clear break point
3. D: Dump memory content
4. E: Enter new register data
5. G: Go, execute the program
6. H: Help
7. I: Input data from port
8. L: Down line load a program
9. M: Move memory
10. O: Out put data to port
11. R: Display registers
12. S: Stop processor
13. T: Test analog input/output
14. X: Examine memory

To get this monitor program to start up right after power up the system, a far jump instruction into the EPROM at location corresponding to address FFFF0H.

Monitor program

```
;The following code and data form the body of
;the 8086/88 cpu software Monitor program
;
code    segment para    'code'
```

```
main    proc    far
        assume  cs:code,ds:data,es:data2
;segment usage is as follows:
;       CS for machine code
;       DS for rom-based data tables
;       ES for user ram operations
;       SS for ram-stack operations

        org     100h
;
;this is the starting address of the
;reserved system-ram area.
rtop    equ     1f80h
;-----
;This is where the re-start (int 95h) and
;breakpoint (int 03h) return vectors are
;generated.
        lea     bx,re_strt
        mov     cx,0e00h
        mov     ax,0
        mov     si,0
        mov     ds,ax
        mov     [si+254h],bx
        mov     [si+256h],cx
        lea     bx,br_entr
        mov     [si+0ch],bx
        mov     [si+0eh],cx
;-----
;Initialize all segment registers and create  the system stack. Also initialize all ;system input / output devices, then greet the user with the sign-on message and go get a
;command!
        mov     ax,cs
        add     ax,data
        mov     ds,ax
```

```
        sub     ax,ax
        mov     es,ax
        mov     ss,ax
        mov     sp,2000h
        call    in_it
        lea     si,hello
        call    s_end
        jmp     get_com
;-----
;BR_ENTR
;This is where we re-enter the monitor via
;the occurance of a breakpoint interrupt (int 03h) in the user program. The stack
;is cleaned up and the CS and IP from the caller stored for later  use. All registers
;and flags are saved and printed out along with a message showing the
;memory location where the breakpoint was encountered. In addition, the opcode
;that was previously replaced by the int 03h byte (0cch) is replaced, thus ;restoring the
original user program.

;save and display current environment
br_entr:call    envir
;
;print breakpoint message and CS:IP
        lea     si,br_bak
        call    s_end
        pop     ax
        mov     es:[br_ip+rtop],ax
        pop     ax
        mov     es:[br_cs+rtop],ax
        pop     ax
        mov     dx,es:[br_cs+rtop]
        call    h_out
        mov     al,':'
        call    c_out
        mov     dx,es:[br_ip+rtop]
```

```
        dec     dx
        call    h_out
        mov     al,']'
        call    c_out
;
;restore user code
        mov     si,es:[b_mma+rtop]
        mov     al,es:[op_kode+rtop]
        mov     es:[si],al
;
;clear breakpoint status
        mov     es:[br_stat+rtop],0
        jmp     get_com
;-----
;RE_STRT
;This is where the monitor is re-entered by the occurrence of an 'int 95h' in the
;user program. All registers and flags are saved and printed along with a message
saying from what CS:IP the user program interrupted.
;
;This routine falls into GET_COM.
re_strt:call    envir
        lea     si,renter
        call    s_end
        pop     ax
        mov     es:[br_ip+rtop],ax
        pop     ax
        mov     es:[br_cs+rtop],ax
        pop     ax
        mov     dx,es:[br_cs+rtop]
        call    h_out
        mov     al,':'
        call    c_out
        mov     dx,es:[br_ip+rtop]
        sub     dx,2
```

```
        call    h_out
        mov     al,']'
        call    c_out
;-----
;GET_COM
;This is where we get a command from the user and decide what routine address ;to
jump to for execution. Two tables are used. One contains a list of all valid
;command letters. The second contains a respective list of the addresses of the
;command routines. A sequential comparison search is made of the first table,
;comparing each element with the command letter just entered by the user.
;When a match is found, si points to the memory locations where the address for ;the
command routine is saved. The command routines are always jumped to,
;never called.
;output prompt, get a command letter,
;echo it and convert into upper-case.
get_com:mov     sp,2000h
        call    crlf
        mov     al,'>'
        call    c_out
        call    c_in
        call    c_out
        call    ch_case
;
;ignore if CR
        cmp     al,13
        jz      get_com
;
;look for command match
        mov     cx,num_com
        mov     si,0
c_test: cmp     al,coms[si]
        jz      chk_sn
        add     si,2
        loop    c_test
```

```
;
;command not found, oops
        call    error
        jmp     get_com
;
;only CR and SP are valid after the
;command letter.
chk_sn: call    c_in
        cmp     al,13
        jz      do_jmp
        call    c_out
        cmp     al,20h
        jz      do_jmp
        call    error
        jmp     get_com
;
;jump to command routine
do_jmp: jmp     j_umps[si]
;-----
;L_OAD
;This routine takes care of loading
;a standard Intel-hex file. It recog-
;nizes 4 record types:
;(00) data record
;(01) end-of-file record
;(02) extended address record
;(03) start address record
;
;init values and wait for record start
l_oad:  call    crlf
        mov     bp,400h
nxt_rec:mov     es:[c_sum+rtop],0
kolon:  call    c_in
        call    c_out
```

```
        cmp     al,':'
        jnz     kolon
;
;get record length
        call    get_byt
        mov     cl,al
        mov     ch,0
;
;get offset
        call    get_wrd
        mov     di,ax
;
;get and decode record type
        call    get_byt
        cmp     al,0
        jz      d_rec
        cmp     al,1
        jz      eof_rec
        cmp     al,2
        jz      ea_rec
        cmp     al,3
        jz      sa_rec
;
;oops, unidentified record type
        lea     si,urt
        call    s_end
        jmp     get_com
;
;read bytes into memory
d_rec:  call    get_byt
        mov     es:[bp][di],al
        inc     di
        loop    d_rec
;
```

```
;and check sum
        call    get_byt
        call    chk_sum
        jmp     nxt_rec
;
;check sum and return to command level
eof_rec:call    get_byt
        call    chk_sum
        jmp     get_com
;
;load new segment address
ea_rec: call    get_wrd
        shl     ax,1
        shl     ax,1
        shl     ax,1
        shl     ax,1
        mov     bp,ax
;
;check sum and get next record
        call    get_byt
        call    chk_sum
        jmp     nxt_rec
;
;load starting address even though
;we don't use it
sa_rec: call    get_wrd ;CS value
        call    get_wrd ;IP value
        call    get_byt
        call    chk_sum
        jmp     nxt_rec
;-----
;B_RKP
;This routine is used to get a breakpoint address from the user. The opcode byte
;at the selected location is saved and replaced  with the byte for int 03h (0cch). If
```

```
;a breakpoint is already saved the user is informed and no changes are made to
;the current breakpoint.
;get the breakpoint address
b_rkp:  call    get_num
;
;check breakpoint status
        mov     al,es:[br_stat+rtop]
        cmp     al,0
        jz      do_bp
;
;sorry, breakpoint already saved
        lea     si,bp_as
        call    s_end
        jmp     get_com
;
;save breakpoint address and user opcode
do_bp:  mov     es:[b_mma+rtop],dx
        mov     si,dx
        mov     al,es:[si]
        mov     es:[op_kode+rtop],al
;
;insert breakpoint code
        mov     al,0cch
        mov     es:[si],al
;
;adjust status and inform user that
;the breakpoint is saved
        mov     es:[br_stat+rtop],1
        lea     si,bp_sa
        call    s_end
        jmp     get_com
;-----
;C_BRP
;This routine determines if a breakpoint is currently in effect by checking the
```

```
;status byte and if so, replaces the original user code and clears the breakpoint
;status.
;
;check breakpoint status
c_brp:  mov     al,es:[br_stat+rtop]
        cmp     al,1
        jz      op_ld
;
;breakpoint not active, sorry
        lea     si,br_alc
        call    s_end
        jmp     get_com
;
;output breakpoint cleared message, restore
;user code and clear status byte.
op_ld:  lea     si,br_clr
        call    s_end
        mov     si,es:[b_mma+rtop]
        mov     al,es:[op_kode+rtop]
        mov     es:[si],al
        mov     es:[br_stat+rtop],0
        jmp     get_com
;-----
;T_EST
;This routine is used to test the analog input / output circuitry. There are two routines.
;The first repeatedly outputs sine-wave data from a rom-table to the 1408 DAC.
;The second routine echo's data read into the ADC back out to the 1408.
;
;offer a choice
t_est:  lea     si,tst_msg
        call    s_end
get_rp: call    c_in
        call    c_out
        cmp     al,'1'
```

```
        jz      waver
        cmp     al,'2'
        jz      ek_o
;
;must choose 1 or 2
        call    error
        jmp     get_rp
;
;'sine' is a rom-table containing 256 values that sequentially make up a
;normalized sine wave. There is no exit
;from the loop except for a system reset.
waver:  mov     cx,256
        mov     si,0
p_ie:   mov     al,sine[si]
        out     d_ac,al
        add     si,1
        loop    p_ie
        jmp     waver
;
;check ADC status. If active, read ADC data, complement it (a necessary
;conversion if the 1408 is to track the ADC) and send it out to the 1408. This
;routine loops forever and ever.
ek_o:   in      al,ad_stat
        and     al,01h
        jnz     ek_o
        mov     al,ad_rd
        out     ad_stat,al
        in      al,a_dc
        not     al
        out     d_ac,al
        mov     al,ad_wr
        out     ad_stat,al
        mov     al,ad_nom
        out     ad_stat,al
```

```
        jmp     ek_o
 ;-----
;I_NIT
;This routine is used to fill selected registers with custom data for use in
;a user program. The data for the 11 registers is stored in memory as 2-byte
;words starting at address r_data. When the 'go' command executes, it loads
;ax, bx, cx, dx, bp, si and di with their values and branches to the user
;program. The other registers sp, ss, es and ds are not changed because this
;may have an adverse effect on the operation of the monitor.
;
;prepare for 11 passes
i_nit:  mov     si,0
        mov     cx,11
;
;output register name
i_nex:  call    crlf
        mov     al,r_lets[si]
        call    c_out
        mov     al,r_lets[si+1]
        call    c_out
        call    blank
        mov     al,'-'
        call    c_out
        call    blank
;
;output saved register value
        mov     dx,es:r_data[rtop+si]
        call    h_out
;
;ask use for new value
        mov     al,'?'
        call    c_out
        call    get_num
;
```

```
;no change?
        cmp     bh,'0'
        jz      nun
;
;load new value into register storage
        mov     es:r_data[rtop+si],dx
;
;prepare for next pass
nun:    add     si,2
        loop    i_nex
        jmp     get_com
 ;-----
;H_ELP
;Send the system help message to the user.
;It is simply a very long data string that
;terminates with a '$'.
h_elp:  lea     si,h_msg
        call    s_end
        jmp     get_com
 ;-----
;A_UTH
;Displays the Author message. I have to get some recognition, don't I?
a_uth:  lea     si,wr_by
        call    s_end
        jmp     get_com
 ;-----
;S_TOP
;This routine puts the 8088 into the HALT state, a useful place to examine ;signals.
s_top:  lea     si,s_msg
        call    s_end
        hlt
 ;-----
;M_OVE
;This routine is used to move a block of ram around in memory. The data of
```

```
;the ram block pointed to by si is moved into memory pointed to by di.
;get starting address
m_ove:  call    get_num
        mov     si,dx
;
;get ending address
        call    get_num
;
;compute length of block
        sub     dx,si
        mov     cx,dx
        inc     cx
;
;get transfer address
        call    get_num
        mov     di,dx
;
;do the data move
mvit:   mov     al,es:[si]
        mov     es:[di],al
        inc     si
        inc     di
        loop    mvit
        jmp     get_com
;-----
;D_REG
;This routine eventually jumps to
;D_ENV within the ENVIR routine and
;displays the saved contents of the
;11 system registers.
d_reg:  call    dpr
        jmp     get_com
;-----
;E_XEC
```

```
;This routine is used to transfer execution from the monitor to the user program
;stored somewhere in the first segment of memory (where CS=0). The stored
;data for registers ax, bx, cx, dx, bp, si and di is loaded into its respective
;register. The actual jump to the user code is actually a return accomplished by
;pushing the starting address specified by the user onto the stack as a 'return'
;address.
e_xec:  call    get_num
        sub     ax,ax
        push    ax      ;CS=0
        push    dx      ;IP=user #
        call    crlf
        mov     ax,es:r_data[rtop+0]
        mov     bx,es:r_data[rtop+2]
        mov     cx,es:r_data[rtop+4]
        mov     dx,es:r_data[rtop+6]
        mov     bp,es:r_data[rtop+8]
        mov     si,es:r_data[rtop+10]
        mov     di,es:r_data[rtop+12]
        ret     ;jump to routine
;-----
;E_XAM
;This routine is used to enter/change data in the system ram. The user enters
;the starting ram address. The routine prints the address, the data stored in the
;location and then gives the option to change the data byte stored there. The
;routine exits when only a CR is entered as data.
;
;get starting address
e_xam:  call    get_num
        mov     si,dx
;
;output the address
do_lyn: call    crlf
        mov     dx,si
        call    h_out
```

```
        call    blank
        call    blank
;
;output the data
        mov     al,es:[si]
        call    htoa
;
;ask for new data
        mov     al,'?'
        call    c_out
        call    get_num
;
;no change?
        cmp     bh,'0'
        jz      e_mip
;
;replace data
        mov     es:[si],dl
ex_gna: inc     si
        jmp     do_lyn
;
;keep going or quit?
e_mip:  cmp     al,20h
        jz      ex_gna
        jmp     get_com
;-----
;D_UMP
;This routine displays the contents of a selected
;range of memory locations.
;
;get starting address
d_ump:  call    get_num
        push    dx
        call    blank
```

```
;
;get ending address
        call    get_num
        pop     si
;
;compute length of block
        sub     dx,si
        mov     cx,dx
        inc     cx
;
;go start dump
        mov     dx,si
        jmp     do_dmp
;
;check for address wrap-around (is the
;least-significant digit a '0'?)
chk_adr:mov     dx,si
        mov     al,dl
        and     al,0fh
        jnz     no_adr
;
;output address of next paragraph
do_dmp: call    crlf
        call    h_out
        call    blank
no_adr: call    blank
;
;output data byte
        mov     al,es:[si]
        call    htoa
;
;next location
        inc     si
        loop    chk_adr
```

```
        jmp     get_com
;-----
;P_IN
;This is the port-input routine. Data from the port address specified by the user is
;displayed in LCD's
;
;get port address
p_in:   call    get_num
        call    crlf
        mov     al,'['
        call    c_out
;
;read port and output data
        in      al,dx
        call    htoa
        mov     al,']'
        call    c_out
        jmp     get_com
;-----
;P_OUT
;This is the port-output routine. The user enters the port address and the byte of
;data to be output.
;
;get port address
p_out:  call    get_num
        push    dx
        mov     al,' '
        call    c_out
;
;get data byte
        call    get_num
        mov     al,dl
        pop     dx
;
```

```
;output data to port
        out     dx,al
        jmp     get_com
main    endp
;-----
;The remaining routines are all of a support nature. They are called by the main
;command routines, thus making the job of writing a 'command' a simple task
;aided by judicious calls to the following subroutines:
;
;-----
;IN_IT
;This routine initializes all system input / output devices and some status-type ram
locations.
;Destroyed: al
in_it   proc near
        mov     al,m8251
        out     s_ctrl,al
        mov     al,c8251
        out     s_ctrl,al
        mov     al,c8255
        out     ad_ctrl,al
        mov     al,80h
        out     d_ac,al
        mov     al,ad_wr
        out     ad_stat,al
        mov     al,ad_nom
        out     ad_stat,al
        mov     es:[br_stat+rtop],0
        in      al,s_data
        ret
in_it   endp
;-----
;BLANK
;Output a blank (ASCII SP) to the display.
```

```
;Destroyed: al
blank   proc near
        mov     al,20h
        call    c_out
        ret
blank   endp
;-----
;CRLF
;Output a newline sequence to the display
;(ASCII CR, LF).
;Destroyed: al
crlf    proc near
        mov     al,13
        call    c_out
        mov     al,10
        call    c_out
        ret
crlf    endp
;-----
;HTOA
;This routine performs hex-to-ASCII conversion. The data in register al is
;converted into a 2 character sequence of ASCII characters representing
;the hex pair in al. For example, if al equals 3FH then an ASCII '3' and an ASCII 'F'
are output to the
;display.
;Input: al
;Destroyed: al
htoa    proc near
        push    ax
        shr     al,1
        shr     al,1
        shr     al,1
        shr     al,1
        call    a_bias
```

```
        pop     ax
        call    a_bias
        ret
htoa    endp
;-----
;A_BIAS
;This routine converts the lower nybble of al into a printable ASCII equivalent
;character '0-9' or 'A-F'.
;Input: al
;Destroyed: al
a_bias  proc near
        and     al,0fh
        add     al,30h
        cmp     al,3ah
        jc      no_7
        add     al,7
no_7:   call    c_out
        ret
a_bias  endp
;-----
;H_OUT
;This routine prints the 4 character ASCII equivalent of the 16-bit value in dx.
;Input: dx
;Destroyed: al
h_out   proc near
        mov     al,dh
        call    htoa
        mov     al,dl
        call    htoa
        ret
h_out   endp
;-----
;C_OUT
;This routine interrogates the 8251 transmitter ready flag and sends the character
```

```
;in al to the display.
;Input: al
c_out   proc near
        push    ax
        mov     ah,al
co_s:   in      al,s_ctrl
        and     al,trdy
        jz      co_s
        mov     al,ah
        out     s_data,al
        pop     ax
        ret
c_out   endp
;-----
;CH_CASE
;This routine converts lower case 'a-z' characters in al to upper case characters.
;Input: al
;Destroyed: al
;Output: al
ch_case proc near
        cmp     al,'A'
        jc      un_alph
        and     al,0dfh
un_alph:ret
ch_case endp
;-----
;ERROR
;This routine prints out the standard monitor error message, a '?' followed
;by a terminal beep.
;Destroyed: al
error   proc near
        mov     al,'?'
        call    c_out
        mov     al,7
```

```
        call    c_out
        ret
error   endp
;-----
;C_IN
;This routine interrogates the 8251 receiver ready flag and when active
;loads register al with the received character. The most-significant bit
;is cleared.
;Destroyed: al
;Output: al
c_in    proc near
ci_s:   in      al,s_ctrl
        and     al,rrdy
        jz      ci_s
        in      al,s_data
        and     al,7fh
        ret
c_in    endp
;-----
;CHK_SUM
;This routine checks the C_SUM location  and returns an error message if it is
;not 0.
;Destroyed: si
chk_sum proc near
        cmp     es:[c_sum+rtop],0
        jz      gd_ld
        lea     si,cse
        call    s_end
        jmp     get_com
gd_ld:  ret
chk_sum endp
;-----
;GET_BYT
;This routine is used to read two successive bytes from the serial port and
```

```
;convert them back into their hex-pair equivalent.
;Destroyed: al,bl
;Output: al
get_byt proc near
        call    c_in
        call    c_out
        call    con_v
        shl     al,1
        shl     al,1
        shl     al,1
        shl     al,1
        mov     bl,al
        call    c_in
        call    c_out
        call    con_v
        or      al,bl
        add     es:[c_sum+rtop],al
        ret
get_byt endp
;-----
;GET_WRD
;This routine calls GET_BYT twice to read a word from the serial port.
;Destroyed: ax, bh
get_wrd proc near
        call    get_byt
        mov     bh,al
        call    get_byt
        mov     ah,bh
        ret
get_wrd endp
;-----
;CON_V
;This routine converts an ASCII character into its equivalent hex nibble.
;Input: al
```

```
;Destroyed: al
;Output: al
con_v   proc near
        sub     al,'0'
        cmp     al,10
        jc      no_sub7
        sub     al,7
no_sub7:ret
con_v   endp
;-----
;GET_NUM
;This routine accepts a multi-digit hex number from the user and converts them
;into a 16-bit equivalent result in dx. If more than 4 characters are entered, only
;the last 4 are used as the value. A CR or a SP will properly terminate the input
;request. If no data is entered at all, bh will contain '0' and otherwise will contain '1'.
;Destroyed: al, bh, dx
;Output: bh, dx
get_num proc near
top_n:  mov     bh,'0'
        mov     dx,0
gt_nm:  call    c_in
        cmp     al,13
        jz      gt_bye
        call    c_out
        cmp     al,20h
        jz      gt_bye
        mov     bh,'1'
        call    ch_case
        cmp     al,'0'
        jc      bad_nm
        cmp     al,'9'+1
        jc      ok_sub
        cmp     al,'A'
        jc      bad_nm
```

```
        cmp     al,'F'+1
        jnc     bad_nm
        sub     al,7
ok_sub: sub     al,30h
        shl     dx,1
        shl     dx,1
        shl     dx,1
        shl     dx,1
        add     dl,al
        jmp     gt_nm
gt_bye: clc
        ret
bad_nm: call    error
        jmp     top_n
get_num endp
;-----
;S_END
;This routine sends ASCII characters stored in memory locations pointed to by si
;to the display. The message must end with a '$'.
Input: si
Destroyed: al, si
s_end   proc near
s_nxt:  mov     al,[si]
        cmp     al,'$'
        jz      s_qwt
        call    c_out
        inc     si
        jmp     s_nxt
s_qwt:  ret
s_end   endp
;-----
;ENVIR
;This routine takes care of saving the system environment (all registers and flags)
;when the monitor is re-entered from an external source. ENVIR also re-
```

```
;establishes the monitor segment registers, which may have been altered
;by the external software. All registers are stored in memory as 2-byte words (lo-;byte
first)
;beginning at location r_data[rtop+0] (address 1F80 in system ram) in this order:
;ax, bx, cx dx, bp, si, di, sp, ds, ss, es. The flags are stored in memory locations
;1FA1-1FA2.

;This routine falls into D_ENV. Input: everything  Destroyed: ax, bx, ds, es
envir   proc near
        pushf
        push    ax
        mov     ax,ds
        push    ax
        mov     ax,data
        add     ax,0e00h
        mov     ds,ax
        mov     ax,es
        push    ax
        mov     ax,0
        mov     es,ax
        pop     ax
        mov     es:r_data[rtop+20],ax
        pop     ax
        mov     es:r_data[rtop+16],ax
        pop     ax
        mov     es:r_data[rtop+0],ax
        mov     es:r_data[rtop+2],bx
        mov     es:r_data[rtop+4],cx
        mov     es:r_data[rtop+6],dx
        mov     es:r_data[rtop+8],bp
        mov     es:r_data[rtop+10],si
        mov     es:r_data[rtop+12],di
        pop     ax
        pop     bx
```

```
        mov     es:r_data[rtop+14],sp
        push    bx
        push    ax
        mov     es:r_data[rtop+18],ss
        pop     ax
        mov     es:[f_lags+rtop],ax
;-----
;D_ENV
;This routine displays the contents of all
;cpu registers.
;
;This routine falls into the next one.
;Destroyed: ax, cx, dx, si
d_env:  mov     si,0
        mov     cx,11
        call    crlf
t_dr:   mov     al,r_lets[si]
        call    c_out
        mov     al,r_lets[si+1]
        call    c_out
        mov     al,':'
        call    c_out
        mov     dx,es:r_data[rtop+si]
        call    h_out
        call    blank
        call    blank
        add     si,2
        mov     ax,si
        and     al,7
        jnz     adjst
        call    crlf
adjst:  loop    t_dr
;-----
;D_FLG
```

```
;This routine displays the 5 most important system flags, zero, sign, parity, carry
;and aux-carry.
;Destroyed: al, cx, si

d_flg:  lea     si,fl_msg
        call    s_end
        mov     si,0
        mov     cx,5
n_flg:  mov     al,f_sym[si]
        call    c_out
        mov     al,'='
        call    c_out
        mov     ax,es:[f_lags+rtop]
        and     al,f_mask[si]
        mov     al,'0'
        jz      not_1
        inc     al
not_1:  call    c_out
        call    blank
        call    blank
        inc     si
        loop    n_flg
        ret
envir   endp
;-----
;DPR
;This routine displays the processor registers with having had to save them
;first (like ENVIR does...).
dpr     proc near
        jmp     d_env
dpr     endp
code    ends
;-----
```

```
;This data segment contains all the ROM-based data needed for proper monitor
;execution. All monitor messages are saved here.

data    segment para    'data'
hello   db      13,10,'8088 Monitor, Ver:3.0$'
wr_by   db      13,10,'DOE$'
s_msg   db      13,10,'Placing 8088 in HALT state...',13,10,'$'
renter  db      13,10,'Re-entry by external program at [$'
tst_msg db      13,10,'1) Send sinewave to D/A converter, or'
        db      13,10,'2) Echo A/D to D/A'
        db      13,10,'Choice ?$'
br_clr  db      13,10,'Breakpoint cleared.$'
bp_as   db      13,10,'Breakpoint already saved...$'
bp_sa   db      13,10,'Breakpoint saved.$'
br_bak  db      13,10,'Breakpoint encountered at [$'
br_alc  db      13,10,'Breakpoint already cleared.$'
urt     db      '<-Unidentified record type->$'
cse     db      '<-Checksum error->$'
r_lets  db      'AXBXCXDXBPSIDISPDSSSES'
fl_msg  db      13,10,'Flags: $'
f_sym   db      'SZAPC'
f_mask  db      80h,40h,10h,4,1
coms    db      'A B C D E G H I L M O R S T X '
num_com dw      15
j_umps  dw      a_uth
        dw      b_rkp
        dw      c_brp
        dw      d_ump
        dw      i_nit
        dw      e_xec
        dw      h_elp
        dw      p_in
        dw      l_oad
        dw      m_ove
```

```
        dw      p_out
        dw      d_reg
        dw      s_top
        dw      t_est
        dw      e_xam
sine    db      82h,85h,88h,8bh,8eh,91h,94h,97h
        db      9bh,9eh,0a1h,0a4h,0a7h,0aah,0adh,0afh
        db      0b2h,0b5h,0b8h,0bbh,0beh,0c0h,0c3h,0c6h
        db      0c8h,0cbh,0cdh,0d0h,0d2h,0d4h,0d7h,0d9h
        db      0dbh,0ddh,0dfh,0e1h,0e3h,0e5h,0e7h,0e9h
        db      0ebh,0ech,0eeh,0efh,0f1h,0f2h,0f4h,0f5h
        db      0f6h,0f7h,0f8h,0f9h,0fah,0fbh,0fbh,0fch
        db      0fdh,0fdh,0feh,0feh,0feh,0feh,0feh,0ffh
        db      0feh,0feh,0feh,0feh,0feh,0fdh,0fdh,0fch
        db      0fbh,0fbh,0fah,0f9h,0f8h,0f7h,0f6h,0f5h
        db      0f4h,0f2h,0f1h,0efh,0eeh,0ech,0ebh,0e9h
        db      0e7h,0e5h,0e3h,0e1h,0dfh,0ddh,0dbh,0d9h
        db      0d7h,0d4h,0d2h,0d0h,0cdh,0cbh,0c8h,0c6h
        db      0c3h,0c0h,0beh,0bbh,0b8h,0b5h,0b2h,0afh
        db      0adh,0aah,0a7h,0a4h,0a1h,9eh,9bh,97h
        db      94h,91h,8eh,8bh,88h,85h,82h,7eh
        db      7bh,78h,75h,72h,6fh,6ch,69h,66h
        db      62h,5fh,5ch,59h,56h,53h,50h,4eh
        db      4bh,48h,45h,42h,3fh,3dh,3ah,37h
        db      35h,32h,30h,2dh,2bh,29h,26h,24h
        db      22h,20h,1eh,1ch,1ah,18h,16h,14h
        db      12h,11h,0fh,0eh,0ch,0bh,9,8
        db      7,6,5,4,3,2,2,1,0,0,0,0,0,0,0,0
        db      0,0,0,0,0,0,0,1,2,2,3,4,5,6,7,8
        db      9,0bh,0ch,0eh,0fh,11h,12h,14h
        db      16h,18h,1ah,1ch,1eh,20h,22h,24h
        db      26h,29h,2bh,2dh,30h,32h,35h,37h
        db      3ah,3dh,3fh,42h,45h,48h,4bh,4eh
        db      50h,53h,56h,59h,5ch,5fh,62h,66h
```

```
        db      69h,6ch,6fh,72h,75h,78h,7bh,7fh
d_ac    equ     0
a_dc    equ     1
ad_stat equ     2
ad_ctrl equ     3
ad_rd   equ     20h
ad_wr   equ     10h
ad_nom  equ     30h
c8255   equ     83h     ;Aout, Bin, CLin, CHout
s_data  equ     40h
s_stat  equ     41h
s_ctrl  equ     41h
m8251   equ     0ceh    ;2 stop, no parity, 8 data, *16 clock
c8251   equ     05h     ;R and T enable only
trdy    equ     01h
rrdy    equ     02h
h_msg   db      13,10,'8088 Monitor Command Summary'
        db      13,10
        db      13,10,'Command     Result'
        db      13,10
        db      13,10,'A           Author identification'
        db      13,10,'B #1        set Breakpoint to #1'
        db      13,10,'C           clear Breakpoint'
        db      13,10,'D #1 #2     Dump memory from #1 to #2'
        db      13,10,'E           Enter register data'
        db      13,10,'G #1        Execute program at #1'
        db      13,10,'H           Display this message'
        db      13,10,'I #1        Input data from port #1'
        db      13,10,'L           Load program into memory'
        db      13,10,'M #1 #2 #3  Move data to #3 from range #1 to #2'
        db      13,10,'O #1 #2     Output data #2 to port #1'
        db      13,10,'R           Display registers'
        db      13,10,'S           Stop (HALT) processor'
        db      13,10,'T           Test Analog input / output'
```

```
        db      13,10,'X #1         eXamine memory starting at #1'
        db      13,10,'            <cr> to exit, <sp> no change'
        db      13,10
        db      13,10,'where #1, #2 and #3 are hex addresses'
        db      13,10,'$'
data    ends
;-----
;This data segment is used to generate the required ram-based addresses for
;temporary storage.

data2   segment para 'data'
r_data  dw      11 dup(?)
go_adr  dw      ?
br_stat db      ?
b_mma   dw      ?       ;breakpoint main-memory address
op_kode db      ?       ;opcode byte
br_cs   dw      ?       ;breakpoint code-segment
br_ip   dw      ?       ;breakpoint instruction-pointer
f_lags  dw      ?       ;system flags
c_sum   db      ?       ;checksum storage
data2   ends
        end
```

Terminal emulation program

Communicate with Single-Board Computer via COM1.

```
        .MODEL SMALL

        .DATA
XMSG    DB    'SBCIO: Press Control-X to exit...',0DH,0AH,0DH,0AH,'$'
```

```
        .CODE
        .STARTUP
        LEA   DX,XMSG     ;set up pointer to exit message
        MOV   AH,9        ;display string function
        INT   21H         ;DOS call
COM1:   MOV   DX,3FDH     ;set up status port address
MORE:   IN    AL,DX       ;read UART status
        TEST  AL,1        ;has a character been received?
        JNZ   READ        ;yes, go get it
        MOV   AH,0BH      ;check keyboard status
        INT   21H
        CMP   AL,0FFH     ;has any key been pressed?
        JNZ   COM1
        MOV   AH,7        ;read keyboard function, no echo
        INT   21H         ;DOS call
        CMP   AL,18H      ;is it Control-X?
        JZ    BYE
        PUSH  AX          ;save character
UWAIT:  IN    AL,DX       ;read UART status
        TEST  AL,20H      ;check transmitter ready bit
        JZ    UWAIT
        MOV   DX,3F8H     ;set up data port address
        POP   AX          ;get character back
        OUT   DX,AL       ;output it
        JMP   COM1
READ:   MOV   DX,3F8H     ;set up data port address
        IN    AL,DX       ;read UART receiver
        MOV   DL,AL       ;load character for output
        MOV   AH,2        ;display character function
        INT   21H         ;DOS call
        JMP   COM1
BYE:    .EXIT

        END
```

CHAPTER 4

STUDY OF NEW SEMICONDUCTOR TECHNOLOGY

4.1 Introduction

When advance microprocessors were designed, they required more and more supporting chips, which make motherboard more complicated to design. More supporting chips also effect on form factor of motherboard. Fabrication technology of semiconductor chips permitted smaller designed rules and danser packing. Over the years of development table of complex support circuits has changed but there function has remained same. Today's chip sets essentially define the personal computer.

Chipset refers to a group of integrated chips, better designed to work together, but usually marketed as a single product. In computing, the chipset refers to the specialized motherboard chips on a motherboard. Chip set often refers to the two main motherboard chips: the north bridge & south bridge. Chip set manages flow of data belt the CPU, the buses, the board disk etc. moreover, because a multitude of selected could be grown together at the same time, this integrated approach made the motherboard cost effective.

So, I have studied various Intel chipsets from technical point of view and market point of view. In this chapter I would like to give overview of chipset followed by comparison, discussion and data from research experience from Intel series chipsets.

4.2 Chipset Technology

- **FUNCTION**

The chip sets must act as a system controller, peripheral controller and memory controller.

As the basic function of a chip set is to turn a chip in to motherboard it has to provide the entire necessary support that a microprocessor needs.

The basic necessity of microprocessor is timers and oscillators, DMA controller, interrupt controller and power management.

- **SYSTEM CONTROLLER FUNCTIONS OF CHIPSET**

Circuit to work efficiently needs timers and oscillators which create the time base required for the microprocessor memory and the remaining motherboard.

. To manage the hardware interrupt, that gives priority to important functions motherboard circuit require interrupt controller. While to manage data transfers between memory and peripherals independently of the microprocessor it required DMA controller.

Finally, to manage the overall electrical use to save power and conserve battery reserves, chips set require power management function. But in recent personal computer power management functions although share software control system under the system controller, a hard disk interface to connect hard disk drives, keyboard controller to translate ports from the keyboard in to understandable from for microprocessor and it's program, sometimes for the expansion, allows independent development of the dedicated chip to revise the design of on section of the chips set without effecting the circuit of the other functions. The Pentium PRO ORION chip set put the PCI bridge controller in a separate package is good example to understand dedicated chips added by manufacturer.

Early motherboards needed additional circuit design more than a simple decoder chip, to communicate between memory and microprocessor. Today's advanced memory architectures are more complex, incorporating layers of caching, error correction and auto configuration. To handle all these complication of memory system, chip set must require memory controller function. This memory controller also acts as a cache controller for modern motherboard's external memory cache.

The capacity of RAM, type of RAM operating speed of memory modules and capacity of external cache depend on the memory controller function of chip sets.

In general core logic of chip sets are divided in to two main parts, North Bridge and South Bridge.

1. North Bridge

The Northbridge typically handles communications between the CPU, RAM, AGP port or PCI express, and the Southbridge. Some Northbridge also contain integrated video controllers. Because different processors and RAM require different signaling, a Northbridge will typically work with only one or two classes of CPUs and only one type of RAM. There are a few chipsets that support

two types of RAMs, although there are no modern examples of these that allow for the use of a different types at once. For example, the Northbridge from the NVIDIA nForce2 chipset will only work with Duron, Athlon, and Athlon XP processors combined with DDR-SDRAM, and the Intel i875 chipset will only work with Pentium 4 processors and Celeron processors that have a clock speed grater than 1.3 GHz.

The Northbridge on a particular motherboard is the most prominent factor in dictating the number of CPUs, speed and type of CPU, and the speed, type, and amount of RAM that can be used in that system. Other factors such as voltage regulation and available number of connectors also play a role. Virtually all consumers – level chipsets support only one processor, with the maximum amount of RAM varying by processor type and motherboard design. Pentium-era machines often had a limitation of 128 MB, while Pentium 4 machines have a limit of 4 GB. Since the Pentium Pro, the Intel architecture can accommodate physical addresses larger than 32 bits, typically 36 bits, which give up to 64 GB of addressing.

A Northbridge typically will only work with one or two different Southbridge ASICs; in this respect, it determines some of the other features that a given system can have by limiting which technologies are available on its Southbridge partner.

The traditional Northbridge may be dying. The memory controller, which handles communication between the CPU and RAM, has been moved onto the processor die in AMD64 processors. Other CPU designers such as Intel and IBM have considered this change for their own product lines, although it is uncertain whether either company would make such a change.

In additional, with the development of the PCI-Express bus, AGP will become obsolete. This will remove one of the traditional functions of the Northbridge for all consumer-level systems once its adoption is widespread.

An example of this change is NVIDIA's nForce3 chipset for AMD 64 systems that is a single chip. It combines all of the features of a normal Southbridge with a AGP port and connects directly to the CPU.

The Northbridge plays a important part in how far a computer can be over clocked, as its frequency is used as a baseline for the CPU to establish its own operating frequency. In today's machines, the chip is becoming increasingly hotter as computers become faster. It is not unusual for the Northbridge to now use some type of heat sink or active cooling.

2. South Bridge

The Southbridge is one of the two chips in the chipset on a PC motherboard that defines the capabilities of the motherboard. The other chip is the Northbridge. A particular Southbridge will usually work with several different Northbridges, but these two chips must be designed to work together; there is no standard for interoperability between different core logic designers.

The Southbridge chip normally defines and controls the operation of all buses and devices that are not handled by the Northbridge. This almost always includes the PCI bus, the PS/2 interface for keyboard and mouse, the serial port, the parallel port, and the floppy drive controller. Some of these functions are often handled by a secondary I/O controller, and in that case the Southbridge provides an interface to this chip.

On newer machines, it will also generally include support for the parallel ATA interface and/or serial ATA (for connecting hard drives, CD-ROMs, etc.), an Ethernet interface, USB, and IEEE 1394 (Fire wire).

Some Southbridge implement additional features such as a RAID controller or integrated audio codes (integrated sound card.)

• PERIPHERAL CONTROLLER FUNCTIONS OF CHIP SET

To create the interface between peripheral devices and microprocessor, it requires peripheral controller function. The peripheral controller include interface to establish communication between microprocessor and expansion bus, I/O port controller to give access of serial and parallel communication ports with peripherals, floppy disk interface to connect floppy disk drives.

4.3 Detail study of chipsets

Triton430FX

Introduced in early 1995, the 82430FX - to give it its full name - was Intel's first Triton chipset and conformed to the PCI 2.0 specification. It introduced support for EDO memory configurations of up to 128MB and for pipelined burst cache and synchronous cache technologies. However, it did not support a number of emerging technologies such as SDRAM and USB and was superseded in 1996 - little more than a year after its launch - by a pair of higher performance chipsets.

Triton430VX

The Triton 430VX chipset conforms to the PCI 2.1 specification, and is designed to support Intel's Universal Serial Bus (USB) and Concurrent PCI standards. With the earlier 430FX, a bus master (on the ISA or PCI bus), such as a network card or disk controller, would lock the PCI bus whenever it transferred data in order to have a clear path to memory. This interrupted other processes, and was inefficient because the bus master would never make full use of the 100 MBps bandwidth of the PCI bus. With Concurrent PCI, the chipset can wrest control of the PCI bus from an idle bus master to give other processes access on a timeshare basis. Theoretically, this should allow for data transfer rates of up to 100 MBps, 15% more than the 430FX chipset and smooth intensive PCI tasks such as video playback when bus masters are present.

The 430VX chipset was aimed fairly and squarely at the consumer market. It was intended to speed up multimedia and office applications, and it was optimized for 16-bit. Furthermore, it was designed to work with SDRAM, a special type of memory that's optimized for intensive multimedia processing. Although the performance gains are slight for this type of RAM over EDO RAM, the advantage is that it can operate efficiently from a single Dual In-line Memory Module (DIMM) and does not need to be paired.

The 430VX provided improved EDO memory timings which was supposed to allow cache less systems to be built without compromising performance, at least

compared to a PC with asynchronous cache. In practice, though, most manufacturers continued to provide at least some secondary cache, with most using synchronous cache to maximize performance.

Triton430HX

The Triton 430HX chipset is geared towards business machines and was developed with networking, video conferencing and MPEG video playback in mind. It supports multiple processors, has been optimized for 32-bit operation and to work with large memory arrays (up to 512MB) and provides error control (ECC) facilities on the fly when 32-bit parity SIMMs are used. The 430HX does not support SDRAM.

The biggest difference between the HX and VX chipsets is the packaging. Where the VX consists of four separate chips, all built using the traditional plastic quad flat packaging, the HX chipset comprises just two chips, the 82439HX System Controller (SC), which manages the host and PCI buses, and the 82371SB PIIX3 for the ISA bus and all the ports.

The SC comes in a new ball grid array (BGA) packaging which reduces overall chip size and makes it easier to incorporate onto motherboard designs. It exerts the greatest influence on the machine's CPU performance, as it manages communications between the CPU and memory. The CPU has to be fed data from the secondary cache as quickly as possible, and if the necessary data isn’t already in the cache, the SC fetches it from main memory and loads it into the cache. The SC also ensures that data written into cache by the CPU is "flushed" back into main memory.

The PIIX3 chip manages the many processes involved in getting data into and out of RAM from the other devices in the PC. It provides two EIDE channels, both of which can accept two drives. IDE drives contain most of the controlling circuitry built into the hard disk itself, so the PIIX is mainly responsible for shifting data from the drives into RAM and back as quickly as possible. It also provides two 115,200bit/s buffered serial ports, an error correcting Enhanced Parallel Port, a PS/2 mouse port and a keyboard

controller. The PIIX also supports additional connections that many motherboards have yet to adopt as the norm, such as a Universal Serial Bus connector and an infrared port.

Triton430TX

The Triton 430TX includes all the features found on the earlier chipsets, including Concurrent PCI, USB support, aggressive EDO RAM timings and SDRAM support and is optimized for MMX processors and is designed to be used in both desktop and mobile computers.

The Triton 430TX also continues the high-integration two-chip BGA packaging first seen with the 430HX chipset, comprising the 82439TX System Controller (MTXC) and the 82371AB PCI ISA IDE Xcelerator (PIIX4). The former integrates the cache and main memory DRAM control functions and provides bus control to transfers between the CPU, cache, main memory, and the PCI Bus. The latter is a multi-function PCI device implementing a PCI-to-ISA bridge function, a PCI IDE function, a Universal Serial Bus host/hub function, and an Enhanced Power Management function.

The diagram below provides an overview of the overall architecture and shows the division of functionality between the System Controller and the Peripheral Bus Controller components - which are often referred to as "Northbridge" and "Southbridge" chipsets respectively.

The TX incorporates the Dynamic Power Management Architecture (DPMA) which reduces overall system power consumption and offers intelligent power-saving features like suspend to RAM and suspend to disk. The TX chipset also supports the new Ultra DMA disk protocol which enables a data throughput of 33 MBps from the hard disk drive to enhance performance in the most demanding applications.

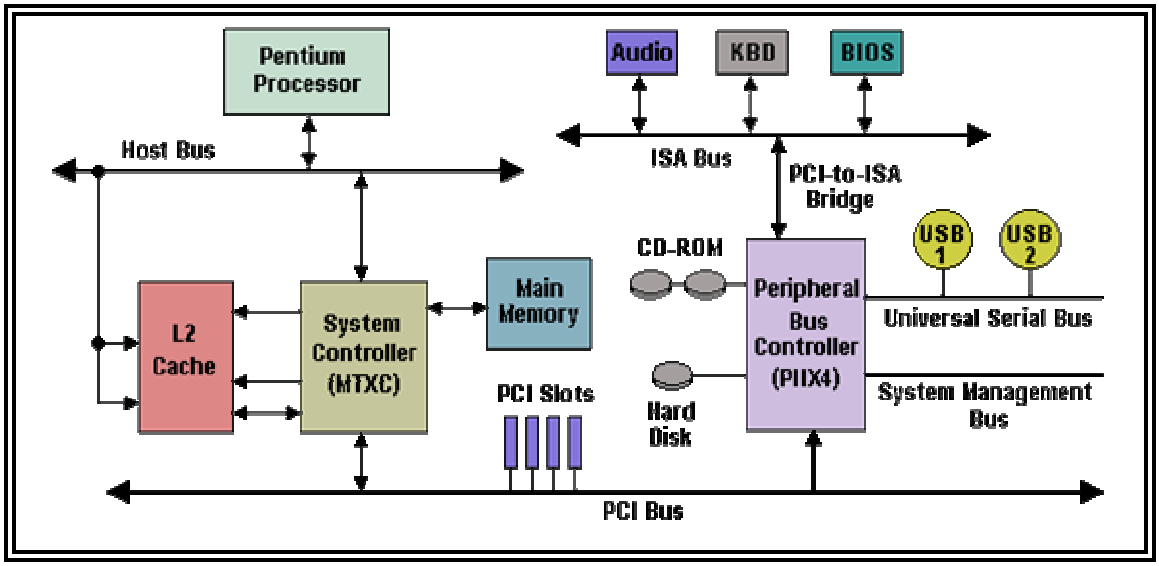


Figure 4.1 BLOCK DIAGRAM OF Triton430TX CHIP SET

440LX

The 440LX (by this time Intel had dropped the term "Triton") was the successor to the Pentium Pro 440FX chipset and was developed by Intel to consolidate on the critical success of the Pentium II processor launched a few months earlier. The most important feature of the 440LX is support for the Accelerated Graphics Port (AGP), a new, fast, dedicated bus designed to eliminate bottlenecks between the CPU, graphics controller and system memory, which will aid fast, high-quality 3D graphics.

Other improvements with the LX are more like housekeeping, bringing the Pentium II chipset up to the feature set of the 430TX by providing support for SDRAM and Ultra DMA IDE channels. The chipset includes the Advanced Configuration and Power Interface (ACPI), allowing quick power down and up, remote start-up over a LAN for remote network management, plus temperature and fan speed sensors. The chipset also has better integration with the capabilities of the Pentium II, such as support for dynamic execution and processor pipelining.

440EX

The 440EX AGPset, based on the core technology of the 440LX AGPset, is designed for use with the Celeron family of processors. It is ACPI-compliant and extends support for a number of advanced features such as AGP, UltraDMA/33, USB and 66MHz SDRAM, to the "Basic PC" market segment.

440BX

The PC's system bus had been a bottleneck for too long. Manufacturers of alternative motherboard chipsets had made the first move, pushing Socket 7 chipsets beyond Intel's 66MHz. Intel's response came in April 1998, with the release of its 440BX chipset, which represented a major step in the Pentium II architecture. The principal advantage of the 440BX chipset is support for a 100MHz system bus and 100MHz SDRAM. The former 66MHz bus speed is supported, allowing the BX chipset to be used with older (233MHz-333MHz) Pentium IIs.

The 440BX chipset features Intel's Quad Port Acceleration (QPA) to improve bandwidth between the Pentium II processor, the Accelerated Graphics Port, 100-MHz SDRAM and the PCI bus. QPA combines enhanced bus arbitration, deeper buffers, open-page memory architecture and ECC memory control to improve system performance. Other features include support for dual processors, 2x AGP, and the Advanced Configuration Interface (ACPI).

440ZX

The 440ZX is designed for lower cost form factors without sacrificing the performance expected from an AGPset, enabling 100MHz performance in form factors like microATX. With footprint compatibility with the 440BX, the 440ZX is intended to allow OEMs to leverage BX design and validation investment to produce new systems to meet entry level market segment needs.

440GX

Released at the same time as the Pentium II Xeon processor in mid-1998, the 440GX chipset was an evolution of the 440BX AGPset intended for use with Xeon-based workstations and servers. Built around the core architecture of its 440BX predecessor, the 440GX includes support for both Slot 1 and Slot 2 implementations, a 2x AGP expansion slot, dual CPUs and a maximum of 2GB of memory.

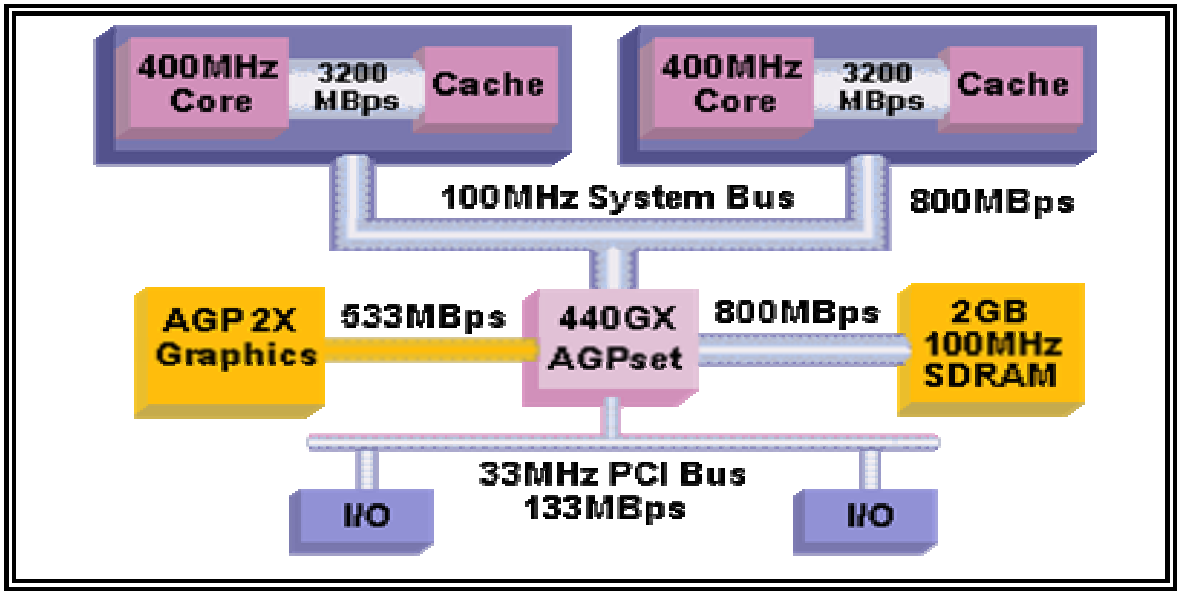


FIGURE 4.2 BLOCK DIAGRAM OF 440GX CHIPSET

Importantly, the chipset supports full speed backside bus operation, enabling the Pentium II Xeon's Level 2 cache to run at the same speed as the core of the CPU.

810 AGPset

Formerly codenamed "Whitney", the 810 AGPset finally reached the market in the summer of 1999. It is a three-chip solution comprising the 82810 Graphics Memory Controller Hub (GMCH), 82801 I/O Controller Hub (ICH) and 82802 Firmware Hub (FWH) for storing the system and video BIOS. A break from tradition is that these components don't communicate with each other over the PCI bus. Instead, they use a dedicated 8-bit 266 MBps proprietary bus, thereby taking load off the PCI subsystem.

The SDRAM memory interface is also unusual in that it runs at 100MHz irrespective of the system bus speed. There's no ISA support, but it could be implemented if a vendor added an extra bridge chip.

At the time of its launch, there were two versions of the 810 - the 82810 and 81810-DC100. The former is 66MHz part with no graphics memory, while the latter is a 100MHz-capable chip with support for 4MB of on-board graphics memory. The Direct AGP graphics architecture uses 11MB of system memory for frame buffer, textures and Z-buffer if no display cache is implemented. This drops to 7MB if the display cache *is* implemented. The whole configuration is known as Direct Video Memory technology. Also incorporated in the chipset is an AC-97 CODEC, which allows software modem and audio functionality. Vendors can link this to an Audio Modem Riser (AMR) slot to facilitate future plug-in audio or modem upgrades.

In the autumn of 1999 a subsequent version of the chipset - the 810E - extended support processors with a 133 MHz system bus. The Intel 810E chipset features a unique internal gear arbitration, allowing it to run seamlessly with 66 MHz, 100 MHz and 133 MHz processor busses.

As the cost of processors come down, the marginal costs of the motherboard, graphics and sound subsystems becomes an increasingly important factor in vendors' efforts to hit ever-lower price points. However, high levels of integration can be a double-edged sword: it reduces vendors' bill-of-materials (BOM) costs, but also limits their capability for product differentiation. Many manufacturers defer their decisions on graphics and sound options to late in the production cycle in order to maintain a competitive marketing advantage. Given that other highly integrated solutions - such as Cyrix's Media GX - haven't fared particularly well in the past, the 810 AGPset represents a bold move on Intel's part and one that signals the company's determination to capture a greater share of the "value PC" market which had been effectively ceded to AMD and Cyrix over the prior couple of years.

820 AGPset

Originally scheduled to be available concurrently with the Pentium III processor in the spring of 1999, Intel's much delayed 820 chipset was finally launched in November that year. Those delays - which had left Intel in the position not having a chipset that supported the 133MHz system bus speed their latest range of processors were capable of - were largely due to delays in the production of Direct Rambus DRAM (DRDRAM), a key component in Intel's 133MHz platform strategy.

Direct RDRAM memory provides a memory bandwidth capable of delivering 1.6 GBps of maximum theoretical memory bandwidth - twice the peak memory bandwidth of 100MHz SDRAM systems. Additionally, the 820's support for AGP 4x technology allows graphics controllers to access main memory at more than 1 GBps - twice that of previous AGP platforms. The net result is the significantly improved graphics and multimedia handling performance expected to be necessary to accommodate future advances in both software and hardware technology.

The 820 chipset employs the Accelerated Hub Architecture that is offered in all Intel 800 series chipsets - the first chipset architecture to move away from the traditional Northbridge /Southbridge design. It supports a bandwidth of 266 MBps and, with it's optimized arbitration rules which allow more functions to run concurrently, delivers significantly improved audio and video handling. The chipset's three primary components are:

- Memory Controller Hub
- I/O Controller Hub, and
- Firmware Hub.

The Memory Controller Hub provides a high-performance interface for the CPU, memory and AGP and supports up to 1GB of memory via a single channel of RDRAM using 64-, 128- and 256-Mbit technology. With an internal bus running at 1.6 GBps and an advanced buffering and queuing structure, the Memory Hub Controller balances

system resources and enables concurrent processing in either single or dual processor configurations.

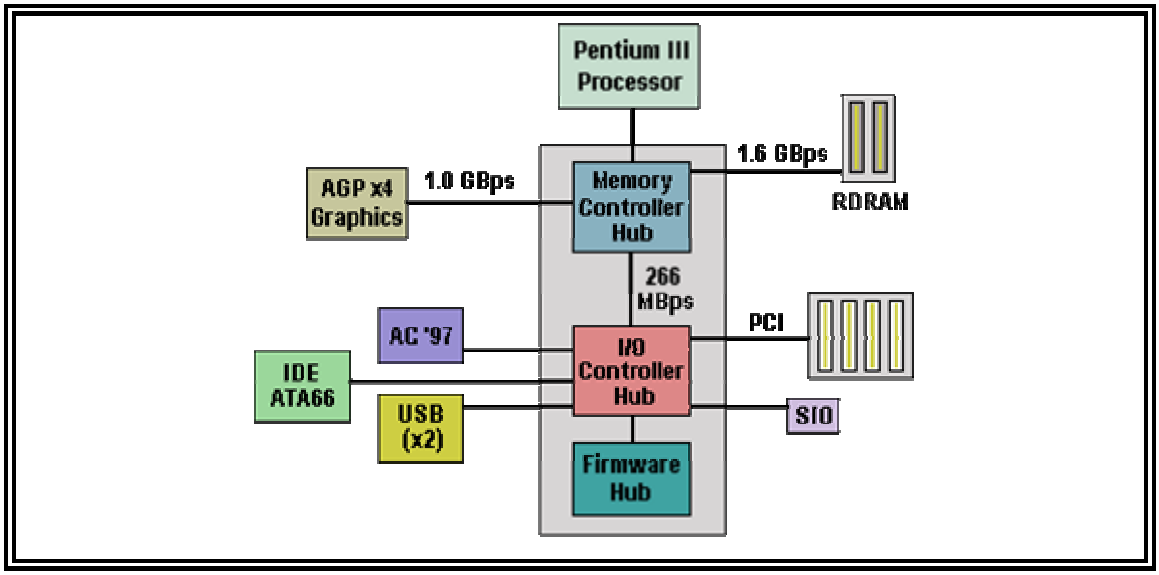


FIGURE 4.3 BLOCK DIAGRAM OF 820 AGP CHIPSET

The I/O Controller Hub forms a direct connection from the PC's I/O devices to the main memory. This results in increased bandwidth and significantly reduced arbitration overhead, creating a faster path to main memory. To capitalize further on this faster path to main memory, the 820 chipset features an integrated AC97 controller in addition to an ATA66 drive controller, dual USB ports and PCI add-in cards.

The Firmware Hub stores system and video BIOS and includes a first for the PC platform - a hardware-based random number generator. The Intel RNG provides truly random numbers through the use of thermal noise - thereby enabling stronger encryption, digital signing and security protocols. This is expected to be of particular benefit to the emerging class of e-commerce applications.

The i820 hadn't long been on the market before Intel - recognizing that the price of RDRAM was likely to remain high for sometime - designed and released an add-on chip, the 82805 Memory Translator Hub (MTH), which, when implemented on the motherboard, allowed the use of PC100 SDRAM. Sitting between the i820's Memory

Controller Hub (MCH) and the RDRAM memory slots, the MTH chip translates the Rambus memory protocol that's used by RDRAM into the parallel protocol required by SDRAM, thereby allowing the i820 to use this much more price attractive memory.

Within a few months, a bug in the MTH component came to light. This was serious enough to cause Intel to recall all MTH-equipped i820-motherboards. Since it wasn't possible to replace the defective chip Intel took the extraordinary step of giving every owner of an MTH-equipped i820 motherboard a replacement non-MTH motherboard *as well as* RDRAM to replace the SDRAM that was used before!

815

The various problems that had so delayed the introduction of Direct Rambus DRAM (DRDRAM), finally resulted in Intel doing what it had been so reluctant to do for so long - release a chipset supporting PC133 SDRAM. In fact, in mid-2000, it announced two such chipsets - formerly codenamed "Solano" - the 815 Chipset and the 815E Chipset.

Both chipsets use Intel's Graphics and Memory Controller Hub (GMCH). This supports both PC133 and PC100 SDRAM and provides onboard graphics, with a 230MHz RAMDAC and limited 3D acceleration. This gives system integrators the option of using the on-board graphics - and system memory - for lower cost systems or upgrading via an external graphics card for either AGP 4x or AGP 2x graphics capabilities.

Additionally, and like the 820E Chipset before it, the 815E features a new I/O Controller Hub (ICH2) for greater system performance and flexibility. This provides an additional USB controller, a Local Area Network (LAN) Connect Interface, dual Ultra ATA /100 controllers and up to six-channel audio capabilities. Integrating a Fast Ethernet controller directly into the chipsets makes it easier for computer manufacturers and system integrators to implement cost-effective network connections into PCs. The ICH2's enhanced AC97 interface supports full surround-sound for Dolby Digital audio found on DVD and simultaneously supports a soft modem connection.

850

Designed in tandem with the Pentium 4 processor, Intel's 850 Chipset represents the next step in the evolution of the Intel Hub Architecture, the successor to the previous Northbridge /Southbridge technology first seen on the 810 Chipset. Comprising the 82850 Memory Controller Hub (MCH) and 82801BA I/O Controller Hub (ICH2), the new chipset's principal features are:

- a 400MHz system bus
- dual RDRAM memory channels, operating in lock step to deliver 3.2 GBps of memory bandwidth to the processor
- support for 1.5V AGP4x technology, allowing graphics controllers to access main memory at over 1 GBps - twice the speed of previous AGP platforms
- two USB controllers, doubling the bandwidth available for USB peripherals to 24 MBps over four ports
- dual Ultra ATA/100 controllers support the fastest IDE interface for transfers to storage devices.

To ensure maximum performance, the system bus is balanced with the dual RDRAM channels at 3.2 GBps, providing 3x the bandwidth of platforms based on Intel III processors and allowing better concurrency for media-rich applications and multitasking.

In the autumn of 2002, some 18 months after the i850 was first introduced, the i850E variant was released, extending the capabilities of the chipset to support Hyper-Threading, a 533MHz system bus and PC1066 memory, for Pentium 4 class processors.

I845

The fact that system builders were obliged to use expensive DRDRAM - by virtue of the absence of any Pentium 4 chipsets supporting conventional SDRAM - had been an issue ever since the Pentium 4's launch at the end of 2000. The situation changed during the course of 2001, with chipmakers SiS and VIA both releasing Pentium 4 chipsets with DDR SDRAM support. Although this was a move of which Intel disapproved, it did have

the effect of boosting the appeal of the Pentium 4, whose sales hitherto had been disappointing.

In the summer of 2001 Intel eventually gave in to market pressures and released their 845 chipset - previously codenamed "Brookdale" - supporting Pentium 4 systems' use of PC133 SDRAM. Whilst the combination of i845 and PC133 SDRAM meant lower prices - given that the speed of the memory bus was about three times slower than that of the Pentium 4 system bus - it also meant significantly poorer performance than that of an i850/DRDRAM based system. The reason the i845 didn't support faster DDR SDRAM at this time was apparently because they were prevented from allowing this until the start of the following year by the terms of a contract they'd entered into with Rambus, the inventors of DRDRAM.

Sure enough, at the beginning of 2002 re-released of the i845 chipset. The new version - sometimes being referred to as i845D - differs from its predecessor only in respect of its memory controller, which now supports PC1600 and PC2100 SDRAM - sometimes referred to as DDR200 and DDR266 respectively - in addition to PC133 SDRAM. It had reportedly been Intel's original intention for the i845 chipset to support only DDR200 SDRAM - capable of providing a maximum bandwidth of 1600MBps. However, the boom in the use of DDR SDRAM - and the consequent dramatic fall in prices - caused a rethink and the subsequent decision to extend support to DDR266 (maximum bandwidth 2100MBps). The fact that the company was prepared to make this decision even though it was bound to adversely impact the market share of its i850 chipset appears to indicate that the company's apparent infatuation with DRDRAM is well and truly over.

The 400MHz system bus of the i845 solution enables up to 3.2GBps of memory bandwidth to Pentium 4 processor. Compare this with the up to 1 GBps of data transfer possible from PC133 SDRAM and it is clear why faster DDR SDRAM makes such a difference to overall system performance. Its 1.5V 4x AGP interface with provides over 1 GBps of graphics bandwidth. Other features of the i845 chipset include an 4x AGP

interface, 133MBps to the PCI, support for four USB ports, six-channel audio, a generally unused LAN connect interface, dual ATA-100 controllers and CNR support.

The i845 is Intel's first chipset to use a Flip Chip BGA packaging for the chip itself. This improves heat conductivity between the Memory & Controller Hub (MCH) and its heat sink which is required for proper operation. It is also the first MCH built using a 0.18-micron process; earlier versions have been 0.25-micron. The smaller die allows another first - the incorporation of a Level 3-like write cache, significantly increasing the speed at which the CPU is able to write data. It is expected that the transition to 0.13-micron MCH/Northbridges will enable this idea to be further developed, to the point where chipsets include much larger, genuine Level 3 caches on the MCH itself. The i845 further capitalizes on the performance advantage realized by its high-speed write cache by the provision of deep data buffers. These play an important role in assisting the CPU and write cache to sustain its high data throughput levels.

A number of newer versions of the i845 chipset were subsequently released, all supporting the USB 2.0 interface (which increases bandwidth up to 40 times over the previous USB 1.1 standard):

- The i845G chipset, incorporating a new generation of integrated graphics - dubbed Intel Extreme Graphics - and targeted at the high-volume business and consumer desktop market segments.
- The i845E chipset, which works with discrete graphics components
- The i845GL chipset, designed for Celeron processor-based PCs.

i845GE chipset

The i845GE chipset was designed and optimized to support Hyper-Threading, Intel's innovative technology that achieves significant performance gains by allowing a single processor to be treated as two logical processors. Whilst not the first i845 chipset to support HT technology, it was the first in which that support was actually implemented, being launched at the same time as the first Intel's first HT-enabled desktop CPU, the 3.06GHz Pentium 4 unveiled in late 2002.

As well as supporting a faster, 266MHz version of Intel's Extreme Graphics core, the i845GE also supports a system bus speed of either 400 or 533MHz, up to DDR333 main memory and offers maximum display (digital CRT or TV) flexibility through an AGP4x connector.

The i845PE and i845GV chipsets are lower-spec variants of the i845GE, the former having no integrated graphics and the latter limiting both the Intel Extreme Graphics core and main memory support to DDR266 SDRAM.

875P chipset

Originally, Intel had planned to introduce a 800MHz FSB in the context of the Prescott, the upcoming 90nm Pentium 4 core. However, in the event this was brought forward to the spring of 2003. The rationale was to extend the Pentium 4's performance curve within the confines of their current 0.13-micron process, without having to increase clock speeds to unsustainable levels. The transition from 533MHz to 800MHz FSB was aided and abetted by an associated new chipset platform, the 875P chipset, formerly codenamed Canterwood.

A 64-bit 800MHz FSB provides 6.4GBps of bandwidth between the Memory Controller Hub (or Northbridge) and the CPU. In a move that appears to further reduce the strategic importance of DRDRAM in Intel's product planning, and that had been signaled by the earlier E7205 chipset, the memory subsystem the 875P uses to balance bandwidth between the Memory Controller Hub (MCH) and memory banks is dual channel DDR SDRAM, all of the DDR400, DDR333 and DD266 variants.

Currently, there are two different strategies being employed in dual-channel memory controllers, one in which where each memory bank has its own memory channel and an arbiter distributes the load between them and the other to actually create a wider memory channel, thereby "doubling up" on standard DDR's 64-bit data paths. The i875P employs the latter technique, with each pair of installed DIMMs acting as a 128-bit memory module, able to transfer twice as much data as a single-channel solution, without the need for an arbiter.

As a consequence, dual channel operation is dependent on a number of conditions being met, Intel specifying that motherboards should default to single-channel mode in the event of any of these being violated:

- DIMMs must be installed in pairs
- Both DIMMs must use the same density memory chips
- Both DIMMs must use the same DRAM bus width
- Both DIMMs must be either single-sided or dual-sided.

The 875P chipset also introduces two significant platform innovations:

- Intel Performance Acceleration Technology (PAT), and
- Communications Streaming Architecture (CSA).

PAT optimizes memory access between the processor and system memory for platforms configured with both the new 800Mhz FSB and Dual-Channel DDR400 memory. CSA is a new communications architecture that creates a dedicated link from the Memory Controller Hub (MCH) to the network interface, thereby offloading network traffic from the PCI bus. Used in conjunction with the new Intel PRO/1000 CT Desktop Connection gigabit Ethernet controller, it is claimed that CSA doubles the networking bandwidth possible with traditional PCI bus-based solutions.

Additionally, the 875P chipset includes a high-performance AGP 8x graphics interface, integrated Hi-Speed USB 2.0, optional ECC is supported for users that demand memory data reliability and integrity and dual independent DMA audio engines, enabling a user to make a PC phone call whilst at the same time playing digital music streams. The chipset is also Intel's first to offer native Serial ATA (SATA), a special version designated by the "-R" suffix adding RAID - albeit only RAID 0 (data striping) - support.

865 Chipset

If the i875 chipset can be viewed as the logical successor to i850E, then its mainstream variant, the i865 chipset - formerly codenamed Springdale - can be viewed as

the logical successor to the i845 series of chipsets. Not only do the i875/i865 chipsets represent a huge technological leap compared to their predecessors, but the performance gap between the pair of recent chipsets is significantly less than it was between the i850E and i845 family.

There is a clear trend in PC hardware towards parallel processes, epitomized by Intel's Hyper-Threading technology. However, there are other examples of where performing several tasks at the same time is preferable to carrying out a single task quickly. Hence the increasing popularity of small RAID arrays and now the trend towards dual-channel memory subsystems.

Currently, there are two different strategies being employed in dual-channel memory controllers, one in which where each memory bank has its own memory channel and an arbiter distributes the load between them and the other to actually create a wider memory channel, thereby "doubling up" on standard DDR's 64-bit data paths. In common with the i875P chipset, the i865's Memory Controller Hub employs the latter, the same conditions for dual-channel operation also applying.

The i865 memory controller is the same as that used by the i875P chipset, supporting:

- Hyper Threading
- Dual 64-bit DDR memory channels
- Communication Streaming Architecture bus for gigabit Ethernet

and capable of being paired with either the ICH5 or ICH5R chip - which handles things like the 10/100 Ethernet interface, 6-channel AC97 audio interface, USB 2.0, the PCI bus, etc., to provide the following additional features:

- 8 USB 2.0 ports
- Dual independent Serial ATA ports

The ICH5R also provides software RAID for Serial ATA drives.

The upshot is that - unlike the i875P - i865 chipsets are available in three different versions:

- **i865P:** supports DDR266 and DDR333 memory only and doesn't support the 800MHz FSB.
- **i865PE:** as per i865P, plus 800MHz FSB and DDR400 memory support.
- **i865G:** as per i865PE, plus Intel's integrated graphics core.

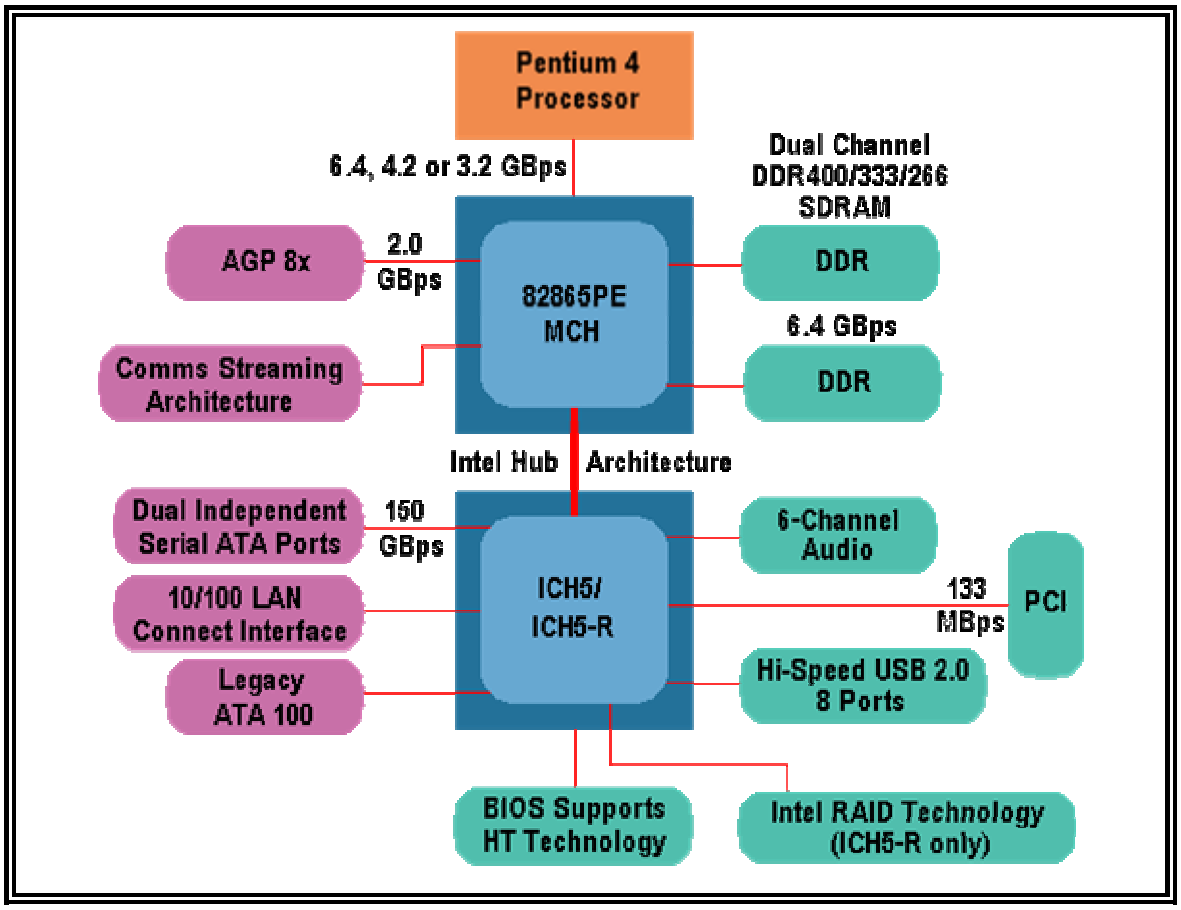


FIGURE 4.4 BLOCK DIAGRAM OF 865 CHIPSET

While the i865G's graphics core is the same as was featured on the i845G chipset, its performance will be faster, due both to a faster memory subsystem and a higher working frequency of the graphics core itself.

The following Table 4.1 shows comparisons of major characteristics of the i865P chipset with a selection of Intel's other recent Hyper-Threading chipset offerings:

	i865PE	**i875P**	**E7205**	**i845PE**	**i850E**
Processor	Pentium 4	Pentium 4	Pentium 4	Pentium 4 Celeron	Pentium 4 Celeron
System Bus (MHz)	800/533/400	800/533/400	533/400	533/400	533/400
Memory Modules	4 DIMMs	4 DIMMs	4 DIMMs	2 double-sided DDR DIMMs	4 RIMMs
Memory Type	Dual-Channel DDR 400/333/266 SDRAM	Dual-Channel DDR 400/333/266 SDRAM	Un-buffered only x72 or x64 DIMMs DDR SDRAM	DDR 333/266	PC1066 PC800-40 PC800-45 RDRAM
FSB/Memory Configurations	800/400 800/333 533/333 533/266 400/333 400/266	800/400 800/333 533/333 533/266	533/266 400/200	533/333 533/266 400/266	533/PC1066 533/PC800-40 400/PC800-45 400/PC800-40
Peak Memory Bandwidth	6.4GBps	6.4GBps	4.2GBps	2.7GBps	4.2GBps
Error Correction	N/A	ECC	ECC	N/A	ECC/Non-ECC

Graphics Interface	AGP 8x	AGP 8x	AGP 8x	AGP 4x	AGP 4x
Serial ATA	2 ports ATA 150	2 ports ATA 150	N/A	N/A	N/A
USB	8 ports Hi-Speed USB 2.0	8 ports Hi-Speed USB 2.0	6 ports Hi-Speed USB 2.0	6 ports Hi-Speed USB 2.0	4 ports USB 1.1

Table 4.1 comparisons of major characteristics of i865P chipset with a selection of Intel's other recent Hyper-Threading chipset

925X PCI express chipset

In the summer of 2004 Intel introduced a new family of chipsets that they claimed brought the most profound changes in PC platform architecture in more than a decade. The relative positioning of the chipsets - codenamed Alderwood and Grantsdale - is similar to that of the Canterwood and Springdale chipsets which preceded it. The 925X PCI Express chipset is the higher-end of the two, boasting a number of specific performance enhancements and being designed to deliver the ultimate gaming experience when coupled with Pentium 4 Extreme Edition CPUs.

The new chipsets are designed for use with the latest Prescott-cored Pentium 4 CPUs, designated by the new numeric model naming scheme - initially the 560 at 3.6GHz, down to the 520 at 2.8GHz. They will therefore only be used in motherboards that support Intel's innovative LGA775 package, which facilitates a direct electrical connection between the chip module substrate and the motherboard which the company claims will provide the robust power and signal delivery needed for future performance headroom.

All the new chipsets support Hyper-Threading, an 800MHz FSB and dual-channel DDR2-533 memory and enable a broad spectrum of new platform capabilities:

- Intel High Definition Audio enables multistreaming, 7.1 surround sound and dynamic jack retasking in a groundbreaking PC audio solution that provides performance comparable to high-end consumer electronics (CE) equipment.
- Intel Matrix Storage Technology provides the performance benefits of RAID 0 for media-intensive applications and the added protection of RAID 1 for critical digital media files and data on just two drives.
- The I/O Controller Hub 6 (ICH6R version) supports four 1.5 GBps Serial ATA (SATA) ports with Advanced Host Controller Interface (AHCI) capability, enabling Native Command Queuing for enhanced storage performance.
- Four PCI Express x1 high-speed expansion ports are ready for Gigabit Ethernet and future applications, including multiple TV tuners implemented in a single card.
- Intel Wireless Connect Technology enables users to create or expand a wireless network without external access point hardware. Intel Wireless Connect Technology requires a specific Intel 9XX Express Chipset and a separate Intel wireless LAN solution to operate.

Intel's new Flex memory system introduces some welcome flexibility, with dual-channel operation no longer being restricted to identical memory modules bought in matched pairs. Now the requirement is simply for the same amount of memory - whatever the configuration - in each of the two available banks.

Foremost amongst the innovations is the introduction of the PCI Express (PCX) bus technology. As digital video content becomes ever more important in today's electronic universe, no single aspect of the personal computing platform requires as much performance increase as the graphics interface.

The new chipsets address this need in the shape of the revolutionary 16x PCI Express graphics interface, as its name implies, an aggregation of 16 lanes. This provides the increased bandwidth and scalability necessary to tackle the most demanding multimedia tasks, with up to four times the theoretical maximum bandwidth over previous generation

AGP8X-based solutions - up to 4 GBps of peak bandwidth per direction and up to 8 GBps concurrent bandwidth.

AGP is unceremoniously consigned to history, the new chipsets providing no AGP interface at all. In time 1x PCX will replace the decade-old PCI standard.

915 Expressed Chipset

Announced at the same time as the i925X Express", the i915 Express chipset family - codenamed Grantsdale and comprising the i915P and i915G chipsets - have the same features as its sibling with the exception of some specific performance improvements.

The principal differences between the i915 and i925X chipsets are in graphics and memory support. The i915 supports traditional dual-channel DDR memory as well as the more expensive DDR2 variety. In addition, the i915G chipset includes an integrated Intel Graphics Media Accelerator 900, optimized for Microsoft DirectX 9 and capable of providing dual independent display capability with support for the latest 16:9 ratio monitors, in addition to conventional 4:3 displays.

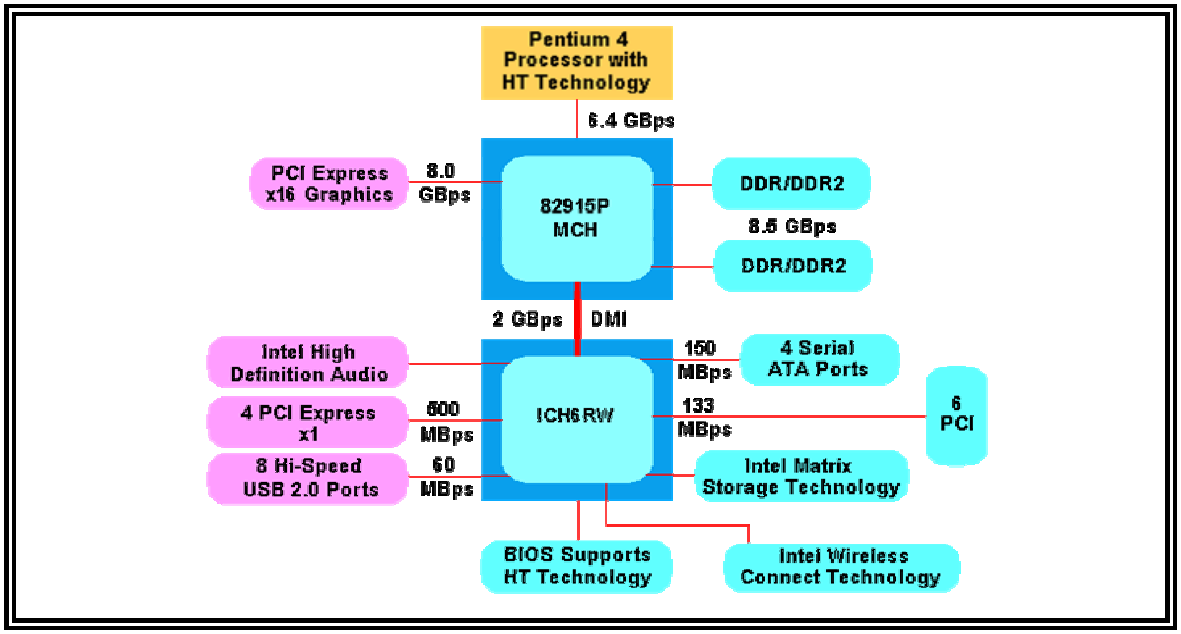


FIGURE 4.5 BLOCK DIAGRAM OF 915 EXPRESSED CHIPSET

The 3D graphics pipeline is broken up into four major stages, including geometry processing, setup (vertex processing), texture application and rasterisation. The Intel GMA 900 is optimized to use the Intel Pentium 4 processor for software-based geometry processing (such as transform and lighting) defined by Microsoft DirectX 9.

The Intel GMA 900 handles the remaining three stages, including converting vertices to pixels, applying textures to pixels, and rasterisation - the application of lighting and other effects to produce the final pixel value. From the rasterisation stage the Intel GMA 900 writes the final pixel value to the frame buffer for display. Intel GMA 900 includes two independent display pipelines that enable operation of dual displays.

The Intel GMA 900 utilizes a shared memory architecture, its support for dual-channel DDR2/533-MHz memory ensuring the memory bandwidth so critically important for quality and performance.

945 Express Chipset

Since the current chipsets didn't recognize more than one CPU, Intel had no choice but to release new chipsets at the same time as its dual-core Pentium D and Extreme Edition processors. Formerly codenamed "Lakeport", the mainstream 945 chipsets essentially provide the same features as the earlier 915 chipsets, plus support for the Pentium D processors.

However, the new chipsets aren't simply an artificial designation to allow motherboards to handle dual-core processors; there are technical differences and improvements too, albeit incremental ones.

The 945 chipset can handle front-side bus speeds up to 1066MHz and DDR2 memory up to 667MHz, providing up to 10.7GBps of peak memory bandwidth. The latter should provide a noticeable performance boost compared to DDR2-533 systems. Maximum RAM is be limited to 4GB and there's no provision for ECC support.

As with its predecessor, the 945 chipset is available in both P and G versions. The 845G differs from the 945P by the integration of Intel's GMA 950 onboard graphics on to

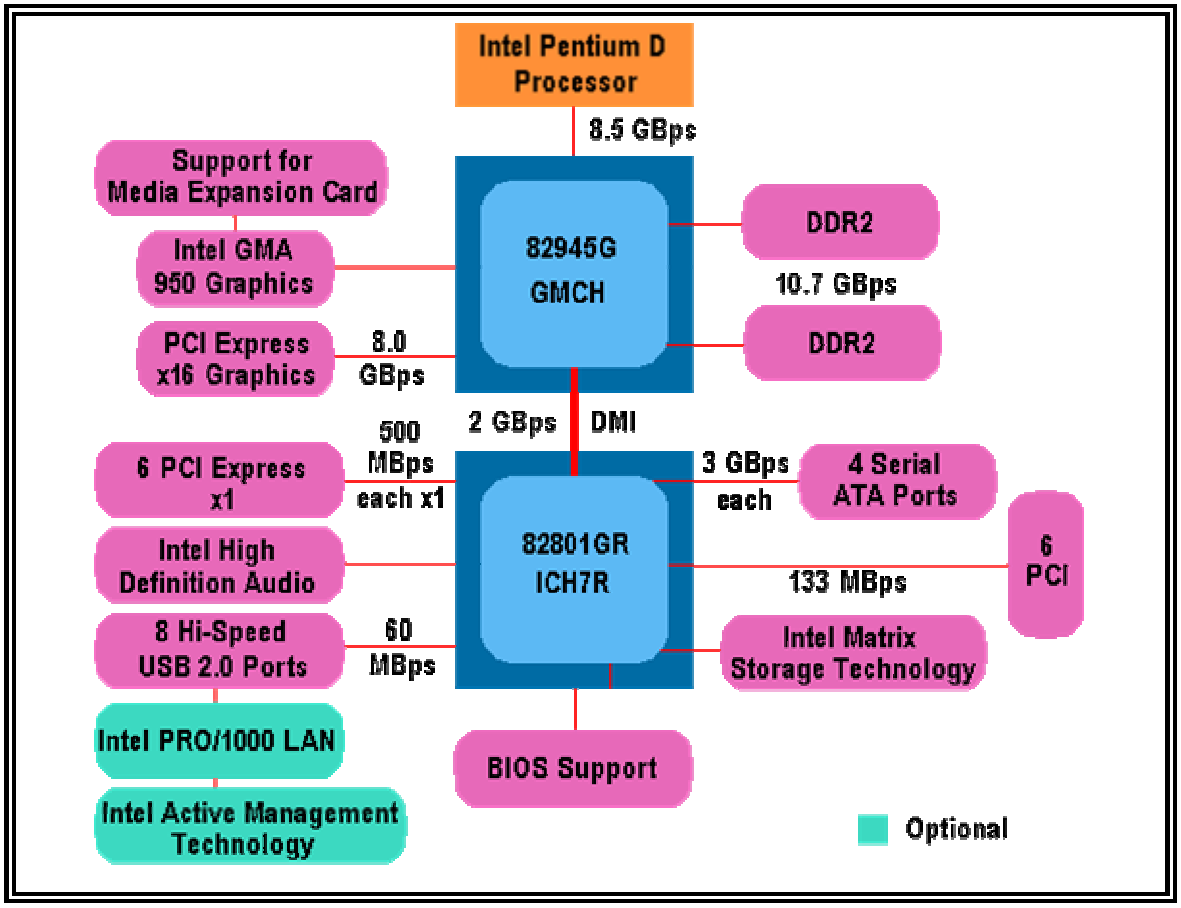


FIGURE 4.6 BLOCK DIAGRAM OF 945 EXPRESSED CHIPSET

the Northbridge. This is a faster version of the GMA 900 present on the 915G chipset, up from 333MHz to 400MHz. The moderate speed hike allows a throughput of up to 1.6GTexels/sec, more than adequate for 2D applications, allowing desktop resolutions of up to 2048 x 1536 pixels at 75Hz and the ability to run two monitors simultaneously via an ADD2 extension card. However, while DirectX 9 3D performance is improved, with Vertex Shader 3.0 and T&L operations still performed in software, this is insufficient for geometry-intensive games.

In addition to providing improved graphics, the GMA 950 supports Media Expansion Cards, allowing a user to take advantage of several video output options in a single-card solution, to provide video input capability and PVR functionality and to

support a wide range of display types and configurations, including support of HDTV playback on consumer electronic displays at either 720p or 1080i resolutions.

As with the previous ICH6 chips, the 945's Southbridge chip is available in two versions, the basic ICH7 or the ICH7R. Both include four integrated Serial ATA ports - supporting SATA's new 3 GBps transfer rates to suitable hard drives or optical devices - 8 USB2.0 ports, support for 6 PCI slots and High-Definition Audio, a worthwhile improvement over basic AC'97. The ICH7 offers 4 PCI-Express x1 lanes that can be combined to form a single x4 port. The ICHR version differs by adding a further two PCI-Express and an enhanced version of Intel's Matrix Storage Technology.

The latter allows two separate RAID partitions to be combined on one physical set of drives. In addition to RAID0 (striping) and RAID1 (mirroring), the new version also allows for a combination of RAID5 (striping with parity) and RAID10 (stripped mirrors). It also provides support for the AHCI specification, enabling hardware-assisted Native Command Queuing (NCQ) for faster boot times and file transfers and the hot-plugging of devices.

Available as options on both 945P and 945G chipsets are an Intel Gigabit LAN interface and Intel's Active Management technology, useful for monitoring and controlling PCs in an enterprise environment.

955X Express Chipset

Formerly codenamed "Glenwood", continues this practice, essentially providing the same features as the earlier 925X chipset, plus support for Pentium Extreme Edition processors.

Ever since the release of its 865 and 875 chipsets in 2003, Intel has segregated its Pentium 4 chipsets into performance and mainstream lines, the former using northbridges equipped with the best silicon and, consequently, lower memory access latencies than on the mainstream chipsets.

In a move that appears confirms the company's desire to differentiate its XE line of processors from their mainstream series processors, the 955X chipset continues this practice, its marginally superior performance being attributed to what is rather extravagantly referred to as Intel Memory Pipeline Technology (IMPT). A further differentiation of the 955X from the 945 chipsets is its lack of support for 533MHz system bus processors.

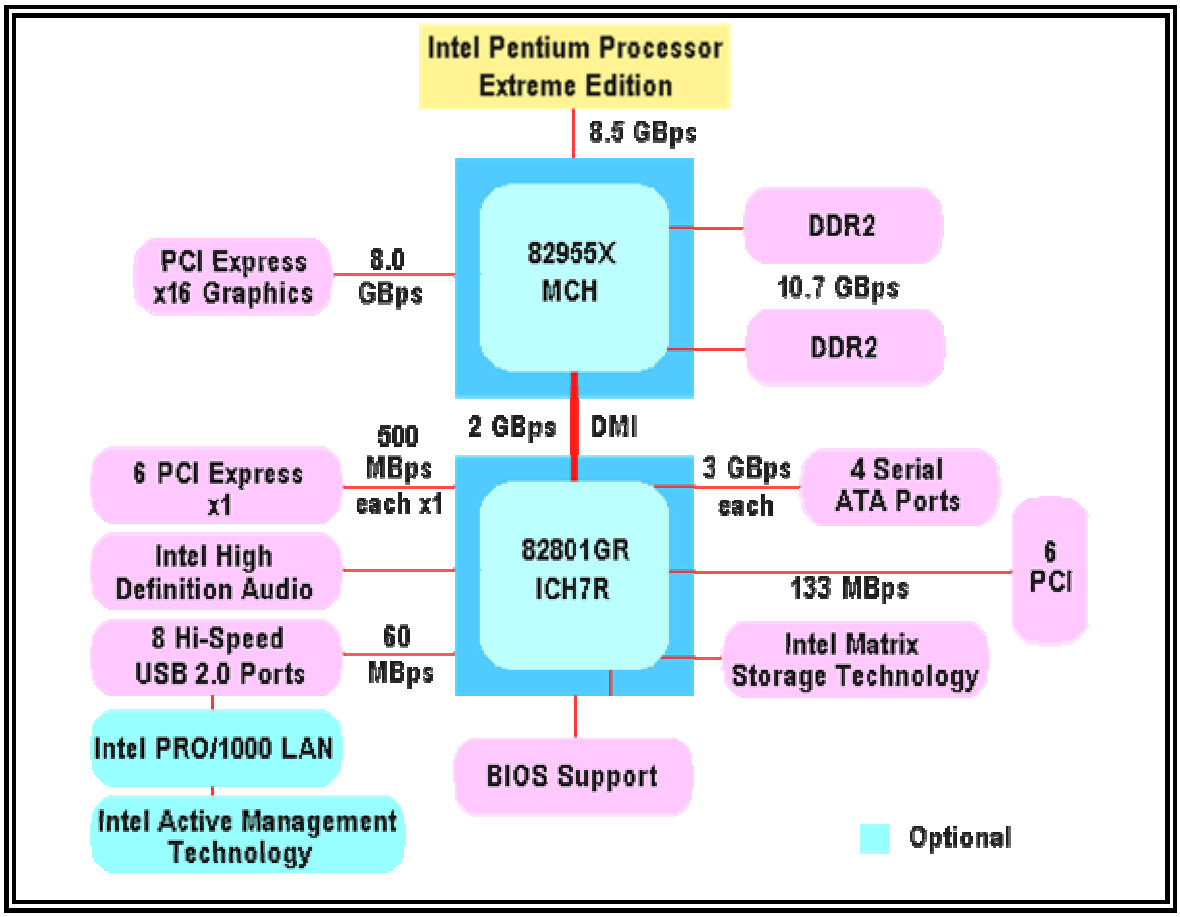


FIGURE 4.7 BLOCK DIAGRAM OF 955X EXPRESSED CHIPSET

However, the 955X's principal difference from its mainstream sibling is rather more significant; it's Intel's first desktop chipset to break the 4GB memory barrier, capitalizing on the removal of the 4GB limit facilitated by 64-bit edition of Windows XP by supporting up to 8GB of addressable main memory.

CHAPTER 5

VARIOUS MOTHERBOARDS

5.1 Introduction

To study various motherboards I took one motherboard from each stage of evolution. I have studied on the microprocessor, bus architectures and form factor from different generation of motherboards. We need to study microprocessor because actually we design motherboard for particular microprocessor.

Major new features of 286 is ISA Bus, coprocessor interrupt, bi-directional address bus, intelligent keyboard interface, software reset shadow Ram for system BIOS and Video Bios etc.. In 386 features introduced are EISA bus, cache memory, SIMM RAM, MIPS counter etc.. When 486 systems came, a local bus called VESA Bus was introduced to support high-speed transfer to display adaptor. Other than bus it introduced software speed selection, programmable DRAM wait states, PS/2 mouse support and power management. Soon Intel introduced PCI bus, which becomes standard for Pentium systems. Front side Bus has become the system bus due to the dual bus architecture of Pentium pro. And subsequent processors in Pentium family (Pentium IV) have drastically changed the system bus speed to 400 MHz. New version of P-IV offers 533 MHz speed. In near future this may go to several GHz.

5.2 286 Motherboard

In this part I have studied various 286 motherboards and specially on 286 processor and ISA bus architecture.

Intel 80286

The 80286 was the chip used in IBM's AT (advanced technology) system. The 286 was the first major step up in PC processors, providing significant performance increases over the 8088 and 8086--double or more performance at the same clock speed. The 286 also widened the address bus to allow access to 16 MB of memory, and introduced protected mode operation. It was originally available in 6 MHz and 8 MHz versions, but was later expanded to faster versions, all the way up to 20 MHz.

The 286 opened up the PC world to many users, but still was used mainly as the equivalent of a "turbo-charged 8088". At this time DOS was still the virtually exclusive operating system, and the protected mode the 286 offered was largely ignored.

The 80286 is of course considered obsolete today.

General Information	Manufacturer	Intel					
	Family Name	80286					
	Code name	--					
	Processor Generation	Second					
	Motherboard Generation	Second					
	Version	80286-6	80286-8	80286-10	80286-12	80286-16	80286-20
	Introduced	Feb. 1982		--	--	--	--
	Variants and Licensed Equivalents	AMD 286, 286S					
Speed Specifications	Memory Bus Speed (MHz)	6	8	10	12	16	20
	Processor Clock Multiplier	1.0					
	Processor Speed (MHz)	6	8	10	12	16	20
	"P" Rating	--					
Benchmarks	iCOMP Rating	--					
	iCOMP 2.0 Rating	--					
	Norton SI	3.1	4.4	5.6	6.7	~9.0	~11.5

	Norton SI32	--					
	CPUmark32	--					
Physical Characteristics	Process Technology	CMOS					
	Circuit Size (microns)	1.5					
	Die Size (mm^2)	47					
	Transistors (millions)	0.134					
Voltage, Power and Cooling	External or I/O Voltage (V)	5					
	Internal or Core Voltage (V)	5					
	Power Management	None					
	Cooling Requirements	None					
Packaging	Packaging Style	68-Pin PGA					
	Motherboard Interface	68-Pin Socket					
External Architecture	Data Bus Width (bits)	16					
	Maximum Data Bus Bandwidth (Mbytes/sec)	11.4	15.3	19.1	22.9	30.5	38.1
	Address Bus Width (bits)	24					
	Maximum Addressable Memory	16 MB					
	Level 2 Cache Type	None					
	Level 2 Cache Size	--					

	Level 2 Cache Bus Speed	--
	Multiprocessing	No
Internal Architecture	Instruction Set	80286
	MMX Support	No
	Processor Modes	Real, Protected
	x86 Execution Method	Native
Internal Components	Register Size (bits)	16
	Pipeline Depth (stages)	1
	Level 1 Cache Size	None
	Level 1 Cache Mapping	--
	Level 1 Cache Write Policy	--
	Integer Units	1
	Floating Point Unit / Math Coprocessor	Optional 80287 Coprocessor
	Instruction Decoders	1
	Branch Prediction Buffer Size / Accuracy	None
	Write Buffers	None
	Performance Enhancing Features	None

TABLE 5.1 INTEL 80286 PROCESSOR SUMMARY

Industry Standard Architecture (ISA) Bus

The most common bus in the PC world, ISA stands for *Industry Standard Architecture*, and unlike many uses of the word "standard", in this case it actually fits. The ISA bus is *still* a mainstay in even the newest computers, despite the fact that it is largely unchanged since it was expanded to 16 bits in 1984! The ISA bus eventually became a bottleneck to performance and was augmented with additional high-speed buses, but ISA persists because of the truly enormous base of existing peripherals using the standard. Also, there are still many devices for which the ISA's speed is more than sufficient, and will be for some time to come (standard modems being an example).

(As a side note, after 17 years it appears that ISA may finally be going the way of the dodo. Market leaders Intel and Microsoft want to move the industry away from the use of the ISA bus in new machines. My personal prediction is that they will succeed in this effort, but that it will take at least five years to do it fully. There are few standards in the PC world as pervasive as ISA, and the hundreds of millions of existing ISA cards will ensure that ISA sticks around for some time.)

The choices made in defining the main characteristics of the ISA bus--its width and speed--can be seen by looking at the processors with which it was paired on early machines. The original ISA bus on the IBM PC was 8 bits wide, reflecting the 8 bit data width of the Intel 8088 processor's system bus, and ran at 4.77 MHz, again, the speed of the first 8088s. In 1984 the IBM AT was introduced using the Intel 80286; at this time the bus was doubled to 16 bits (the 80286's data bus width) and speed increased to 8 MHz (the maximum speed of the original AT, which came in 6 MHz and 8 MHz versions).

Later, the AT processors of course got faster, and eventually data buses got wider, but by this time the desire for compatibility with existing devices led manufacturers to resist change to the standard, and it has remained pretty much identical since that time. The ISA bus provides reasonable throughput for low-bandwidth devices and virtually assures compatibility with almost every PC on the market.

Many expansion cards, even modern ones, are still only 8-bit cards (you can tell by looking at the edge connector on the card; 8-bit cards use only the first part of the ISA

slot, while 16-bit cards use both parts). Generally, these are cards for which the lower performance of the ISA bus is not a concern. However, access to IRQs 9 through 15 is provided through wires in the 16-bit portion of the bus slots. This is why most modems, for example, cannot be set to the higher-number IRQs. IRQs cannot be shared among ISA devices.

5.3 386 Motherboard

Intel 80386DX

The third generation processors (the 386 family) represent another step forward from the second generation class. Intel started with this generation of chips to create "subfamilies" of related chips with different capabilities, using the "DX" and "SX" designations.

The Intel 80386DX was the first true 32-bit processor used on the PC platform. Its internal register size was increased to 32 bits, and its data and address buses were as well, doubling data path width to the processor and increasing addressable memory to 4 GB theoretical. The 80386 family of chips offered more performance than the 80286s they replaced, largely through processor speed increases. The 386 did not offer the very large improvements over the 286 that the 286 did over the 8088.

The 386DX's increased power and the improved processor modes it offered (including full protected mode and virtual real modes) spurred the introduction of GUI-based operating systems on the PC, such as Microsoft Windows (although they are still quite slow on 386 chips). The instruction set of the 386 has set the standard for what is now called "x86" and hasn't changed very much since it was introduced. The 80386DX was the first to use pipelining to allow much improved processor performance through the use of much higher clock frequencies.

Invented by Intel, the 386 was also "cloned" by AMD and Cyrix. These are very good copies with no real compatibility problems. Intel only produced the 386DX up to 33

MHz (presumably to prevent overlapping into the 486's performance range) while AMD and Cyrix produced a 40 MHz version as well. This latter chip produced very good performance (for the time), comparable to many lower-end 486s. They are still however obsolete by today's standards, of course.

General Information	Manufacturer	Intel			Intel, AMD, Cyrix	AMD, Cyrix
	Family Name	80386DX				
	Code name	--				
	Processor Generation	Third				
	Motherboard Generation	Third				
	Version	80386DX-16	80386DX-20	80386DX-25	80386DX-33	80386DX-40
	Introduced	Oct. 1985	Feb. 1987	April 1988	April 1989	--
	Variants and Licensed Equivalents	--				
Speed Specifications	Memory Bus Speed (MHz)	16	20	25	33	40
	Processor Clock Multiplier	1.0				
	Processor Speed (MHz)	16	20	25	33	40
	"P" Rating	--				
Benchmarks	iCOMP Rating	~29	~38	49	68	~85
	iCOMP 2.0 Rating	--				
	Norton SI	~15	~20	~25	35	~43

	Norton SI32	--				
	CPUmark32	--				
Physical Characteristics	Process Technology	CMOS				
	Circuit Size (microns)	originally 1.5, now 1.0				
	Die Size (mm^2)	--				
	Transistors (millions)	0.275				
Voltage, Power and Cooling	External or I/O Voltage (V)	5				
	Internal or Core Voltage (V)	5				
	Power Management	None				
	Cooling Requirements	None				
Packaging	Packaging Style	132-Pin PGA				
	Motherboard Interface	132-Pin Socket				
External Architecture	Data Bus Width (bits)	32				
	Maximum Data Bus Bandwidth (Mbytes/sec)	63.6	76.3	95.4	127.2	152.6
	Address Bus Width (bits)	32				
	Maximum Addressable Memory	4 GB				
	Level 2 Cache Type	None				
	Level 2 Cache Size	--				
	Level 2 Cache Bus Speed	--				
	Multiprocessing	No				
Internal	Instruction Set	x86				

Architecture	MMX Support	No
	Processor Modes	Real, Protected, Virtual Real
	x86 Execution Method	Native
Internal Components	Register Size (bits)	32
	Pipeline Depth (stages)	4
	Level 1 Cache Size	None
	Level 1 Cache Mapping	--
	Level 1 Cache Write Policy	--
	Integer Units	1
	Floating Point Unit / Math Coprocessor	Optional 80387 Coprocessor
	Instruction Decoders	1
	Branch Prediction Buffer Size / Accuracy	None
	Write Buffers	None
	Performance Enhancing Features	None

TABLE 5.2 INTEL AND OTHER MANUFACTURE OF 80386DX PROCESSOR SUMMARY

Intel 80386SX

The 80386SX is a "lite" version of the 80386DX chip. It uses only a 16-bit data bus, the same external bus width as the 80286. It also only can address 16 MB of memory, like the 80286. The SX version of the 386 chip was in fact released well after the DX, and in some ways was intended to move the market away from the 286 since it had roughly the same interfaces but better performance. It was actually introduced several *years* after the DX.

Note: It is a common myth that the 80386SX can be substituted in place of an 80286 in a 286 motherboard. While the chips are compatible in terms of their external interfaces they use different packaging. A 386SX could be put into a 286 motherboard if a proper adapter were used.

The narrower data bus width of the 386SX creates a reduction in performance of about 20-25% compared to an equivalent-speed 386DX; a significant difference but not excessive. The 80386SX chip was a popular choice for the first small notebook computers, especially the 386SL variant that introduced the SMM power management features to the Intel line. The 386SX is still a 32-bit processor internally and will run 32-bit software (really slowly). It is available in speeds from 16 to 33 MHz; a 40 MHz version was not produced for the SX, and is also available in cloned versions from AMD and Cyrix.

The 386SX chip is today considered obsolete.

General Information	Manufacturer	Intel, AMD, Cyrix			
	Family Name	80386SX			
	Code name	"P9"			
	Processor Generation	Third			
	Motherboard Generation	Third			
	Version	80386SX -16	80386SX -20	80386SX -25	80386SX -33
	Introduced	June 1988	Jan. 1989	June 1988	Oct. 1992
	Variants and Licensed Equivalents	386SL (power management features)			
Speed Specifications	Memory Bus Speed (MHz)	16	20	25	33
	Processor Clock Multiplier	1.0			

	Processor Speed (MHz)	16	20	25	33
	"P" Rating	--			
Benchmarks	iCOMP Rating	22	32	39	56
	iCOMP 2.0 Rating	--			
	Norton SI	~11	15	~20	~27
	Norton SI32	--			
	CPUmark32	--			
Physical Characteristics	Process Technology	CMOS			
	Circuit Size (microns)	Originally 1.5, now 1.0			
	Die Size (mm^2)	--			
	Transistors (millions)	0.275 (0.855 for 386SL)			
Voltage, Power and Cooling	External or I/O Voltage (V)	5			
	Internal or Core Voltage (V)	5			
	Power Management	SMM on 386SL only			
	Cooling Requirements	None			
Packaging	Packaging Style	132-Pin PGA			
	Motherboard Interface	132-Pin Socket			
External Architecture	Data Bus Width (bits)	16			
	Maximum Data Bus Bandwidth	31.8	38.1	47.7	63.6

	(Mbytes/sec)				
	Address Bus Width (bits)	24			
	Maximum Addressable Memory	16 MB			
	Level 2 Cache Type	None			
	Level 2 Cache Size	--			
	Level 2 Cache Bus Speed	--			
	Multiprocessing	No			
Internal Architecture	Instruction Set	x86			
	MMX Support	No			
	Processor Modes	Real, Protected, Virtual Real			
	x86 Execution Method	Native			
Internal Components	Register Size (bits)	32			
	Pipeline Depth (stages)	4			
	Level 1 Cache Size	None			
	Level 1 Cache Mapping	--			
	Level 1 Cache Write Policy	--			
	Integer Units	1			
	Floating Point Unit / Math Coprocessor	Optional 80387 Coprocessor			
	Instruction	1			

	Decoders	
	Branch Prediction Buffer Size / Accuracy	None
	Write Buffers	None
	Performance Enhancing Features	None

TABLE 5.3 INTEL AND OTHER MANUFACTURE OF 80386SX PROCESSOR SUMMARY

Micro Channel Architecture (MCA) Bus

The MCA bus (also called the *Micro Channel* bus; MCA stands for "Micro Channel Architecture") was IBM's attempt to replace the ISA bus with something "bigger and better". When the 80386DX was introduced in the mid-80s with its 32-bit data bus, IBM decided (much like it did with the AT) to create a bus to match this width. MCA is 32 bits wide, and offers several significant improvements over ISA. (One of MCA's disadvantages was rather poor DMA controller circuitry.)

The MCA bus has some pretty impressive features considering that it was introduced in 1987, a full seven years before the PCI bus made similar features common on the PC. In some ways it was ahead of its time, because back then the ISA bus really wasn't a major performance limiting factor:

- 32 Bit Bus Width: The MCA bus features a full 32 bit bus width, the same width as the VESA and PCI local buses. It had far superior throughput to the ISA bus.
- Bus Mastering: The MCA bus supported bus mastering adapters for greater efficiency, including proper bus arbitration.
- Plug and Play: MCA automatically configured adapter cards, so there was no need to fiddle with jumpers. This was eight years before Windows 95 brought PnP into the mainstream!

MCA had a great deal of potential. Unfortunately, IBM made two decisions that would doom MCA to utter failure in the marketplace. First, they made MCA incompatible with ISA; this means ISA cards will not work at all in an MCA system, one of the few categories of PCs for which this is true. The PC market is *very* sensitive to backwards-compatibility issues, as evidenced by the number of older standards that persist to this day (such as ISA!) Second, IBM decided to make the MCA bus proprietary. It in fact did this with ISA as well; however in 1981 IBM could afford to flex its muscles in this manner, while by this time the clone makers were starting to come into their own and weren't interested in bending to IBM's wishes.

These two factors, combined with the increased cost of MCA systems, led to the demise of the MCA bus. With the PS/2 now discontinued, MCA is dead on the PC platform, though it is still used by IBM on some of its RISC 6000 UNIX servers. It is one of the classical examples in the field of computing of how non-technical issues often dominate over technical ones.

Extended Industry Standard Architecture (EISA) Bus

EISA stands for *Extended Industry Standard Architecture*. Unlike ISA, here the name is *not* indicative of reality, for the EISA bus never became widely used and cannot by any stretch of the imagination be considered an industry standard. EISA began as Compaq's answer to IBM's MCA bus, and followed a similar path of development--with very similar results.

Compaq avoided the two key mistakes that IBM made when they developed EISA. First, they made it compatible with the ISA bus. Second, they opened the design to all manufacturers instead of keeping it proprietary, by forming the non-profit EISA committee to manage the design of the standard. EISA was similar to MCA both in terms of technology and market acceptance: it had significant technical advantages over ISA, and it never caught on with the PC-buying public.

Some of the key features of the EISA bus:

- ISA Compatibility: ISA cards will work in EISA slots.
- 32 Bit Bus Width: Like MCA, the bus was expanded to 32 bits.
- Bus Mastering: The EISA bus supports bus mastering adapters for greater efficiency, including proper bus arbitration.
- Plug and Play: EISA automatically configures adapter cards, similar to the Plug and Play standards of modern systems.

EISA-based systems have today been mostly relegated to a specialty role; they are sometimes found in network fileservers. The EISA bus is virtually non-existent on desktop systems for several reasons. First, EISA-based systems tend to be much more expensive than other types of systems. Second, there are few EISA-based cards available. Finally, the performance of this bus is quite low compared to the popular local buses like the VESA Local Bus and PCI. EISA is not totally dead as a platform the way MCA is, but it is pretty close.

5.4 486 Motherboard

Intel 80486DX

The fourth generation of processors saw rapid growth in the CPUs' power and capabilities, and the introduction of several new technologies as well. It was here that AMD and Cyrix had their first real early successes in not just emulating Intel's designs but coming up with real value leaders of their own, at the top end of the fourth generation scale. Also, it was at this time that the new phenomenon of upgradeable processors and standardized motherboard sockets began to become prevalent.

The Intel 80486DX was the first member of the 486 family (which has many more members than the 386 family did). It provided a very significant increase in power over the 80386DX processor, in fact, far more proportionately than the 386 did over the 286. A 486DX processor provides approximately 100 to 150% more performance than a 386DX of the same clock speed. The 80486 brought GUIs to the mainstream on PCs; it is the minimum processor that most people consider "usable" for running an operating system like Microsoft Windows.

Interestingly, the 486 does not provide its performance improvements by widening any of the buses, as had been the case in the previous two generations: it is still a 32-bit processor with 32-bit data and address buses, just like the 386DX. However, internally, the 486 incorporates several significant improvements over the 386:

- Faster Overall Instruction Execution time: The core of the chip can execute instructions in less time than earlier processors.
- Deeper Pipeline: The execution pipeline was increased by one step.
- Primary Cache: The 486 processor was the first to incorporate level 1 cache on the chip, to reduce the number of required accesses to memory.
- Integrated Floating Point Unit: The chip includes an integrated math coprocessor (not on the SX version however). In addition, the coprocessor provides much more performance than the optional 80387 used with 386 chips, in part because it is integrated into the chip.
- Burst Mode: The 486 introduced the use of burst mode to reduce wait time on memory accesses.
- Power Management: SL power management enhancements as an option (instead of a specialized SL chip as was the case for the 386SL).
- Improved Support Architecture: In general, 486 motherboards were more efficient than 386 ones, and began to use secondary cache as well. This improves processor performance significantly.

Interestingly, the 80487SX coprocessor that is intended for use with the SX version of the 80486, is in fact a full-blown 80486DX processor. This is part of how Intel structured its fourth generation family's upgrade path. See the description of the 80486SX for more details on this.

AMD and Cyrix did produce clones of the original Intel 80486 processors, but they were not a big player in the 486 clone market until the higher-speed DX2 and DX4 processors.

The 486DX is considered obsolete, although the chip still has fairly good power for performing a wide variety of light tasks, such as word processing and some older games,

and light Internet access. The 486DX-50, which runs on a 50 MHz system bus, provides performance comparable to the 486DX2-66 in many ways, because the latter uses only a 33 MHz system bus. The 486DX-50 was not used in nearly as many systems as the other processor speeds were. It should not be confused with the 486DX2-50, which runs at the same processor clock speed but is clock-doubled relative to the system bus (which runs at 25 MHz).

Note: The 486DX processor normally was purchased as part of a new system only, not as part of an upgrade. Most early 486 systems used a 168-pin socket for the chip, which predates the numbered standardized socket system that Intel created. The 486DX will fit into a Socket 1, Socket 2 or Socket 3 however.

General Information	Manufacturer	Intel, AMD, Cyrix		
	Family Name	80486DX		
	Code name	"P4"		
	Processor Generation	Fourth		
	Motherboard Generation	Fourth		
	Version	80486DX-25	80486DX-33	80486DX-50
	Introduced	April 1989	May 1990	June 1991
	Variants and Licensed Equivalents	80487SX		
Speed Specifications	Memory Bus Speed (MHz)	25	33	50
	Processor Clock Multiplier	1.0		
	Processor Speed (MHz)	25	33	50
	"P" Rating	--		
Benchmarks	iCOMP Rating	122	166	249
	iCOMP 2.0 Rating	--		

	Norton SI	54	72	109
	Norton SI32	--		
	CPUmark32	--		
Physical Characteristics	Process Technology	CMOS		
	Circuit Size (microns)	1.0	1.0	0.8
	Die Size (mm^2)	81		81?
	Transistors (millions)	1.2		
Voltage, Power and Cooling	External or I/O Voltage (V)	5		
	Internal or Core Voltage (V)	5		
	Power Management	SMM in SL-enhanced versions		
	Cooling Requirements	Generally none, some use passive heat sink.		
Packaging	Packaging Style	168-Pin PGA		
	Motherboard Interface	168-Pin Socket, Socket 1, Socket 2, Socket 3		
External Architecture	Data Bus Width (bits)	32		
	Maximum Data Bus Bandwidth (Mbytes/sec)	95.4	127.2	190.7
	Address Bus Width (bits)	32		
	Maximum Addressable Memory	4 GB		
	Level 2 Cache Type	Motherboard		
	Level 2 Cache Size	Usually 0 KB to 256 KB		
	Level 2 Cache Bus Speed	Same as Memory Bus		
	Multiprocessing	No		

Internal Architecture	Instruction Set	x86
	MMX Support	No
	Processor Modes	Real, Protected, Virtual Real
	x86 Execution Method	Native
Internal Components	Register Size (bits)	32
	Pipeline Depth (stages)	5
	Level 1 Cache Size	8 KB Unified
	Level 1 Cache Mapping	4-Way Set Associative
	Level 1 Cache Write Policy	Write-Through
	Integer Units	1
	Floating Point Unit / Math Coprocessor	Integrated
	Instruction Decoders	1
	Branch Prediction Buffer Size / Accuracy	None
	Write Buffers	None
	Performance Enhancing Features	None

TABLE 5.4 INTEL AND OTHER MANUFACTURE OF 80486DX PROCESSOR SUMMARY

Intel 80486SX

The Intel 80486SX is the same chip as the 80486DX with one exception: the lack of an integrated math coprocessor (floating point unit). Note that this is a different kind of difference between the SX and DX versions than is the case with the 386; the 386SX and

386DX both had no coprocessor and the 386SX had narrower data and address buses than the 386DX. The 486DX and 486SX have the same bus widths. Since it is the same chip except for the floating point processor, the 486SX has the same advantages over the 386 that the 486DX does. Note that the 486SX was made available in slower clock speeds than the 486DX; the SX comes in 16, 20, 25 and 33 MHz versions, while the DX is 25, 33 and 50 MHz.

Intel made some rather interesting (some would say bizarre) decisions about how to market the 486 line. The 486SX was in many ways a marketing gimmick only; it was in fact a 486DX with the math coprocessor disabled! Why did Intel bother to sell a full 486DX and a "crippled" one (the 486SX)? Mainly for market targeting reasons: they wanted to be able to fill the niche of low-end buyers without dropping the price on their "top of the line" 486DX. The 486SX was a popular choice for laptops due to its lower cost and lower power consumption.

The 80486SX uses a "math coprocessor" called the 80487SX (there is no 80487DX at all so don't be confused by that). The 80487SX is, in fact, a fully functional 80486DX chip! What Intel wanted people to think was that (like with its earlier coprocessors) you would put the 80487SX in and it would handle the math functions. In fact, when inserted, the 80487SX shuts down the 80486SX and handles both integer and floating point operations (since it is internally a 80486DX, which does both). This makes no difference from a performance standpoint but is kind of a technical curiosity.

With the large number of very cheap 486DX and faster chips and systems around, the 486SX is really quite obsolete. In addition, it does not come in a 50 MHz version. If you are not doing any floating point operations at all it is comparable to a 486DX of the same speed, but there is usually no need to have to worry about this given that even the 486DX is quite obsolete at this time anyway.

Note: The 486SX processor normally was purchased as part of a new system only, not as part of an upgrade. Most early 486 systems used a 168-pin socket for the chip, which predates the numbered standardized socket system that Intel created. The 486SX will fit into a Socket 1, Socket 2 or Socket 3 however.

General Information	Manufacturer	Intel, AMD, Cyrix			
	Family Name	80486SX			
	Code name	"P4S"			
	Processor Generation	Fourth			
	Motherboard Generation	Fourth			
	Version	80486SX-16	80486SX-20	80486SX-25	80486SX-33
	Introduced	Sept. 1991	April 1991	Sept. 1991	Sept. 1992
	Variants and Licensed Equivalents	--			
Speed Specifications	Memory Bus Speed (MHz)	16	20	25	33
	Processor Clock Multiplier	1.0			
	Processor Speed (MHz)	16	20	25	33
	"P" Rating	--			
Benchmarks	iCOMP Rating	63	78	100	136
	iCOMP 2.0 Rating	--			
	Norton SI	34	42	54	72
	Norton SI32	--			
	CPUmark32	--			
Physical Characteristics	Process Technology	CMOS			
	Circuit Size (microns)	1.0			
	Die Size (mm^2)	67			

	Transistors (millions)	1.185			
Voltage, Power and Cooling	External or I/O Voltage (V)	5			
	Internal or Core Voltage (V)	5			
	Power Management	SMM in SL-enhanced versions			
	Cooling Requirements	None			
Packaging	Packaging Style	168-Pin PGA			
	Motherboard Interface	168-Pin Socket, Socket 1, Socket 2, Socket 3			
External Architecture	Data Bus Width (bits)	32			
	Maximum Data Bus Bandwidth (Mbytes/sec)	63.6	76.3	95.4	127.2
	Address Bus Width (bits)	32			
	Maximum Addressable Memory	4 GB			
	Level 2 Cache Type	Motherboard			
	Level 2 Cache Size	Usually 0 KB to 256 KB			
	Level 2 Cache Bus Speed	Same as Memory Bus			
	Multiprocessing	No			
Internal Architecture	Instruction Set	x86			
	MMX Support	No			
	Processor Modes	Real, Protected, Virtual Real			
	x86 Execution	Native			

	Method	
Internal Components	Register Size (bits)	32
	Pipeline Depth (stages)	5
	Level 1 Cache Size	8 KB Unified
	Level 1 Cache Mapping	4-Way Set Associative
	Level 1 Cache Write Policy	Write-Through
	Integer Units	1
	Floating Point Unit / Math Coprocessor	Optional 80487SX Coprocessor
	Instruction Decoders	1
	Branch Prediction Buffer Size / Accuracy	None
	Write Buffers	None
	Performance Enhancing Features	None

TABLE 5.5 INTEL AND OTHER MANUFACTURE OF 80486SX PROCESSOR SUMMARY

Intel 80486DX2 and 80486DX2 Overdrive

The 80486DX2 was the first chip to use "clock doubling" technology, where the processor runs at a faster speed than the memory bus it talks to. This was done to allow the processor speed to be increased without having to deal with the much more difficult task of increasing motherboard speed. Chips that run at faster than memory bus speed improve performance but at a diminishing rate as the multiplier increases, due to the processor waiting for data from memory. This is discussed in detail here.

Intel produced 50 and 66 MHz DX2 chips, intended for use in 25 and 33 MHz system bus systems. These chips have been sold as regular chips intended for use in new systems, which generally come in 168 pin packages to go in the original 168 pin socket used in 486 systems. They have also been made in 169 pin OverDrive versions to go in Socket 1 (the original OverDrive socket). These can be used to upgrade older 486DX or 486SX systems.

AMD and Cyrix not only cloned the 66 MHz DX2 processor, they took Intel one step further with the 80486DX2-80, running at 80 MHz. This uses a 40 MHz system bus, which isn't a speed that is normally used by Intel systems but that became more popular late in the 486 life cycle due to the performance increase it gives over 33 MHz bus systems. In addition, the AMD (enhanced version) and Cyrix chips have several advantages over the Intel chips (they had the benefit of developing them well after Intel):

- Power Management: They have SMM (power management) built in.
- Write-Back Cache: They have write-back capability for the primary cache which provides a small boost in performance.
- Lower Power: They use less power because they run at 3 volts, but will still work in standard 5 volt motherboard sockets without a converter or regulator (they are said to be "5 volt tolerant").

Other than clock speed, the 80486DX2 is virtually identical to the 80486DX. They are obsolete due to the availability of faster, very inexpensive processors such as the 5x86-133 that go in the same sockets, but 486DX2 systems are perfectly viable for many uses, including routine office word processing and spreadsheet work under DOS and Windows 3.x. The 80486DX2-66 is by far the most common version of this chip; a great number of these systems were produced and many are still in use today, especially in small businesses.

Note: The 486DX2 was the first processor to really require a heat sink in order to operate reliably. The increased speed of this chip means that it runs very hot (at least the Intel 5 volt versions).

General Information	Manufacturer	Intel	Intel, AMD, Cyrix	AMD, Cyrix
	Family Name	80486DX2		
	Code name	"P24"		
	Processor Generation	Fourth		
	Motherboard Generation	Fourth		
	Version	80486DX2-50	80486DX2-66	80486 DX2-80
	Introduced	March 1992	Aug. 1992	--
	Variants and Licensed Equivalents	--		
Speed Specifications	Memory Bus Speed (MHz)	25	33	40
	Processor Clock Multiplier	2.0		
	Processor Speed (MHz)	50	66	80
	"P" Rating	--		
Benchmarks	iCOMP Rating	231	297	~340
	iCOMP 2.0 Rating	--		
	Norton SI	109	144	173
	Norton SI32	~7	~8	~10
	CPUmark32	~45	~65	~105
Physical Characteristics	Process Technology	CMOS		
	Circuit Size (microns)	0.8 (Intel), 0.5 (AMD), 0.65? (Cyrix)		
	Die Size (mm^2)	76 (Intel)		
	Transistors (millions)	1.2 (Intel)		

Voltage, Power and Cooling	External or I/O Voltage (V)	5 (Intel) 3.3 (AMD, Cyrix, 5V tolerant)		
	Internal or Core Voltage (V)	5 (Intel), 3.3 (AMD, Cyrix)		
	Power Management	SMM in AMD, Cyrix, and SL-enhanced Intel versions		
	Cooling Requirements	Passive or active heat sink		
Packaging	Packaging Style	168-Pin PGA (Regular), 169-Pin PGA (OverDrive)		
	Motherboard Interface	168-Pin Socket (Regular only), Socket 1, Socket 2, Socket 3 (Regular or OverDrive)		
External Architecture	Data Bus Width (bits)	32		
	Maximum Data Bus Bandwidth (Mbytes/sec)	95.4	127.2	152.6
	Address Bus Width (bits)	32		
	Maximum Addressable Memory	4 GB		
	Level 2 Cache Type	Motherboard		
	Level 2 Cache Size	Usually 64 KB to 256 KB		
	Level 2 Cache Bus Speed	Same as Memory Bus		
	Multiprocessing	No		
Internal Architecture	Instruction Set	x86		
	MMX Support	No		
	Processor Modes	Real, Protected, Virtual Real		
	x86 Execution Method	Native		
Internal	Register Size (bits)	32		

Components	Pipeline Depth (stages)	5
	Level 1 Cache Size	8 KB Unified
	Level 1 Cache Mapping	4-Way Set Associative
	Level 1 Cache Write Policy	Write-Through, Write-Back (AMD, Cyrix)
	Integer Units	1
	Floating Point Unit / Math Coprocessor	Integrated
	Instruction Decoders	1
	Branch Prediction Buffer Size / Accuracy	None
	Write Buffers	None
	Performance Enhancing Features	None

TABLE 5.6 INTEL AND OTHER MANUFACTURE OF 80486DX2 PROCESSOR SUMMARY

Intel 80486DX4 and 80486DX4 OverDrive

The 80486DX4 continued the trend started by the 80486DX2 toward faster clock speed processors on slower motherboards. The DX4 uses "clock tripling", where the processor runs at three times the speed of the memory bus (motherboard). Note that despite the name the DX4 does not run at four times the memory bus speed ("DX3" refers to 2.5 times the memory bus, but never became a shipping product for the 486.)

Intel's DX4 runs at two speeds: 75 MHz (for the 25 MHz bus) and 100 MHz (for the 33 MHz), with the 100 being by far the most popular. In order to keep power and heat to a manageable level the voltage of these chips is reduced to 3.3 volts. The AMD and Cyrix versions are very similar to their DX2 cousins, in that they are 5 volt tolerant, meaning they can handle being put in a 5 volt motherboard; the Intel chip is not 5 volt tolerant and cannot be put into a 5 volt board (well, you can physically put them into the board, but you shouldn't do it!) AMD and Cyrix also make a 120 MHz version that runs on a 40

MHz bus; this is relatively rare. The Intel, AMD and Cyrix versions of the DX4 chip are quite similar. One difference is that the AMD and Cyrix support write-back cache while the Intel does not; however the Intel has its level 1 cache doubled to 16 KB. All three chips support power management.

The 486DX4 was most commonly put into systems as OverDrive processors for older, slower systems. Intel did not originally make the DX4 available as a stand-alone chip, presumably since it shipped after the Pentium had already been introduced. However, they did make the chip available stand-alone later on. New 486DX4-100 systems were often made with the AMD version of the chip. The OverDrive version is generally required for older motherboards that don't support 3.3 volt power.

Like the 80486DX2, the 80486DX4 is technically obsolete, but the 100 MHz version especially has good processing power for routine applications. It is also acceptable for some other uses, but is not found in modern systems.

General Information	Manufacturer	Intel, AMD, Cyrix		AMD, Cyrix
	Family Name	80486DX4		
	Code name	"P24C"		
	Processor Generation	Fourth		
	Motherboard Generation	Fourth		
	Version	80486DX4-75	80486DX4-100	80486DX4-120
	Introduced	March 1994		--
	Variants and Licensed Equivalents	--		
Speed Specifications	Memory Bus Speed (MHz)	25	33	40
	Processor Clock Multiplier	3.0		
	Processor Speed	75	100	120

	(MHz)			
	"P" Rating	--		
Benchmarks	iCOMP Rating	319	435	~530
	iCOMP 2.0 Rating	--		
	Norton SI	~155	198	259
	Norton SI32	~12	15	17
	CPUmark32	~100	~125	~150
Physical Characteristics	Process Technology	CMOS		
	Circuit Size (microns)	0.6 (Intel), 0.5 (AMD), 0.65? (Cyrix)		
	Die Size (mm^2)	--		
	Transistors (millions)	1.6 (Intel)		
Voltage, Power and Cooling	External or I/O Voltage (V)	3.3 (AMD, Cyrix, 5V tolerant)		
	Internal or Core Voltage (V)	3.3		
	Power Management	SMM		
	Cooling Requirements	Active heat sink		
Packaging	Packaging Style	168-Pin PGA (Regular), 169-Pin PGA (OverDrive)		
	Motherboard Interface	168-Pin Socket (Regular only), Socket 1, Socket 2, Socket 3 (Regular or OverDrive)		
External Architecture	Data Bus Width (bits)	32		
	Maximum Data Bus Bandwidth (Mbytes/sec)	95.4	127.2	152.6
	Address Bus Width (bits)	32		
	Maximum	4 GB		

	Addressable Memory	
	Level 2 Cache Type	Motherboard
	Level 2 Cache Size	Usually 64 KB to 256 KB
	Level 2 Cache Bus Speed	Same as Memory Bus
	Multiprocessing	No
Internal Architecture	Instruction Set	x86
	MMX Support	No
	Processor Modes	Real, Protected, Virtual Real
	x86 Execution Method	Native
Internal Components	Register Size (bits)	32
	Pipeline Depth (stages)	5
	Level 1 Cache Size	8 KB (AMD, Cyrix), 16 KB (Intel) Unified
	Level 1 Cache Mapping	4-Way Set Associative
	Level 1 Cache Write Policy	Write-Through, Write-Back (AMD, Cyrix)
	Integer Units	1
	Floating Point Unit / Math Coprocessor	Integrated
	Instruction Decoders	1
	Branch Prediction Buffer Size / Accuracy	None
	Write Buffers	None
	Performance Enhancing Features	None

TABLE 5.7 INTEL AND OTHER MANUFACTURE OF 80486DX4 PROCESSOR SUMMARY

AMD 5x86 (80486DX5)

The 486DX4-100 was the fastest 486 processor made by Intel before they decided to leave the fourth generation and concentrate on the Pentium. AMD took clock multiplying one increment further with what it calls the "5x86" chip. Despite the implication that it is a fifth-generation chip, it is not--it is a high speed, clock-quadrupled 486 processor, that runs in 486 motherboards. It does use a rather advanced 0.35 micron CMOS process (advanced for a fourth-generation chip that is).

The AMD 5x86 was made available in one speed only, 133 MHz, for use in 33 MHz motherboards. The processor runs at four times the system clock and fits into a Socket 3. Since having a clock multiplier of four was not part of the original Socket 3 design, AMD made the 5x86 look for a two times setting from the motherboard and interpret that as four times instead. In other words, to use the 5x86 you want to set the motherboard to the 2x setting. This will actually cause the 5x86 to run at 4x. The chip will actually physically fit into an older 486 socket such as a socket 1 or 2 or the original 168-pin 486 socket, but doing this requires a voltage regulator since the AMD chip runs at 3.3 volts.

AMD calls this chip the "5x86-P75" because it offers performance comparable to low-end fifth-generation chips. In fact, it is comparable in integer performance to a 75 MHz Pentium. It is also called the 80486DX5-133, which is a more accurate reflection of what the chip really is. Architecturally, the chip is virtually identical to the 80486DX4, except that it matches Intel's 16 KB level 1 cache (the AMD 80486DX4 has only 8 KB). Note that this is not the case with Cyrix's 5x86 chip, which is quite different.

The 5x86-133 is the most powerful 486-class chip available (the Cyrix 5x86-120 is actually faster but was discontinued by Cyrix many months before AMD stopped mass-producing the 5x86-133). The 5x86-133 has traditionally been an excellent choice for an economy PC, especially for home-builders, because not only is the chip itself very inexpensive, so are the motherboards that use it. With 200+ MHz Pentium and Pentium-compatible chips coming well down in price, the 5x86-133 is following the rest of the fourth-generation chips to obsolescence, quickly. However, since it is the "king of the hill" for 486 motherboards, it remains a good choice for those who want to upgrade their

486 systems without replacing the motherboard. The chips are still out there, though they are getting much more difficult to find now.

General Information	Manufacturer	AMD
	Family Name	80486DX5, 5x86
	Code name	"X5"
	Processor Generation	Fourth
	Motherboard Generation	Fourth
	Version	80486DX5-133
	Introduced	1995?
	Variants and Licensed Equivalents	--
Speed Specifications	Memory Bus Speed (MHz)	33
	Processor Clock Multiplier	4.0
	Processor Speed (MHz)	133
	"P" Rating	P75
Benchmarks	iCOMP Rating	~610
	iCOMP 2.0 Rating	~67
	Norton SI	288
	Norton SI32	18
	CPUmark32	~160
Physical Characteristics	Process Technology	CMOS

	Circuit Size (microns)	0.35
	Die Size (mm^2)	--
	Transistors (millions)	--
Voltage, Power and Cooling	External or I/O Voltage (V)	3.45
	Internal or Core Voltage (V)	3.45
	Power Management	SMM
	Cooling Requirements	Active heat sink
Packaging	Packaging Style	168-Pin PGA
	Motherboard Interface	Socket 3; or 168-Pin Socket, Socket 1, Socket 2 (with voltage regulator)
External Architecture	Data Bus Width (bits)	32
	Maximum Data Bus Bandwidth (Mbytes/sec)	127.2
	Address Bus Width (bits)	32
	Maximum Addressable Memory	4 GB
	Level 2 Cache Type	Motherboard
	Level 2 Cache Size	Usually 256 KB
	Level 2 Cache Bus Speed	Same as Memory Bus

	Multiprocessing	No
Internal Architecture	Instruction Set	x86
	MMX Support	No
	Processor Modes	Real, Protected, Virtual Real
	x86 Execution Method	Native
Internal Components	Register Size (bits)	32
	Pipeline Depth (stages)	5
	Level 1 Cache Size	16 KB Unified
	Level 1 Cache Mapping	4-Way Set Associative
	Level 1 Cache Write Policy	Write-Through, Write-Back
	Integer Units	1
	Floating Point Unit / Math Coprocessor	Integrated
	Instruction Decoders	1
	Branch Prediction Buffer Size / Accuracy	None
	Write Buffers	None
	Performance Enhancing Features	None

TABLE 5.8 AMD 80486DX5, 5x86 PROCESSOR SUMMARY

Cyrix 5x86 ("M1sc")

Despite having the same name as AMD's 5x86 processor, the Cyrix 5x86 is a totally different animal. While AMD designed its 5x86 by further increasing the clock on the 486DX4, Cyrix took the opposite approach by modifying its M1 processor core (used for the 6x86 processor) to make a "lite" version to work on 486 motherboards. As such, the Cyrix 5x86 in some ways resembles a Pentium OverDrive (which is a Pentium core modified to work in a 486 motherboard) internally more than it resembles the AMD 5x86. This chip is probably the hardest to classify as either fourth or fifth generation.

The 5x86 employs several architectural features that are normally found only in fifth-generation designs. The pipeline is extended to six stages, and the internal architecture is 64 bits wide. It has a larger (16 KB) primary cache than the 486DX4 chip. It uses branch prediction to improve performance.

The 5x86 was available in two speeds, 100 and 120 MHz. The 5x86-120 is the most powerful chip that will run in a 486 motherboard--it offers performance comparable to a Pentium 90 or 100. The 5x86 is still a clock-tripled design, so it runs in 33 and 40 MHz motherboards. (The 100 MHz version will actually run at 50x2 as well, but normally was run at 33 MHz.) It is a 3 volt design and is intended for a Socket 3 motherboard. It will run in an earlier 486 socket if a voltage regulator is used. I have heard that some motherboards will not run this chip properly so you may need to check with Cyrix if trying to use this chip in an older board. These chips have been discontinued by Cyrix but are still good performers, and for those with a compatible motherboard, as good as you can get. Unfortunately, they are extremely difficult to find now.

General Information	Manufacturer	Cyrix	
	Family Name	5x86	
	Code name	"M1sc"	
	Processor Generation	Fourth	
	Motherboard Generation	Fourth	
	Version	5x86-100	5x86-120

	Introduced	1996?	
	Variants and Licensed Equivalents	--	
Speed Specifications	Memory Bus Speed (MHz)	33 / 50	40
	Processor Clock Multiplier	3.0 / 2.0	3.0
	Processor Speed (MHz)	100	120
	"P" Rating	P75	P90
Benchmarks	iCOMP Rating	~610	~735
	iCOMP 2.0 Rating	~67	~81
	Norton SI	264	316
	Norton SI32	~16	19
	CPUmark32	~150	~180
Physical Characteristics	Process Technology	CMOS	
	Circuit Size (microns)	0.65	
	Die Size (mm^2)	144	
	Transistors (millions)	2.0	
Voltage, Power and Cooling	External or I/O Voltage (V)	3.45	
	Internal or Core Voltage (V)	3.45	
	Power Management	SMM	
	Cooling Requirements	Active heat sink	
Packaging	Packaging Style	168-Pin PGA	
	Motherboard Interface	Socket 3; or 168-Pin Socket, Socket 1, Socket 2 (with voltage regulator)	

External Architecture	Data Bus Width (bits)	32	
	Maximum Data Bus Bandwidth (Mbytes/sec)	127.2	152.6
	Address Bus Width (bits)	32	
	Maximum Addressable Memory	4 GB	
	Level 2 Cache Type	Motherboard	
	Level 2 Cache Size	Usually 256 KB	
	Level 2 Cache Bus Speed	Same as Memory Bus	
	Multiprocessing	No	
Internal Architecture	Instruction Set	x86	
	MMX Support	No	
	Processor Modes	Real, Protected, Virtual Real	
	x86 Execution Method	Native	
Internal Components	Register Size (bits)	32	
	Pipeline Depth (stages)	6	
	Level 1 Cache Size	16 KB Unified	
	Level 1 Cache Mapping	4-Way Set Associative	
	Level 1 Cache Write Policy	Write-Through, Write-Back	
	Integer Units	1	
	Floating Point Unit / Math Coprocessor	Integrated	
	Instruction Decoders	1	
	Branch Prediction Buffer Size / Accuracy	-- entries / -- %	

	Write Buffers	--
	Performance Enhancing Features	--

TABLE 5.9 CYRIX 5x86 PROCESSOR SUMMARY

VESA Local Bus (VLB)

The first local bus to gain popularity, the VESA local bus (also called VL-Bus or VLB for short) was introduced in 1992. VESA stands for the Video Electronics Standards Association, a standards group that was formed in the late eighties to address video-related issues in personal computers. Indeed, the major reason for the development of VLB was to improve video performance in PCs.

The VLB is a 32-bit bus which is in a way a direct extension of the 486 processor/memory bus. A VLB slot is a 16-bit ISA slot with third and fourth slot connectors added on the end. The VLB normally runs at 33 MHz, although higher speeds are possible on some systems. Since it is an extension of the ISA bus, an ISA card can be used in a VLB slot, although it makes sense to use the regular ISA slots first and leave the (small number of) VLB slots open for VLB cards, which won't work in an ISA slot of course. Use of a VLB video card and I/O controller greatly increases system performance over an ISA-only system.

While VLB was extremely popular during the reign of the 486, with the introduction of the Pentium and its PCI local bus in 1994, wholesale abandonment of the VLB began in earnest. While Intel pushing PCI was one reason why this happened, there were also several key problems with the VLB implementation. First, the design was strongly based on the 486 processor, and adapting it to the Pentium caused a host of compatibility and other problems. Second, the bus itself was tricky electrically; for example, the number of cards that could be used on the bus was low (often only two or even one), and occasionally there could be timing problems on the bus when more than one card was used. Finally, the bus did not support bus mastering properly since there was no good arbitration scheme, and did not support Plug and Play.

Today VLB is obsolete for new systems; even the latest 486 motherboards use PCI, and all Pentiums and higher use PCI. However, these systems do still offer reasonable performance, and are now plentiful and very inexpensive--if you can still find them.

5.5 Celeron Motherboard

Pentium III and Celeron processors have not had any interesting developments of late. Rumors have made the rounds that the 100MHz FSB Celerons will debut soon, which could spark a bit of interest, but they will still not match the performance of the AMD Duron (which, however, has infrastructure issues that must be addressed). On the other hand, the Celerons are very popular in the mobile market, because they compete very well against the K6-2 - which is the only mobile offering from AMD at the moment.

Also it is using EISA and PCI technology so please refer appropriate section for bus architecture 5.3 and 5.6.

5.6 Pentium Motherboard

The fifth generation of processors saw several changes from earlier CPU families, and several trends continue as well. Chips continued to get faster and faster, and architectural changes were made to increase overall system speed as well. AMD and Cyrix developed their own compatible processors instead of just trying to clone Intel's, leading to more variety and choice in the marketplace. The Pentium and the compatibles that followed it opened up the world of computers for millions of users and propelled computing to the next level.

Intel Pentium ("P5" / "P54C")

Intel's new fifth-generation chip was expected to be called the 586, following their earlier naming conventions. However, with the rise of AMD and Cyrix, Intel wanted to be able to register as a trademark the name of their new CPU, and numbers can't be trademarked. Thus, the Pentium was born. It is now one of the most recognized trademarks in the computer world, one reason why Intel doesn't seem to ever want to make another processor whose name doesn't have "Pentium" in it somewhere.

The Pentium is the defining processor of the fifth generation. It has in fact had several generations itself; the first Pentiums are different in many ways from the latest ones. It

has been the target for compatibility for AMD's K5 and Cyrix's 6x86 chips, as well as generations that have followed. The chip itself is instruction set compatible with earlier x86 CPUs, although it does include a few new (rarely used) instructions.

The Pentium provides greatly increased performance over the 486 chips that precede it, due to several architectural changes. Roughly speaking, a Pentium chip is double the speed of a 486 chip of the same clock speed. In addition, the Pentium goes to much higher clock speeds than the 486 ever did. The following are the key architectural enhancements made in the Pentium over the 486-class chips (note that some of these are present in Cyrix's 5x86 processor, but that chip was developed after the Pentium):

- Superscalar Architecture: The Pentium is the first superscalar processor; it uses two parallel execution units. Some people have likened the Pentium to being a pair of 486s in the same chip for this reason, though this really isn't totally accurate. It is really only partially superscalar because the second execution unit doesn't have all the capabilities of the first; some instructions won't run in the second pipeline. In order to take advantage of the dual pipelines, code must be optimized to arrange the instructions in a way that will let both pipelines run at the same time. This is why you sometimes see reference to "Pentium optimization". Regardless, the performance is much higher than the single pipeline of the 486.
- Wider Data Bus: The Pentium's data bus is doubled to 64 bits, providing double the bandwidth for transfers to and from memory.
- Much Faster Memory Bus: Most Pentiums run on 60 or 66 MHz system buses; most 486s run on 33 MHz system buses. This greatly improves performance. Pentium motherboards also incorporate other performance-enhancing features, such as pipelined burst cache. The Pentium processor was also the first specifically designed to work with the (then new) PCI bus.
- Branch Prediction: The Pentium uses branch prediction to prevent pipeline stalls when branches are encountered.
- Integrated Power Management: All Pentiums have built in SMM power management (optional on most of the 486s).

- Split Level 1 Cache: The Pentium uses a split level 1 cache, 8 KB each for data and instructions. The cache was split so that the data and instruction caches could be individually tuned for their specific use.
- Improved Floating Point Unit: The floating point unit of the Pentium is significantly faster than that of the 486.

The Pentium is available in a wide variety of speeds, and in regular and OverDrive versions. It is also available in several packaging styles, although the pin grid array (PGA) is still the most prevalent. The original Pentiums, the 60 and 66 MHz versions, were very different than the later versions that are used in most PCs; they used older, 5 volt technology and significant problems with heat. Intel solved this with later (75-200 MHz) versions by going to a smaller circuit size and 3.3 volt power.

Pentiums use three different sockets. The original Pentium 60 and 66 use Socket 4. Pentiums from 75 to 133 will fit in either socket 5 or socket 7; Pentium 150s, 166s and 200s require Socket 7. Intel makes Pentium OverDrives that allow the use of faster Pentiums in older Pentium sockets (in addition to OverDrives that go in 486 motherboards).

The Pentium processor achieved a certain level of "fame" as a result of the bug that was discovered in its floating point unit not long after it was released. This is commonly known as the "FDIV" bug after the instruction (floating point divide) that it most commonly turns up in. While bugs in processors are relatively common, they usually are minor and don't have a direct impact on computation results. This one did, and achieved great notoriety in part because Intel didn't own up to the problem and offer to correct it immediately. Intel does offer a replacement on affected processors, which were only found in early versions (60 to 100) sold in 1994 and earlier.

If you suspect your Pentium of having the FDIV bug, try this computation test using a spreadsheet or calculator program: take the number 4,195,835 and divide it by 3,145,727. Then take the result and multiply it by the same number again (3,145,727). You should of course get the same 4,195,835 back that you started with. On a PC with the FDIV bug you will get 4,195,579 (an error of 256), but beware that some operating systems and applications have been patched to compensate for this bug, so a simple math test isn't

necessarily conclusive. Try looking at this page on Intel's web site for replacement information, if you suspect that you have an FDIV bug on your older Pentium chip.

For many years, the Pentium processor was the mainstream processor of choice, but finally the Pentium with MMX has driven it to the economy market. With the regular Pentium maxing out at 200 MHz and the Pentium with MMX 166 dropping well below $200, the "Pentium Classic" doesn't make nearly as much sense as it used to for new PCs. The 60 and 66 are obsolete due to their slow speed and older technology, and the 75 to 150 are obsolete because their performance is much lower than the 166 and 200, for almost the same amount of money.

The entire classic Pentium line is now technically obsolete, due to the availability of inexpensive, faster Pentium with MMX chips (as well as comparable offerings from AMD and Cyrix). The non-MMX Pentium is no longer generally used in new systems. However, since the Pentium with MMX requires split rail voltage, the classic Pentium 200 remains a great chip for those who have socket 7 motherboards and want to upgrade, but who do not have split rail voltage support.

General Information	Manufacturer	Intel									
	Family Name	Pentium									
	Code name	"P5"		"P54C"							
	Processor Generation	Fifth									
	Motherboard Generation	Fifth									
	Version	P60	P66	P75	P90	P100	P120	P133	P150	P166	P200
	Introduced	Mar 93		Oct. 94	Mar 94		Mar 95	June 95	Jan. 96	Jan. 96	June 96
	Variants and Licensed Equivalents	--									

Speed Specifications	Memory Bus Speed (MHz)	60	66	50	60	66	60	66	60	66	66
	Processor Clock Multiplier	1.0		1.5			2.0		2.5		3.0
	Processor Speed (MHz)	60	66	75	90	100	120	133	150	166	200
	"P" Rating	60	66	75	90	100	120	133	150	166	200
Benchmarks	iCOMP Rating	510	567	610	735	815	1000	1110	1176	1308	~1575
	iCOMP 2.0 Rating	51	57	67	81	90	100	111	114	127	142
	Norton SI	190	211	237	285	317	380	421	476	529	637
	Norton SI32	~16	~18	23	27	30	32	36	35	40	44
	CPUmark32	~120	~140	181	219	243	270	300	308	343	382
Physical Characteristics	Process Technology	Bipolar CMOS									
	Circuit Size (microns)	0.8		0.6			0.6 / 0.35		0.35		
	Die Size (mm^2)	295		147			147 / 90		90		
	Transistors (millions)	3.1		3.2					3.3		
Voltage, Power and Cooling	External or I/O Voltage (V)	5		3.3 (STD) / 3.52 (VRE)							
	Internal or Core Voltage (V)	5		3.3 (STD) / 3.52 (VRE)							

	Power Management	SMM									
	Cooling Requirements	Passive or active heat sink									
Packaging	Packaging Style	273-Pin PGA	296-Pin SPGA								
	Motherboard Interface	Socket 4	Socket 5, Socket 7					Socket 7			
External Architecture	Data Bus Width (bits)	64									
	Maximum Data Bus Bandwidth (Mbytes/sec)	457.8	508.6	381.5	457.8	508.6	457.8	508.6	457.8	508.6	508.6
	Address Bus Width (bits)	32									
	Maximum Addressable Memory	4 GB									
	Level 2 Cache Type	Motherboard									
	Level 2 Cache Size	Usually 256 KB - 512 KB									
	Level 2 Cache Bus Speed	Same as Memory Bus									
	Multiprocessing	Dual (SMP) with Compatible Motherboard									
Internal Architecture	Instruction Set	x86 plus Pentium Extensions									
	MMX Support	No									
	Processor Modes	Real, Protected, Virtual Real									
	x86 Execution Method	Native									
Internal	Register Size	32									

Components	(bits)	
	Pipeline Depth (stages)	5
	Level 1 Cache Size	8 KB Data, 8 KB Instruction
	Level 1 Cache Mapping	2-Way Set Associative
	Level 1 Cache Write Policy	Write-Through (Data and Instruction), Write-Back (Data Only)
	Integer Units	2
	Floating Point Unit / Math Coprocessor	Integrated
	Instruction Decoders	1
	Branch Prediction Buffer Size / Accuracy	256 entries / 80%
	Write Buffers	2
	Performance Enhancing Features	--

TABLE 5.10 PENTIUM PROCESSOR SUMMARY

Intel Pentium OverDrive

In order to allow the easy processor upgrades of earlier motherboards, Intel has made available Pentium-class OverDrive chips. These are internally Pentium chips, but have subtle changes made to them to accommodate the unusual sockets and motherboards they are designed to work in. This allows them to be used where a regular Pentium would not work, which is the whole idea behind the OverDrive product--their interfaces appear to the motherboard like an older chip, so you don't need to worry about whether or not the motherboard supports the higher speed of what would be a native chip, in most cases.

Note: In general, if your motherboard can support a higher-speed regular (non-OverDrive) Pentium, that is the better way to go as opposed to an upgrade, because the regular chips are cheaper than the OverDrives. Your motherboard manual will tell you what your system's limitations are.

OverDrive processors have several changes made so that they can be used in older machines. First, when necessary they include integrated voltage regulators (sandwiched between the chip and the heat sink) so they will work with the voltage of the socket they are intended to go into. Second, they of course have the right pin configuration for the socket. Third, they are hard-wired to a specific clock multiplier; they do not set their multiplier based on motherboard jumpers like regular Pentiums do. This saves the user from worrying about what jumpers to change when replacing the processor.

The Pentium OverDrive comes in three basic flavors:

- OverDrive for 486 Systems: This Pentium OverDrive is specially modified to fit the 32-bit data bus of a 486 system. As you know a Pentium normally uses a 64-bit data bus, but this is not consistent with a 486 motherboard's design. Therefore, this OverDrive is modified to use an external 32 bit bus. This reduces performance compared to a real Pentium, which is one reason why these chips score below what would be expected for a real Pentium of the same speed. Interestingly, this chip has a 32 KB primary cache, double the size of regular Pentiums. Presumably this was done to help mitigate the effects of this chip running on a 32-bit motherboard. (It's still slow, and in fact, slower than the top-end 486 and 5x86 chips that run in the same motherboards and cost much less). The Pentium OverDrive for 486s is available in two speeds: 63 MHz for 25 MHz bus systems and 83 MHz for 33 MHz systems. They obviously use a clock multiplier of 2.5. These chips are designed to work in a 5 volt system, and go into a Socket 2 or Socket 3 (the socket 3 must be set on 5 volts).
- OverDrive for Pentium 60/66: The original Pentium chips were different than later versions, in terms of voltage, socket size and power consumption. Intel created a clock-doubling OverDrive for these chips, which is sold as one "120/133" chip: when replacing a Pentium 60 it runs at 120 MHz, and when replacing a 66 it runs at 133. This is a true Pentium chip since it is on a Pentium

motherboard, although it still benchmarks below the real Pentium 120/133, most likely due to the older design of the Pentium 60/66 motherboards.

- OverDrives for the Pentium 75, 90 and 100: These three chips run with a multiplier of 1.5 on system buses of 50, 60 and 66 MHz respectively. Intel has made for these OverDrives running at 125, 150 and 166 MHz (clock multiplier of 2.5). The 125 is an oddity because Intel never made a Pentium 125 as a stand-alone processor. Note that these three OverDrives have been replaced by versions of the Pentium with MMX OverDrive running at the same speed.

Except where noted above, the Pentium OverDrive has the same design and internal functioning as the regular Pentium. Refer to the section on the Pentium for more description of the Pentium's functions and features, and improvements over 486-class chips.

For many systems, OverDrive processors are the simplest way to get a performance increase at a reasonable price. Since those who need to use OverDrives are somewhat of a "captive market", Intel has traditionally priced OverDrives significantly higher than equivalent regular Pentiums. In some cases, the price difference can be exorbitant, with OverDrives selling for double what the street price is for the regular chip. The best time to buy OverDrives is after they have been on the market for a while and the prices have begun to decrease. Still, an upgrade of the motherboard and processor is often less expensive than an OverDrive processor, and yields better performance (at the cost of more work and risk, of course).

It's important to remember that increasing the performance of the processor is only part of the solution to increasing overall system performance. Many other factors impact on system performance, and increasing the processor clock while leaving the system bus speed the same is an exercise in diminishing returns, because the processor is increasingly stuck waiting for the rest of the system.

Note: Many older 486 motherboards do not support write-back primary cache. They will run the Pentium OverDrive slower than those that do.

General Information	Manufacturer	Intel					
	Family Name	Pentium OverDrive					
	Code name	P24T		--			
	Processor Generation	Fifth					
	Motherboard Generation	Fourth		Fifth			
	Version	Pentium Over Drive 63 for 486	Pentium Over Drive 83 for 486	Pentium Over Drive 120/133	Pentium Over Drive 125	Pentium Over Drive 150	Pentium Over Drive 166
	Introduced	--					
	Variants and Licensed Equivalents	--					
Speed Specifications	Memory Bus Speed (MHz)	25	33	60 / 66	50	60	66
	Processor Clock Multiplier	2.5		2.0	2.5		
	Processor Speed (MHz)	63	83	120 / 133	125	150	166
	"P" Rating	~50	~66	~90 / ~100	125	150	166
Benchmarks	iCOMP Rating	443	581	877 / 970	1070	1176	1308
	iCOMP 2.0 Rating	~42	~57	75 / 84	~105	114	127
	Norton SI	~180	~240	~340 /	~410	476	529

				~370			
	Norton SI32	~15	~19	~27 / ~30	~32	35	40
	CPUmark32	~110	~140	~210 / ~235	~260	308	343
Physical Characteristics	Process Technology	Bipolar CMOS					
	Circuit Size (microns)	0.6		0.35			
	Die Size (mm^2)	--		90			
	Transistors (millions)	--		3.3			
Voltage, Power and Cooling	External or I/O Voltage (V)	5			3.3 (STD) / 3.52 (VRE)		
	Internal or Core Voltage (V)	3.3 (STD) / 3.52 (VRE)					
	Power Management	SMM					
	Cooling Requirements	Active heat sink (included)					
Packaging	Packaging Style	168-Pin PGA		273-Pin PGA	296-Pin SPGA		
	Motherboard Interface	Socket 2, Socket 3		Socket 4	Socket 5, Socket 7		
External Architecture	Data Bus Width (bits)	32		64			
	Maximum Data Bus Bandwidth (Mbytes/sec)	95.4	127.2	457.8 / 508.6	381.5	457.8	508.6
	Address Bus Width (bits)	32					

	Maximum Addressable Memory	4 GB	
	Level 2 Cache Type	Motherboard	
	Level 2 Cache Size	Varies	Usually 256 KB - 512 KB
	Level 2 Cache Bus Speed	Same as Memory Bus	
	Multiprocessing	No	
Internal Architecture	Instruction Set	x86 plus Pentium Extensions	
	MMX Support	No	
	Processor Modes	Real, Protected, Virtual Real	
	x86 Execution Method	Native	
Internal Components	Register Size (bits)	32	
	Pipeline Depth (stages)	5	
	Level 1 Cache Size	16 KB Data, 16 KB Instruction	8 KB Data, 8 KB Instruction
	Level 1 Cache Mapping	2-Way Set Associative	
	Level 1 Cache Write Policy	Write-Through (Data and Instruction), Write-Back (Data Only)	
	Integer Units	2	
	Floating Point Unit / Math Coprocessor	Integrated	
	Instruction	1	

	Decoders	
	Branch Prediction Buffer Size / Accuracy	256 entries / 80%
	Write Buffers	2
	Performance Enhancing Features	--

TABLE 5.11 PENTIUM OVERDRIVE PROCESSOR SUMMARY

Intel Pentium with MMX Technology ("P55C")

In January of 1997 Intel released its newest, and presumably last, fifth generation processor, the Intel Pentium with MMX Technology. It is an evolutionary design, adding some enhancements and new capabilities to the "classic" Pentium. In most ways however, it is the same chip.

The chief (well-publicized) difference between the Pentium with MMX and the Pentium Classic is the inclusion of the MMX instruction set extensions. Running software that is designed for MMX on a Pentium MMX will result in a significant increase in performance over the regular Pentium; some claim up to 70%. Even running non-MMX software on the MMX chip results in an improvement of about 20% over a regular Pentium of the same clock speed, due to the following other improvements over the older chip:

- Doubled Primary Cache: The Pentium with MMX has 16 KB for each of the level 1 data and instruction caches, as opposed to 8 KB each for the regular Pentium.
- Improved Cache Mapping: The primary cache is now 4-way set associative instead of 2-way.
- Deepening of Internal Pipelines: Both of the internal integer executions units are increased from 5 to 6 stages.
- Better Use of Internal Pipelines: More types of instructions can be run in parallel down the two execution pipes than on the older Pentium, so more use is made of the second pipe.

- Improved Branch Prediction Unit: The branch prediction unit's accuracy is enhanced over the classic Pentium.
- Improved Instruction Decoder: The instruction decoder is more efficient than the Pentium's.

The Pentium with MMX is pin-compatible with the regular Pentium and goes in a Socket 7 just like the old chip. However, there is one important difference between the two: the Pentium with MMX uses split-rail voltage: 3.3 volts for the external voltage and 2.8 volts for the core. Not all motherboards with a socket 7 support this dual voltage setting, because before the Pentium with MMX came out no processors needed it, so some manufacturers skimped by not planning for the future. Intel is making Overdrives for motherboards that don't support the lower core voltage.

Warning: Some unscrupulous vendors are misleading buyers by telling them that the Pentium with MMX will run with 3.3 volts applied to both external and internal voltage. In many cases this will not instantly fry the chip; it may even work for a little while. Eventually however it will fail and you could void the warranty on the chip.

The MMX capability is implemented on the Pentium with MMX by sharing the registers used by the floating point unit. This has led to silly rumors that using MMX would cause a "major" performance penalty when switching between applications using MMX and those using floating point applications. This is a myth; aside from the fact that very few applications make extensive use of the floating point unit, a switch between MMX and non-MMX applications takes no more time than a switch between floating point and non-floating-point applications. And the amount of time for a switch is less than 100 cycles on a processor that is usually running at 200,000,000 cycles per second or more.

Originally, the Pentium with MMX was supposed to be available only in 166 and 200 MHz versions, after which Intel was going to stop production on Socket 7 chips entirely. Intel surprised the PC world a bit a few months after the introduction of the original 166 and 200 chips by announcing a Pentium with MMX 233, running at 66 MHz and 3.5x multiplier. They now say that this will be the last Pentium, but who knows.

Since socket 7 boards don't support a 3.5x multiplier, Intel made the chip respond to a 1.5x signal as if it were 3.5x. This is becoming a common trick, which I believe AMD used first in creating its 5x86-133 (which runs in 486 motherboards as 4x when configured as 2x). So to set up a Pentium with MMX 233, you jumper its system bus speed and multiplier (but not core voltage!) as if it were a Pentium 100.

Overall, the Pentium with MMX is a very good chip, and with Intel aggressively cutting prices it has become pretty much the mainstream processor of choice over the last year or so. In recent months, however, with new sixth-generation chips by AMD (the K6) and Cyrix (the 6x86MX) that will run in fifth-generation motherboards becoming commonplace, the high-end performance picture on fifth-generation motherboards has become much more cloudy. For its part, Intel has moved on to the Pentium II, and as a result the Pentium with MMX is being used on fewer and fewer new systems, in favor of the newer chip.

General Information	Manufacturer	Intel		
	Family Name	Pentium with MMX Technology		
	Code name	"P55C"		
	Processor Generation	Fifth		
	Motherboard Generation	Fifth		
	Version	Pentium with MMX 166	Pentium with MMX 200	Pentium with MMX 233
	Introduced	Jan. 1997		June 1997
	Variants and Licensed Equivalents	--		
Speed Specifications	Memory Bus Speed (MHz)	66		
	Processor Clock Multiplier	2.5	3.0	3.5 (jumper as 1.5)

	Processor Speed (MHz)	166	200	233
	"P" Rating	~200	~240	~280
Benchmarks	iCOMP Rating	~1575	~1900	~2210
	iCOMP 2.0 Rating	160	182	203
	Norton SI	~640	~770	~890
	Norton SI32	50.7	57.5	62
	CPUmark32	378	425	460
Physical Characteristics	Process Technology	CMOS		
	Circuit Size (microns)	0.35		
	Die Size (mm^2)	141		
	Transistors (millions)	4.5		
Voltage, Power and Cooling	External or I/O Voltage (V)	3.3		
	Internal or Core Voltage (V)	2.8		
	Power Management	SMM		
	Cooling Requirements	Passive or active heat sink		
Packaging	Packaging Style	296-Pin SPGA		
	Motherboard Interface	Socket 7 with 2.8V Core		
External Architecture	Data Bus Width (bits)	64		
	Maximum Data Bus Bandwidth (Mbytes/sec)	508.6		
	Address Bus Width (bits)	32		
	Maximum Addressable	4 GB		

	Memory	
	Level 2 Cache Type	Motherboard
	Level 2 Cache Size	Usually 256 KB - 512 KB
	Level 2 Cache Bus Speed	Same as Memory Bus
	Multiprocessing	Dual (SMP) with Compatible Motherboard
Internal Architecture	Instruction Set	x86 plus Pentium Extensions
	MMX Support	Yes
	Processor Modes	Real, Protected, Virtual Real
	x86 Execution Method	Native
Internal Components	Register Size (bits)	32
	Pipeline Depth (stages)	6
	Level 1 Cache Size	16 KB Data, 16 KB Instruction
	Level 1 Cache Mapping	4-Way Set Associative
	Level 1 Cache Write Policy	Write-Through (Data and Instruction), Write-Back (Data Only)
	Integer Units	4 (2 for MMX)
	Floating Point Unit / Math Coprocessor	Integrated
	Instruction Decoders	1
	Branch Prediction Buffer Size / Accuracy	512 entries / 90%
	Write Buffers	4
	Performance Enhancing Features	--

TABLE 5.12 PENTIUM WITH MMX TECHNOLOGY PROCESSOR SUMMARY

Intel Pentium with MMX Technology OverDrive

With the introduction of the Intel Pentium with MMX Technology, Intel also created OverDrive processors to upgrade existing Pentium motherboards to the new MMX chip. Most older Pentium motherboards cannot handle the new Pentium with MMX because of its requirement for a 2.8V core. Keeping with the tradition of Intel's OverDrive line, the Pentium with MMX OverDrive includes a converter that lets it run in Socket 5 motherboards (except for the 200) and Socket 7s that do not have 2.8V support. Otherwise the chip is identical to the standard Pentium with MMX. Refer to the section on the Pentium with MMX for more description of its improvements over the Pentium. See the section on the Pentium Classic OverDrive for more on using OverDrive processors.

Note: In general, if your motherboard can support the MMX chip directly, it is almost always better to buy a real Pentium with MMX as opposed to an OverDrive, because they are typically less expensive. Your motherboard manual will tell you what your system's limitations are, but generally if your Pentium motherboard supports split-rail voltage, you should be able to run a standard Pentium with MMX.

The initial Pentium with MMX OverDrives run at 125 MHz, 150 MHz and 166 MHz, for use in Pentium 75 (1.5x50), Pentium 90 (1.5x60) and Pentium 100 (1.5x66) systems respectively. These chips replaced identical non-MMX OverDrives which Intel quickly phased out when MMX hit the market. In each case, the chip has a fixed clock multiplier of 2.5, intended for use in place of the older 1.5 multiplier chips on these three system speeds.

Intel later introduced two new Pentium with MMX OverDrives, running at 180 MHz and 200 MHz. These chips run with a 3x clock multiplier; the 180 is intended for upgrading 90, 120 and 150 MHz systems running at 60 MHz system bus speeds, and the 200 for upgrading 100, 133 and 166 systems running with 66 MHz system buses. No 3x clock multiplier OverDrive was created specifically for 50 MHz Pentium 75 systems, but the 180 will function in Pentium 75 systems, running at 150 MHz. (Note that the older 150 MHz OverDrive has a 2.5x multiplier and will only run at 125 in a Pentium 75 PC).

Note: Unlike the other OverDrives, which will work in either socket 5 or socket 7 motherboards, the Pentium with MMX OverDrive 200 is intended for use only in socket 7 systems.

These OverDrives provide significant performance improvements over the original chips and the ability to execute MMX software as well. It's still important to remember that increasing the performance of the processor is only part of the solution to increasing overall system performance. Many other factors impact on system performance, and increasing the processor clock while leaving the system bus speed the same is an exercise in diminishing returns, because the processor is increasingly stuck waiting for the rest of the system.

General Information	Manufacturer	Intel				
	Family Name	Pentium With MMX Technology OverDrive				
	Code name	"P54CTB"				
	Processor Generation	Fifth				
	Motherboard Generation	Fifth				
	Version	Pentium with MMX OverDrive 125	Pentium with MMX OverDrive 150	Pentium with MMX OverDrive 166	Pentium with MMX OverDrive 180	Pentium with MMX OverDrive 200
	Introduced	Jan. 1997			Mid 1997	
	Variants and Licensed Equivalents	--				
Speed Specifications	Memory Bus Speed (MHz)	50	60	66	60	66
	Processor Clock Multiplier	2.5			3.0	
	Processor	125	150	166	180	200

	Speed (MHz)					
	"P" Rating	~150	~180	~200	~215	~240
Benchmarks	iCOMP Rating	~1200	~1400	~1575	~1700	~1900
	iCOMP 2.0 Rating	120	144	160	~167	182
	Norton SI	~480	~570	~640	~670	~770
	Norton SI32	~36	~46	50.7	~53	57.5
	CPUmark32	~290	340	378	~390	425
Physical Characteristics	Process Technology	CMOS				
	Circuit Size (microns)	0.35				
	Die Size (mm^2)	141				
	Transistors (millions)	4.5				
Voltage, Power and Cooling	External or I/O Voltage (V)	3.3				
	Internal or Core Voltage (V)	2.8 (3.3 from Motherboard)				
	Power Management	SMM				
	Cooling Requirements	Active heat sink (included)				
Packaging	Packaging Style	296-Pin SPGA				
	Motherboard Interface	Socket 5, Socket 7				Socket 7
External Architecture	Data Bus Width (bits)	64				
	Maximum Data	381.5	457.8	508.6	457.8	508.6

	Bus Bandwidth (Mbytes/sec)					
	Address Bus Width (bits)	32				
	Maximum Addressable Memory	4 GB				
	Level 2 Cache Type	Motherboard				
	Level 2 Cache Size	Usually 256 KB - 512 KB				
	Level 2 Cache Bus Speed	Same as Memory Bus				
	Multiprocessing	No				
Internal Architecture	Instruction Set	x86 plus Pentium Extensions				
	MMX Support	Yes				
	Processor Modes	Real, Protected, Virtual Real				
	x86 Execution Method	Native				
Internal Components	Register Size (bits)	32				
	Pipeline Depth (stages)	6				
	Level 1 Cache Size	16 KB Data, 16 KB Instruction				
	Level 1 Cache Mapping	4-Way Set Associative				
	Level 1 Cache Write Policy	Write-Through (Data and Instruction), Write-Back (Data Only)				
	Integer Units	4 (2 for MMX)				

	Floating Point Unit / Math Coprocessor	Integrated
	Instruction Decoders	1
	Branch Prediction Buffer Size / Accuracy	512 entries / 90%
	Write Buffers	4
	Performance Enhancing Features	--

TABLE 5.13 PENTIUM WITH MMX TECHNOLOGY OVERDRIVE TECHNOLOGY PROCESSOR SUMMARY

Cyrix 6x86 ("M1")

Cyrix entered the fifth generation processor market in a big way with its (IMO poorly-named) 6x86 processor, formerly called the M1. This is an advanced processor that is an alternative to the Intel Pentium, and is pin- and voltage-compatible with it. Cyrix apparently gave it the "6x86" moniker in reference to some of its more advanced features, which it calls "sixth generation". In reality, the processor is comparable in power (and in many ways, architecturally as well) to the fifth-generation Pentium. It is, like the Pentium, a superscalar, native x86 execution CPU.

The 6x86, like the AMD K5, is not a Pentium "clone", in the traditional sense of the word. Clones are exact or near-exact copies, usually being reversed engineered or based on licensed code. The 6x86 is based on an original design of Cyrix's and is not 100% Pentium compatible. (It is 100% x86 compatible, but the Pentium uses some extensions to the base x86 instruction set that the 6x86 doesn't support).

The 6x86 incorporates several advanced architectural features that allow it to outperform a Pentium of equal clock speed. For example, a Cyrix running at 133 MHz

will typically equal or outperform a Pentium 166 (not Pentium with MMX 166). For this reason, and because the public seems overly obsessed with thinking that clock speed is all that matters, Cyrix helped invent the "P rating" system. This was a good idea, to help people make more valid comparisons. Unfortunately however, it has caused a great deal of confusion of its own, because Cyrix named the chips with their P rating and not their clock speed. Many users (and even vendors!) confuse the P rating with the clock speed and try to set their motherboards to run, say, a 6x86-PR166+ to run at 166 MHz. This is overclocking the chip by 25% and the usual result is that the chip fails. It is important to make sure that the chip is set to its correct clock speed.

Compared to the Pentium (classic), the Cyrix 6x86 has the following enhancements or improvements that contribute to its greater performance to clock speed ratio (it also has all the advantages over the 486 that are mentioned in the section discussing the original Pentium):

- Deepening of Internal Pipelines: Both of the internal integer executions units are increased from 5 to 7 stages.
- Out of Order Completion: Instructions flowing down the two integer pipelines can complete out of order.
- Superior Branch Prediction Unit: The branch prediction unit of the 6x86 can handle multiple branches and has better performance than the Pentium's.
- Improved Cache Mapping: The primary cache in the 6x86 is 4-way set associative instead of the Pentium's 2-way. (It is also unified instead of being split like the Pentium's, though I don't think this is really an enhancement).
- Register Renaming: This feature improves parallel performance of the pipelines.
- Speculative Execution: The 6x86 uses speculative execution to reduce pipeline stall time.
- Of course the 6x86 is also cheaper than a Pentium of the same performance rating. Often, it is less than half the cost. This is in fact the main reason that the 6x86 is purchased instead of the Pentium. Compared to the Pentium (classic), the 6x86 has the following disadvantages:

- Software Compatibility Problems: This is probably the biggest problem with the 6x86 and one reason that many people won't use it, even though it provides excellent value for many applications. Some rather parochial software has been written by programmers who think Intel is the only company making CPUs, and therefore they have used Pentium-specific instructions (recall that the Pentium has extensions to the standard x86 instruction set). This causes problems with the 6x86 because the 6x86 is not 100% Pentium compatible. Some companies have made "patches" available to allow their software to run on the 6x86. This problem is most pronounced with games and high-end graphical utilities that have been heavily optimized for the Pentium. One of the most publicized problems with the 6x86 is that its early versions had a bug that causes Windows NT to run more slowly than it should have. Cyrix has addressed this issue and is offering chip replacements to affected users. New systems shouldn't have this problem.

- Processor Identification Problems: In addition to programs that won't work with the 6x86, there are those that would work with the 6x86 if only they knew what it was. There is a special instruction called "CPUID" that is used to identify the make and model of the processor. Some programs use this to detect if the computer is running on a Pentium, and will refuse to run (or run in a reduced mode) if they don't find one. These tests were originally used by software requiring a lot of power, to distinguish between Pentiums and 486s (which the software would be too slow to run on). The 6x86 by default does not respond to the "CPUID" instruction so some programs think it is a 486. Again, this problem is most pronounced with games. Also, early versions of Windows 95 will not identify the 6x86 properly since they predate it--they think it is a 486. This doesn't cause any performance problems however, other than programs (games) running under Windows 95 looking at the information that Windows 95 detects to decide if the machine is a Pentium or not . This problem is also fixed with the OEM SR2 release, which detects the 6x86 properly. Various utilities are available to "trick" the system into identifying the chip as a Pentium, if needed. This can solve many of these problems, although not all of them. See this page for more information.

- Motherboard Compatibility Problems: Some motherboards (especially cheapo ones) have quirks that cause them not to work correctly with the 6x86. Cyrix maintains an approved motherboard list to help 6x86 buyers choose boards that are compatible with the chip. There are some boards that will work with it despite not being on the list, but the ones on it are known to work.
- Heat and Power Problems: The 6x86 uses significantly more power and generates more heat than the Pentium. This causes problems in some boxes and with some motherboards. Later versions address this issue and the low-power 6x86L eliminates it almost completely (but requires a motherboard with 2.8 volt core power). The best way to avoid this problem is to use an approved motherboard (as mentioned just above) since often the heat problems are caused by motherboards with insufficient voltage regulators. You should also use an active heat sink specifically meant for use on the 6x86; special cooling units are designed for the 6x86 and cool better than generic Pentium fans.
- No Multi-Processor Support: The implementation of SMP (symmetric multiprocessing) supported by today's motherboards is customized to Intel processors and will not work with the 6x86.
- Lower-Power Floating Point Unit: While the 6x86 equals or exceeds a Pentium of its "P rating" at integer operations, it lags far behind in floating-point performance. Cyrix intentionally decided not to spend time optimizing the FPU due to its lack of importance for most applications (which is true). A 6x86 generally has floating point performance slightly below a Pentium of its clock speed (as opposed to P rating). So a 6x86-PR166+ has integer performance equal to a Pentium 166 or higher, but floating point performance more comparable to a Pentium 120 or even 100. The most famous problem this causes is with Quake, the popular game that uses floating point instructions. Most games actually do not use the FPU much at all.

Many of the compatibility problems with the 6x86 are a result of the fact that most software was written when Intel was the "only game in town". With AMD and

Cyrix establishing themselves as serious players, it is likely that in the future programmers will be more comprehensive in their choice of coding methods. However, the 6x86 really isn't totally Pentium compatible, which the buyer needs to take into consideration.

The 6x86 chip is designed for Socket 7, and is available in several different clock speeds, some of them rather unusual. All of the chips use a 2x multiplier, and they support bus speeds of 50, 55, 60, 66 and 75 MHz. Of course only 3 of those will be readily recognizable to those familiar with Pentium processors. The 55 MHz bus speed of the PR133+ is in particular an oddball; many (if not most) motherboards will not support this speed.

The 75 MHz bus speed is an interesting advance in many ways, as it pushes the traditional 66 MHz memory speed limit forward. The system bus speed is for most systems a bottleneck that severely limits overall performance; increasing it can increase performance greatly. This in fact is one major reason that the 150 MHz 6x86-PR200+ performs at the level of a Pentium 200. However, there are several additional caveats associated with the 75 MHz bus speed used by the 6x86-PR200+:

- Chipset Support: Intel is the major producer of Pentium-class chipsets, which are rated to 66 MHz, the fastest memory bus speed that Intel processors use. Not surprisingly :^) Intel is not in a big hurry to support a higher bus speed used only by a Cyrix chip. This leaves two choices for motherboard manufacturers: overclock an Intel chipset (which some do) or use an alternative.
- Motherboard Support: Largely because of the chipset problem, motherboards that support the 75 MHz bus speed of the PR200+ are harder to find than regular motherboards.
- PCI Device Overclocking: Since the PCI bus runs at half of the memory bus speed, when you increase the memory bus to 75 MHz you are increasing the PCI bus to 37.5 MHz. When you do this, you are effectively overclocking all of your PCI cards by 12.5%. This is not usually a problem, but you should bear in mind that *many* peripheral makers do not do extensive testing at the higher bus speed

because of the very small market of the Cyrix chips. This will hopefully change in the future as higher bus speeds become more prevalent.

The Cyrix 6x86 is manufactured by IBM, which in fact also sells the 6x86 under its own name. (Cyrix doesn't have its own manufacturing facilities). These chips are identical; they are manufactured in the same IBM fabrication plant, and independently packaged and tested. The rumor is that the IBM version is superior due to better testing, but I've found nothing reliable to prove this.

Responding to complaints about the heat and power problems of the 6x86, Cyrix came out with a lower powered version called the 6x86L. This chip reduces power consumption by 25% or more compared to the 6x86. It uses a smaller, 0.35 micron circuit size, and split rail voltage much like the Pentium with MMX (3.3V external, 2.8V internal). This version is preferred for those seeking to reduce the power problems of earlier models.

Overall, the 6x86 is a decent processor that provides a lot of value, but at the cost of some headaches that users will have to deal with. When it first came out, it was an excellent alternative to the Intel Pentium CPU, because it was selling for several hundred dollars less. As Pentium prices dropped, so did the spread in cost between it and the 6x86, and there was increasingly little reason to use the Cyrix chip. Today the 6x86 is essentially obsolete; the main reason is the inexpensive availability of Cyrix's own 6x86MX MMX chip.

Note: The 6x86 scores quite impressively on the older processor benchmarks such as Norton SI 8.0. Do not be deceived into thinking that the 6x86-PR200 is 60% faster than the Pentium 200 as this benchmark implies. If anything, this statistic just proves how inadequate the older benchmarks really are at accurately measuring the capabilities of a modern PC. Compare to the same benchmark for the AMD K5, which has similar overall performance.

General Information	Manufacturer	Cyrix				
	Family Name	6x86, 6x86L				
	Code name	"M1"				
	Processor Generation	Fifth				
	Motherboard Generation	Fifth				
	Version	6x86-PR120+	6x86-PR133+	6x86-PR150+	6x86-PR166+	6x86-PR200+
	Introduced	1995			1996	
	Variants and Licensed Equivalents	IBM 6x86 (same chip, marked differently and using different test process)				
Speed Specifications	Memory Bus Speed (MHz)	50	55	60	66	75
	Processor Clock Multiplier	2.0				
	Processor Speed (MHz)	100	110	120	133	150
	"P" Rating	120	133	150	166	200
Benchmarks	iCOMP Rating	--				
	iCOMP 2.0 Rating	--				
	Norton SI	676	750	816	907	1020
	Norton SI32	32	~37	41	48	~55
	CPUmark32	--				
Physical Characteristics	Process Technology	CMOS				
	Circuit Size (microns)	0.6 (0.35 for 6x86L)			0.5 then 0.44 (0.35 for 6x86L)	

	Die Size (mm^2)	210 (169 for 6x86L)			169	
	Transistors (millions)	3.0				
Voltage, Power and Cooling	External or I/O Voltage (V)	3.3				
	Internal or Core Voltage (V)	3.3 (2.8 for 6x86L)				
	Power Management	SMM				
	Cooling Requirements	Active heat sink				
Packaging	Packaging Style	296-Pin SPGA				
	Motherboard Interface	Socket 7				
External Architecture	Data Bus Width (bits)	64				
	Maximum Data Bus Bandwidth (Mbytes/sec)	381.5	419.6	457.8	508.6	572.2
	Address Bus Width (bits)	32				
	Maximum Addressable Memory	4 GB				
	Level 2 Cache Type	Motherboard				
	Level 2 Cache Size	Usually 256 KB - 512 KB				
	Level 2 Cache Bus Speed	Same as Memory Bus				
	Multiprocessing	No				
Internal Architecture	Instruction Set	x86				
	MMX Support	No				

	Processor Modes	Real, Protected, Virtual Real
	x86 Execution Method	Native
Internal Components	Register Size (bits)	32
	Pipeline Depth (stages)	7
	Level 1 Cache Size	16 KB Unified
	Level 1 Cache Mapping	4-Way Set Associative
	Level 1 Cache Write Policy	Write-Through, Write-Back
	Integer Units	2
	Floating Point Unit / Math Coprocessor	Integrated
	Instruction Decoders	1
	Branch Prediction Buffer Size / Accuracy	256 entries / 85-90%
	Write Buffers	4
	Performance Enhancing Features	Out of Order Execution, Speculative Execution, Register Renaming

TABLE 5.14 CYRIX 6x86, 6x86L PROCESSOR SUMMARY

AMD K5 ("K5" / "5k86")

AMD's entry in the fifth generation processor sweepstakes is called the K5. This processor was eagerly awaited and it was hoped that it would provide a viable alternative to the Pentium early in the Pentium's life cycle. Unfortunately, AMD delivered the processor over a year late and at much lower clock speeds than had been originally

anticipated. As a result instead of being the "Pentium killer" AMD had hoped for, the K5 was positioned as a low-cost Pentium alternative, much like the Cyrix 6x86.

The K5 is, internally, a very advanced processor, the most advanced of the fifth-generation chips. Internally it is more comparable to the Pentium Pro. It is an x86 translation/emulation processor, decoding x86 instructions into RISC-like microinstructions and executing them on a 6-pipeline internal core. This allows the K5 to achieve higher performance than a Pentium of the same speed. In many ways, the K5 is sixth-generation, but its performance level is held back to fifth-generation levels due its low clock speeds.

The K5 is not a Pentium "clone" in the traditional sense of the word, because it is a totally different internal design. It is also not 100% Pentium compatible because it is not an exact clone. However, it is a very close approximation, and suffers from fewer compatibility problems than the 6x86.

Due to its advanced internal architecture, the K5 (like the 6x86) is labeled according to its "P rating", which attempts to label the processor according to what speed Pentium the chip's performance approximates. This is a good way for AMD to position its products in a marketplace that overvalues the clock speed number. The K5 is available in P rating speeds from 75 to 166 and goes in a Socket 5 or Socket 7 motherboard. It is produced on a 0.35 micron CMOS process, until recently state of the art. AMD manufactures its own chips.

Compared to the Pentium (classic), the K5 has the following architectural improvements that contribute to its greater performance to clock speed ratio (it also has all the advantages over the 486 that are mentioned in the section discussing the Pentium):

- RISC-Based Internal Architecture: The K5 is internally a highly-parallel RISC processor using an x86 decoding front-end.
- Six Execution Units: The K5 has five integer units and one floating point unit for processing micro-instructions, while the Pentium has only two integer units.
- Out of Order Completion: Instructions flowing down the two integer pipelines can complete out of order.

- Superior Branch Prediction Unit: The branch target buffer is four times the size of the Pentium's (although K5 claims a lower accuracy with it than Intel does with the Pentium. Seems odd to me.)
- Larger Primary Cache: The K5 has a 16 KB instruction cache instead of the Pentium's 8 KB. The data cache is unchanged at 8 KB.
- Improved Cache Mapping: The primary cache in the K5 is 4-way set associative instead of the Pentium's 2-way.
- Register Renaming: This feature improves parallel performance of the pipelines.
- Speculative Execution: The K5 uses speculative execution to reduce pipeline stall time in its RISC core.

While these improvements are impressive, they are the reason that the K5 has P ratings well above its internal clock speeds; if it didn't have the higher P ratings the K5 could not compete with the Pentium at all. Like the Cyrix 6x86, the K5's main's attraction is significantly lower price than the Pentium. Unlike the 6x86, the K5 has a reputation for being reasonably compatible and has far fewer problems than the Cyrix chip does. There aren't the heat and motherboard compatibility problems of the 6x86 associated with the K5. However, it (arguably) does not offer as high a level of performance as the 6x86, and there is no K5 chip near the power level of the 6x86-PR200+.

The K5 can have some compatibility problems due to the fact that it is not a Pentium. Some Pentium-specific software can cause problems with the K5, but these programs are relatively rare. The "CPUID" problem associated with the 6x86 under Windows 95 is not an issue for the K5.

While the K5 exceeds or beats a Pentium of its "P rating" at integer operations, it lags behind in floating-point performance. The K5 falls somewhere between the 6x86 and the Pentium in terms of its FPU, with performance roughly equal to a Pentium 25% slower than the K5's P rating. This really isn't much of an issue for most people since few applications make extensive use of the FPU (although Quake is one high-profile game that does.)

The 6x86 suffers from the problem where it is occasionally overclocked by accident because users set the motherboard to the clock speed of the P rating instead of the actual

clock speed of the chip itself. The K5 avoids this by making two design decisions. First, all of the K5 chips run on the same memory bus speed as the equivalent Pentium of its P rating. So a K5-PR133 runs on a 66 MHz bus speed, like the Pentium 133, and unlike the 6x86-PR133+. Second, all K5 chips are designed so that if you configure them with the multiplier that an equivalent-rated Pentium would use, the chip will run with the correct internal multiplier. The combination of these two characteristics means that to set up a K5 you jumper your motherboard as if it were a Pentium and it will work. This avoids a lot of confusion.

The K5 also has an undocumented, generally unused multipler setting: if you set it to 3x, it runs at 2x. The reason for this is generally believed to be that AMD intended to release a K5-PR200, but never did. This chip was supposed to run at 133 MHz and have performance equivalent to a Pentium 200. Of course the Pentium 200 has a multiplier of 3x, so this would have been consistent with the other K5s' ability to be jumpered as if they were a Pentium. The full table of AMD K5 multipliers is:

Motherboard Multiplier Setting	CPU Interpreted Multiplier	Processors
1.5x	1.5x	K5-PR75 (75 MHz), K5-PR90 (90 MHz), K5-PR100 (100 MHz)
2.0x	1.5x	K5-PR120 (90 MHz), K5-PR133 (100 MHz)
2.5x	1.75x	K5-PR166 (116 MHz)
3.0x	2.0x	K5-PR200 (133 MHz, not released)

As you can see above, the K5 has its own confusions to compensate for its better handling of jumper settings. :^) The main one is that the PR90 and PR120 both run at 90 MHz processor speed, and the PR100 and PR133 both run at 100 MHz. How is it possible for two processors with the same design to have different performance ratings running at the same speed? It isn't--and that's the confusion, because the PR75, PR90 and PR100, have a different internal design than the PR120, PR133 and PR166. The first three chips were created earlier, and were actually at one point called the "5k86".

The K5 is a good alternative to the classic Pentium, with performance equal to a mid-level Pentium. It does not have many of the problems that the Cyrix 6x86 does, but it also doesn't offer performance as high as the top-end 6x86s. However, it is now as obsolete as the 6x86, and for that matter the classic Pentium that both compete with. For the home builder on a very tight budget the K5-PR166 is a decent choice, but for the power user it is already too weak to be considered. The price of the K6 has come down so far, and the K6 is so much better than the k5, that the latter is really only a good choice for a motherboard that does not support split voltage.

Note: The K5 sometimes scores poorly on the older processor benchmarks such as Norton SI 8.0. In fact, the PR120 scores lower than the PR100. This reflects more on the inaccuracy of the old benchmarks (when applied to a modern processor like the K5) than any weakness of the K5, and the fact that the internal core of the K5 was redesigned between the 100 and the 120. Compare to the 6x86, which has similar performance, or the SI32 scores for the K5 itself.

General Information	Manufacturer	AMD					
	Family Name	K5					
	Code name	"5k86"			"K5"		
	Processor Generation	Fifth					
	Motherboard Generation	Fifth					
	Version	K5 PR75	K5 PR90	K5 PR100	K5 PR120	K5 PR133	K5 PR166
	Introduced	1995		1996			1997
	Variants and Licensed Equivalents	--					
Speed Specifications	Memory Bus Speed (MHz)	50	60	66	60	66	66
	Processor Clock	1.5					1.75

	Multiplier						
	Processor Speed (MHz)	75	90	100	90	100	116
	"P" Rating	75	90	100	120	133	166
Benchmarks	iCOMP Rating	--					
	iCOMP 2.0 Rating	--					
	Norton SI	286	359	390	~380	407	~470
	Norton SI32	~25	30	33	~37	43	47
	CPUmark32	--					
Physical Characteristics	Process Technology	CMOS					
	Circuit Size (microns)	0.35					
	Die Size (mm^2)	161					
	Transistors (millions)	4.3					
Voltage, Power and Cooling	External or I/O Voltage (V)	3.52					
	Internal or Core Voltage (V)	3.52					
	Power Management	SMM					
	Cooling Requirements	Active heat sink					
Packaging	Packaging Style	296-Pin SPGA					
	Motherboard Interface	Socket 5, Socket 7					
External	Data Bus Width	64					

Architecture	(bits)						
	Maximum Data Bus Bandwidth (Mbytes/sec)	381.5	457.8	508.6	457.8	508.6	508.6
	Address Bus Width (bits)	32					
	Maximum Addressable Memory	4 GB					
	Level 2 Cache Type	Motherboard					
	Level 2 Cache Size	Usually 256 KB - 512 KB					
	Level 2 Cache Bus Speed	Same as Memory Bus					
	Multiprocessing	No					
Internal Architecture	Instruction Set	x86					
	MMX Support	No					
	Processor Modes	Real, Protected, Virtual Real					
	x86 Execution Method	x86 Emulation					
Internal Components	Register Size (bits)	32					
	Pipeline Depth (stages)	5					
	Level 1 Cache Size	8 KB Data, 16 KB Instruction					
	Level 1 Cache Mapping	4-Way Set Associative					
	Level 1 Cache	Write-Through, Write-Back					

	Write Policy	
	Integer Units	5
	Floating Point Unit / Math Coprocessor	Integrated
	Instruction Decoders	1
	Branch Prediction Buffer Size / Accuracy	1024 entries / 75%
	Write Buffers	--
	Performance Enhancing Features	Out of Order Execution, Speculative Execution, Register Renaming

TABLE 5.15 AMD K-5 PROCESSOR SUMMARY

PCI Bus Performance

The PCI bus provides superior performance to the VESA local bus; in fact, PCI is the highest performance general I/O bus currently used on PCs. This is due to several factors:

- Burst Mode: The PCI bus can transfer information in a burst mode, where after an initial address is provided multiple sets of data can be transmitted in a row. This works in a way similar to how cache bursting works.
- Bus Mastering: PCI supports full bus mastering, which leads to improved performance.
- High Bandwidth Options: The PCI bus specification version 2.1 calls for expandability to 64 bits and 66 MHz speed; if implemented this would quadruple bandwidth over the current design. In practice the 64-bit PCI bus has yet to be implemented on the PC (it does exist in non-PC platforms such as Digital Equipment's Alpha and is also found now on servers) and the speed is currently

limited to 33 MHz in most PC designs, most likely for compatibility reasons. For mainstream PCI, we may be limited to 32 bits and 33 MHz for some time to come. However, it appears that the higher-performance PCI options are going to live on, albeit in modified form, through the new Accelerated Graphics Port.

The speed of the PCI bus can be set synchronously or asynchronously, depending on the chipset and motherboard. In a synchronized setup (used by most PCs), the PCI bus runs at half the memory bus speed; since the memory bus is usually 50, 60 or 66 MHz, the PCI bus would run at 25, 30 or 33 MHz respectively. In an asynchronous setup the speed of the PCI bus can be set independently of the memory bus speed. This is normally controlled through jumpers on the motherboard, or BIOS settings. Overclocking the system bus on a PC that uses synchronous PCI will cause PCI peripherals to be overclocked as well, often leading to system stability problems.

PCI Expansion Slots

The PCI bus offers more expansion slots than most VLB implementations, without the electrical problems that plagued the VESA bus. Most PCI systems support 3 or 4 PCI slots, with some going significantly higher than this.

Note: In some systems, not all of the slots are capable of bus mastering. This is now far less common than it was, but with early systems the motherboard manual should be checked to see if this is the case.

The PCI bus offers a great variety of expansion cards compared to VLB. The most commonly found cards are video cards (of course), SCSI host adapters, and high-speed networking cards. (Hard disk drives are also on the PCI bus but are normally connected directly to the motherboard on a PCI system). However, it should be noted that certain functions cannot be provided on the PCI bus. For example, serial and parallel ports must remain on the ISA bus. Fortunately, the ISA bus still has more than enough speed to handle these devices, even today.

PCI Internal Interrupts

The PCI bus uses its own internal interrupt system for dealing with requests from the cards on the bus. These interrupts are often called "#A", "#B", "#C" and "#D" to avoid confusion with the normal numbered system IRQs, though they are sometimes called "#1" through "#4" as well. These interrupt levels are not generally seen by the user except in the BIOS setup screen for PCI, where they can be used to control how PCI cards operate.

These interrupts, if needed by cards in the slots, are mapped to regular interrupts, normally IRQ9 through IRQ12. The PCI slots in most systems can be mapped to at most four regular IRQs. In systems that have more than four PCI slots, or that have four slots and a USB controller (which uses PCI), two or more of the PCI devices share an IRQ.

If you are using Windows 95 OEM SR2, you may see additional entries for your PCI devices under the Device Manager. Each device may have an additional entry entitled "IRQ Holder for PCI Steering". PCI steering is in fact a feature that is part of the Plug and Play portions of the system, and enables the IRQ used for PCI devices to be controlled by the operating system to avoid resource problems. Having this listed in addition to another device under the IRQ list does not mean you have a resource conflict.

PCI Bus Mastering

As discussed in the section on system bus functions and features, bus mastering is the capability of devices on the PCI bus (other than the system chipset, of course) to take control of the bus and perform transfers directly. The PCI bus is the first bus to popularize bus mastering; probably in part because for the first time there are operating systems and software that are really capable of taking advantage of it.

PCI supports full device bus mastering, and provides bus arbitration facilities through the system chipset. PCI's design allows bus mastering of multiple devices on the bus simultaneously, with the arbitration circuitry working to ensure that no device on the bus (including the processor!) locks out any other device. At the same time though, it allows any given device to use the full bus throughput if no other device needs to transfer anything. In a way, the PCI bus acts like a tiny "local area network" inside the computer,

in which multiple devices can each talk to each other, sharing a communication channel that is managed by the chipset.

PCI IDE Bus Mastering

The PCI bus also allows you to set up compatible IDE/ATA hard disk drives to be bus masters. Under the correct conditions this can increase performance over the use of PIO modes, which are the default way that IDE/ATA hard disks transfer data to and from the system. When PCI bus mastering is used, IDE/ATA devices use DMA modes to transfer data instead of PIO; IDE/ATA DMA modes are described in detail here.

Since this capability was made available to newer machines, it has been one of the most talked about (and most misunderstood) functions of the modern PC. There is a lot of confusion amongst PC users about what PCI IDE bus mastering does and how it works. In particular, there are a lot of misconceptions about its performance advantages. In addition, there have been a lot of problems with compatibility in getting this new technology to work.

IDE bus mastering requires all of the following in order to function at all:

- Bus Mastering Capable System Hardware: This includes the motherboard, chipset, bus and BIOS. Most newer motherboards using the Intel 430 Pentium chipset family (FX, HX, VX, TX) or the Intel 440FX Pentium Pro chipset, will support bus mastering IDE.
- Bus Mastering Hard Disk: Normally this means that the drive must be capable of at least multiword DMA mode 2 transfers. All Ultra ATA hard disks also support bus mastering.
- 32-Bit Multitasking Operating System: This means usually Windows NT, Windows 95, Linux, or similar (but see below for caveats.)
- Bus Mastering Drivers: A special driver must be provided to the operating system to enable bus mastering to work.

Getting this all set up can be a great deal of work. In particular, the following are common problems encountered when trying to set up bus mastering:

- Driver bugs and incompatibility issues, especially with older drivers that are "generic" and not tuned to a particular motherboard.
- Older hard disk drives not working properly.
- Problems with dissimilar drives connected to the same IDE channel as master and slave.
- Problems when using a CD-ROM drive alone on an IDE channel without a hard drive.
- Bus mastering drivers that don't work on certain motherboards; also, some motherboards or systems just will not work with bus mastering at all.

Assuming that you get bus mastering IDE to work, you will see improvement if you are using a true multi-tasking operating system, and you are running multiple applications that is disk-access-intensive. This would not generally include most regular Windows 95 users, for example. Bus mastering IDE will not help at all in the following situations:

- It will not make that 100 MB transfer from C: to D: that you are sitting and watching go much faster at all.
- It will not speed up DOS games.
- It will not make applications load more quickly (unless you somehow are loading more than one at a time).
- It will not speed up single applications.

Especially: IDE bus mastering will not really speed up Windows 95 in general. Windows 95 does not do "true" multitasking and in many cases the processor will be held up waiting for the transfer to complete even if bus mastering is employed. So even though the processor in theory is freed up to do other things, it doesn't really do other things. Also, most people multitask by switching between applications that are open, but rarely have anything running in two or more simultaneously.

For most people, IDE bus mastering is not worth the effort and problems, and I now do not bother with it on new installs of Windows 95. This may be somewhat controversial, but in my opinion it is very overrated as a potential system improvement, given how much effort it requires. You're better off working overtime for a few hours and

buying another 16 MB of RAM. If you feel like trying it, contacting the company that made your motherboard for a driver set is a good place to start. You can also try Intel for a generic driver that may work on your Intel-chipset system. I'd recommend that you back up your hard disk first before trying any of these... refer to this section of the Troubleshooting Expert for more help resolving problems with these drivers if you have difficulties with them.

I am hopeful that in time, bus mastering over the IDE/ATA interface will be improved and these problems will be just a distant memory. With the creation of Ultra ATA and the DMA-33 high-speed transfer mode, it appears that the future lies in the use of PCI bus mastering with the IDE/ATA interface. There is just some work to do until this support is both universal and well-implemented.

PCI Plug and Play

The PCI bus is part of the Plug and Play standard developed by Intel, with cooperation from Microsoft and many other companies. PCI systems were the first to popularize the use of Plug and Play. The PCI chipset circuitry handles the identification of cards and works with the operating system and BIOS to automatically set resource allocations for compatible peripheral cards

CHAPTER 6

FAULT ANALYSIS AND SOLUTION OF PC-XT MOTHERBOARD

6.1 Methods of fault analysis

After study of motherboards from 8088 to recent new technology motherboards, I have decided to analyze most common faults and get solution of those faults. After research I felt that if one wants to study signal behavior of different logic portion of the motherboard, one should study PC-XT motherboard. I have selected a motherboard with specific design of logic output section.

There are many different types of troubleshooting methods. All this depends on the monitor program (BIOS program) written for particular motherboard. I have selected the PC-XT motherboard trainer module with additional specific design for fault analysis. This system has different fault analysis points to simulate the various faults, covering individual sections of a personal computer. I have decided to record signals from following different logic sections of motherboard.

1. CPU logic
2. DRAM logic
3. Wait state logic
4. DMA/FDC logic
5. Serial communication
6. Parallel communication
7. Keyboard logic
8. Video logic
9. Video RAM logic
10. Time of date logic
11. Timer logic
12. DMA logic
13. Parity generation logic
14. Clock generation logic
15. FDD logic
16. FDD BIOS logic

17. RTC logic
18. Control logic
19. Bus arbitration logic

6.2 Observation of probe points

1. SIGNAL NAME : - KBCLK

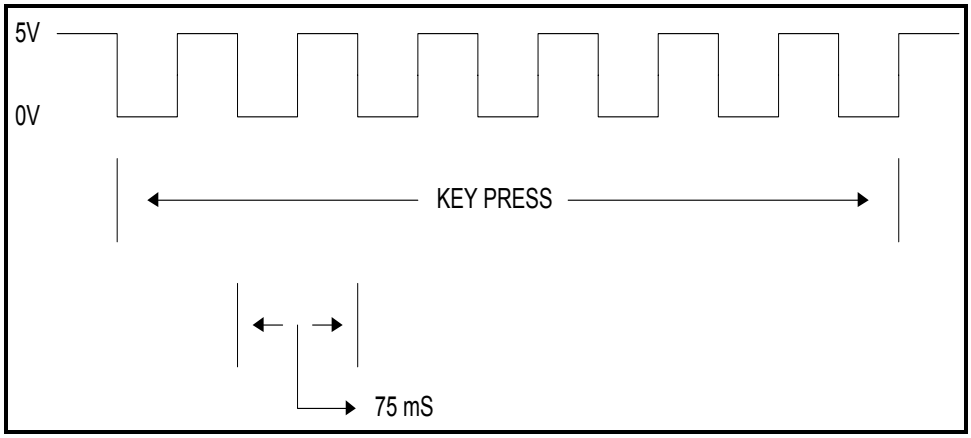
EXPANSION : Keyboard Clock

DESCRIPTION : This clock generated by the keyboard controller is used by the keyboard logic in the motherboard to convert the serial scan code from the keyboard to parallel scan code. for each key press, keyboard sends a 8 bit scan code serially with 8 clock pulses.

OBSERVATION :

1) LOGIC PROBE : Continuous pulses when key is pressed.

2) CRO :



2. SIGNAL NAME : - RST DRV.

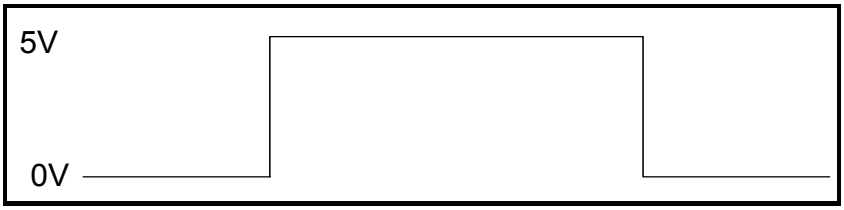
EXPANSION : RESET DRIVE

Description : Resets the CPU and associated controllers. This signal is generated by Reset logic of the clock generator, during power on Reset hardware reset. This is an active high signal.

OBSERVATION :

1) LOGIC PROBE : High pulse during Reset.

2) CRO :



3. SIGNAL NAME : - PCK

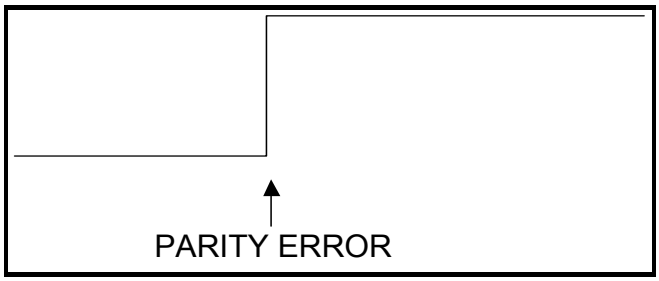
EXPANSION : PARITY CHECK

DESCRIPTION : This signal indicates to the CPU, that the parity error has occurred in the DRAM. This signal generats NMI to the CPU.

OBSERVATION :

1) LOGIC PROBE : High signal during Parity error.

2) CRO :



4. SIGNAL NAME : - RAS

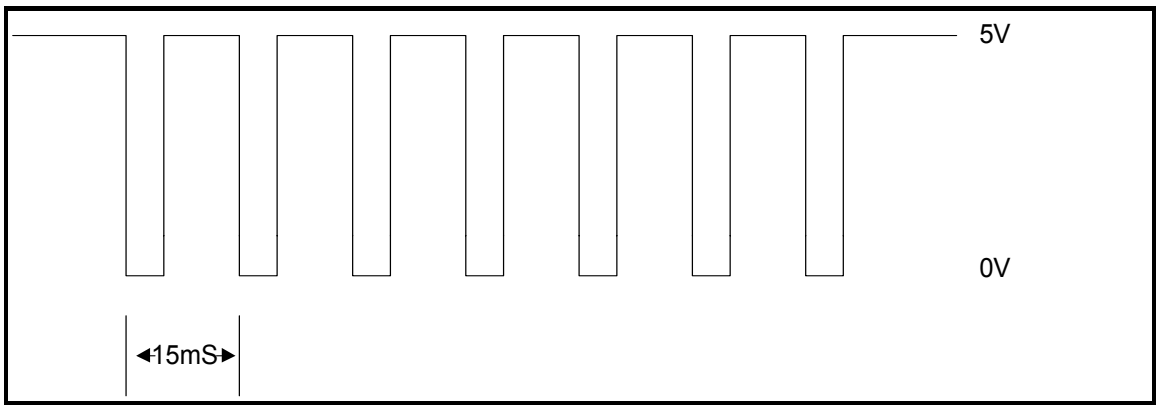
EXPANSION : ROW ADDRESS STROBE

DESCRIPTION : RAS is used by the DRAM chip to latch the row address into its internal latch. The RAM timing section for every memory operation generates this.

OBSERVATION :

1) LOGIC PROBE : Continuous pulses during memory read or write operation.

2) CRO :



5. SIGNAL NAME : - CAS

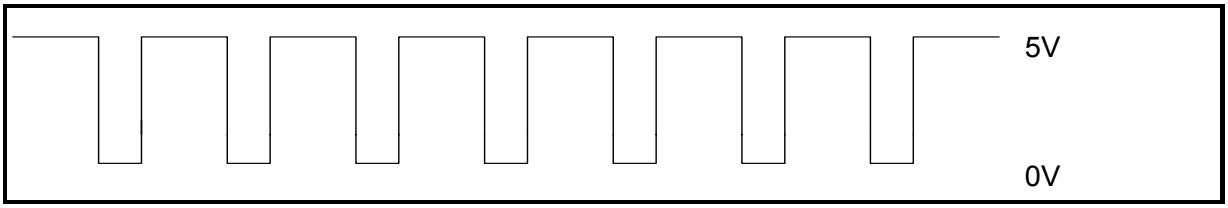
EXPANSION : COLUMN ADDRESS STROBE

DESCRIPTION : CAS is USED by the DRAM chip to latch the column address into its internal latch. The RAM timing section for every memory operation generates this.

OBSERVATION :

1) LOGIC PROBE : Continuous low pulses during memory operation.

2) CRO :



6. SIGNAL NAME : - HSYNC

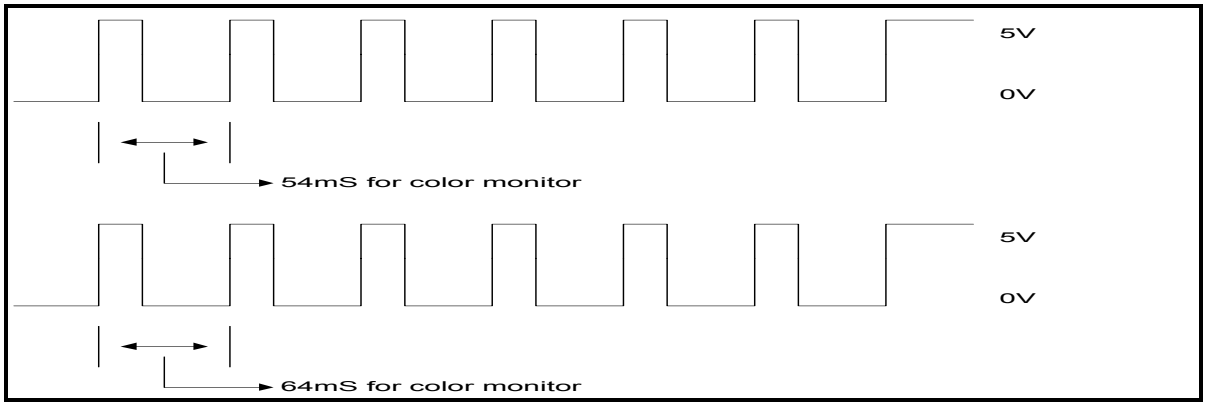
EXPANSION : HORIZONTAL SYNCHRONIZATION PULSES.

DESCRIPTION : It is the timing single generated by the CRT controller to the CRT monitor. This signal specifies the horizontal scanning rate of the monitor. The frequency of HSYNC corresponds to the resolution of the monitor.

OBSERVATION :

1) LOGIC PROBE : Continuous pulses.

2) CRO :



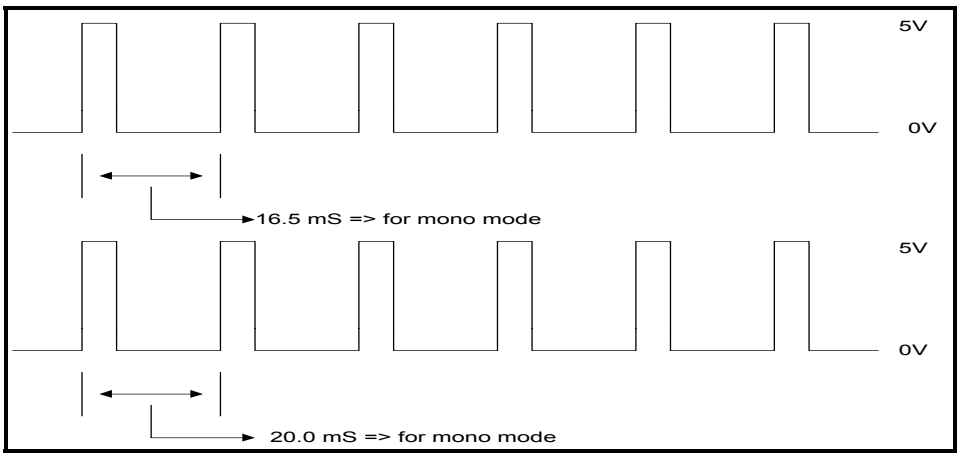
7. SIGNAL NAME : - VSYNC

EXPANSION : VERTICAL SYNCHRONIZATION PULSES.

DESCRIPTION : It is the timing single generated by the CRT controller to the CRT monitor. This signal specifies the vertical scanning rate of the monitor. The frequency of VSYNC corresponds to the resolution of the monitor.

OBSERVATION :

1) LOGIC PROBE : Continuous pulses.
2) CRO :



8. SIGNAL NAME : - SER DATA

EXPANSION : SERIAL DATA OUT LINE.

DESCRIPTION : It is the output of the serial port, used to transmit the serial data during serial communication. Data is transmitted serially through this line with start, stop and parity bits.

OBSERVATION :

1) LOGIC PROBE : Continuous pulses.
2) CRO :

9. SIGNAL NAME : - SPK

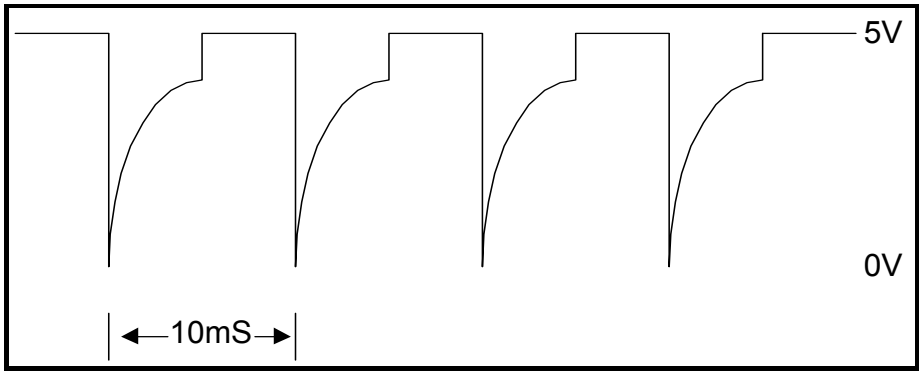
EXPANSION : SPEAKER DATA

DESCRIPTION : This signal is used by the CPU to generate different speaker tones from PC speaker. This is the output of the timer, which produces signal with different frequencies.

OBSERVATION :

1) LOGIC PROBE : Speaker produces continuous pulses.

2) CRO :



10. SIGNAL NAME : - DIR

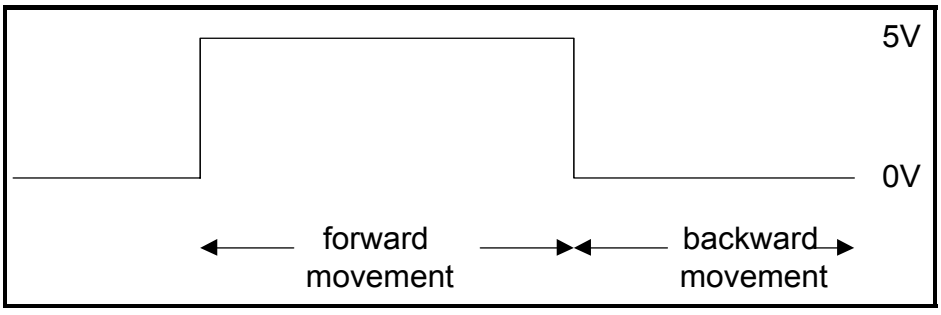
EXPANSION : FDD READ/WRITE HEAD DIRECTION.

DESCRIPTION : It is the generated by floppy disk controller to indicate the direction of read/write head movement to the floppy disk drive.

OBSERVATION :

1) LOGIC PROBE : an active level indicates away from the center of the diskette (OUT). In active level indicates towards the center of the diskette (IN).

2) CRO :



11. SIGNAL NAME : - MOTOR A ON

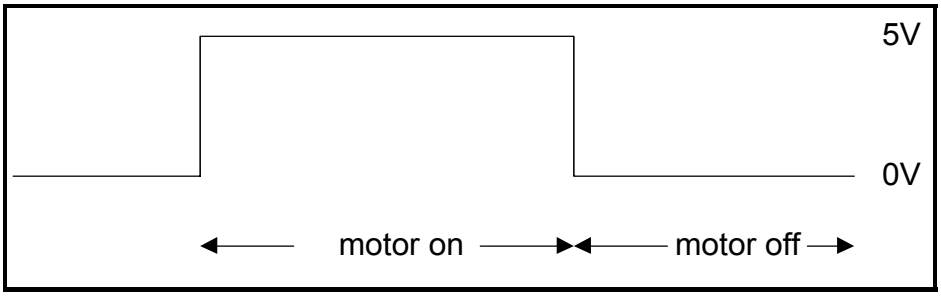
EXPANSION : DRIVE A MOTOR ON.

DESCRIPTION : This single is generated by the FDC which makes the FDD to turn on the DC motor. The controller generates a separate MOTOR-ON signal for each drive.

OBSERVATION :

1) LOGIC PROBE : A high signal during motor rotation.

2) CRO :



12. SIGNAL NAME : - STEP

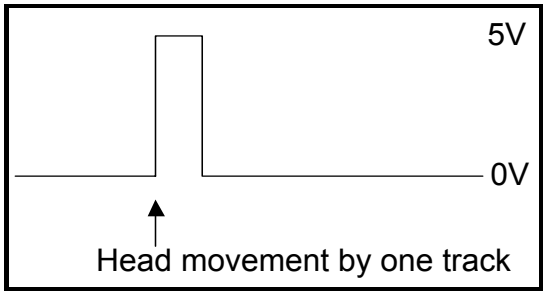
EXPANSION : HEAD STEPPING PULSE.

DESCRIPTION : This signal is generated by FDC to FDD, for the movement of head by one track. These signals along with the direction signal specify the forward and backward movement of read/write head.

OBSERVATION :

1) LOGIC PROBE : One pulse for each head movements.

2) CRO :



13. SIGNAL NAME : - TRK00

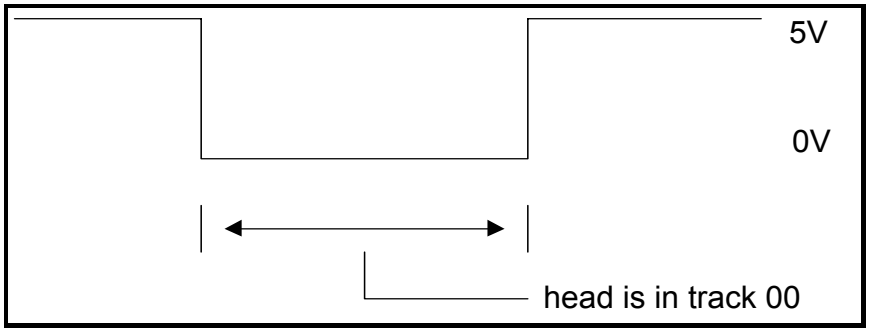
EXPANSION : TRACK 00 SIGNAL

DESCRIPTION : It is generated by FDD to FDC, to indicate that the head is on track 00 position. This signal is sensed using the TRK 00 sensor available in the drive.

OBSERVATION :

1) LOGIC PROBE : The signal low, when the head is in Track 00.

2) CRO :



14. SIGNAL NAME : - INDEX

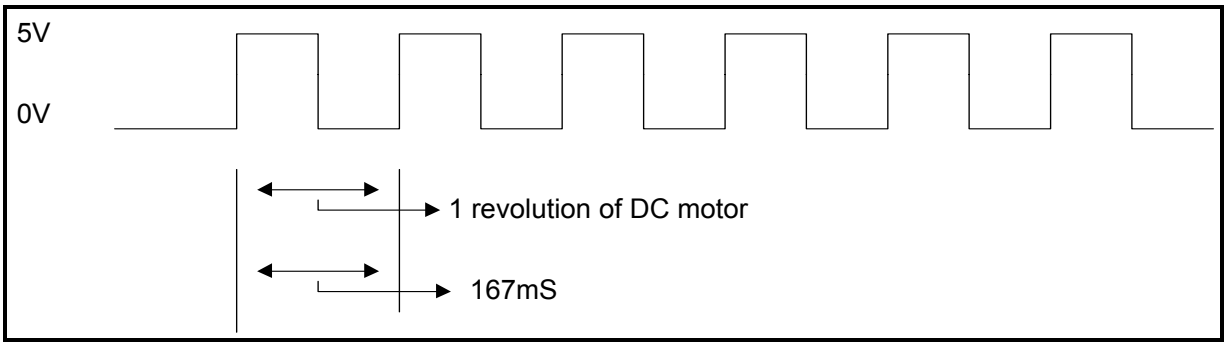
EXPANSION : INDEX SIGNAL.

DESCRIPTION : Generated by FDD to indicate the sector 1 in a floppy diskette. Used by the PC to determine what- ever, floppy diskette is in the drive and also to control the speed of the DC motor

OBSERVATION :

1) LOGIC PROBE : Continuous pulses, when the drive with diskette inserted rotates.

2) CRO :



15. SIGNAL NAME : - DRQ 0

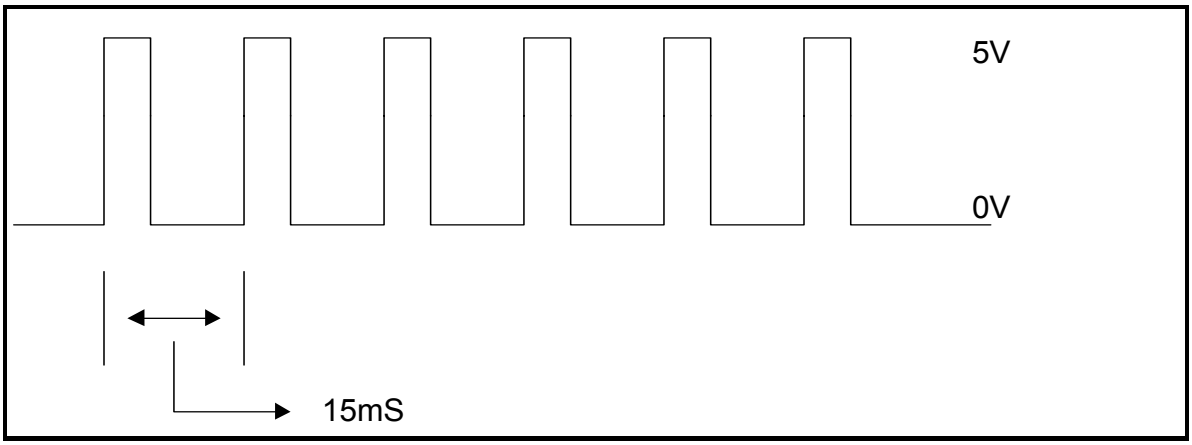
EXPANSION : DMA REQUEST FOR CHANNEL 0.

DESCRIPTION : This signal is used for dynamic memory refreshing. The timer 1 generates request to DMA channel 0. When this line is high all DRAM chips are refreshed row by row. For each DRQ 0, the DMA controller increments the row address and performs dummy read operation.

OBSERVATION :

1) LOGIC PROBE : Continuous high pulses.

2) CRO :



16. SIGNAL NAME : - DACK 0

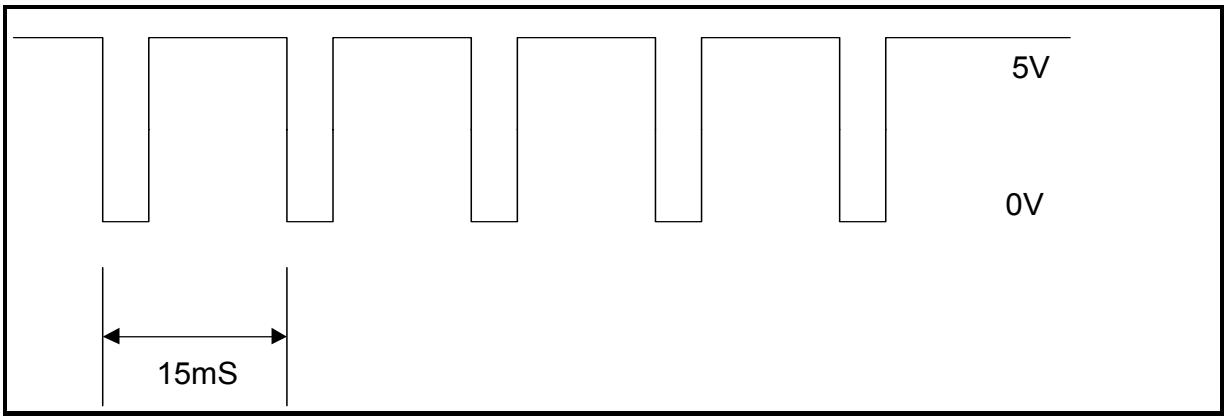
EXPANSION : DMA ACKNOWLEDGE FOR CHANNEL 0

DESCRIPTION : This signal is generated by the DMA controller when a memory refresh is performed in response to DRQ 0 signal.

OBSERVATION :

1) LOGIC PROBE : Continuous pulses.

2) CRO :



17. SIGNAL NAME : - DRQ 2

EXPANSION : DMA REQUEST FOR CHANNEL 2

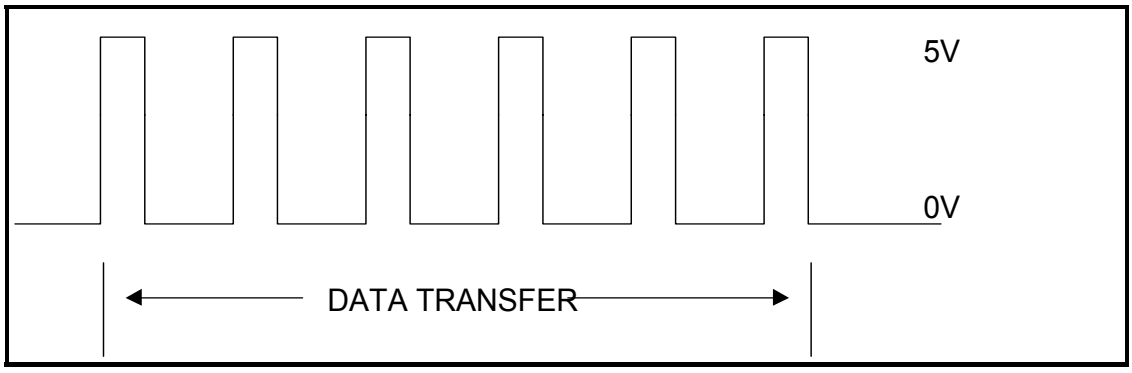
DESCRIPTION : This signal is generated by the FDC under two different condition.

1) The FDC is ready to transfer data to the memory.
2) The FDC is ready to transfer data from the memory.

OBSERVATION :

1) LOGIC PROBE : High pulses at the time of FDD operation.

2) CRO :



18. SIGNAL NAME : - DACK 2

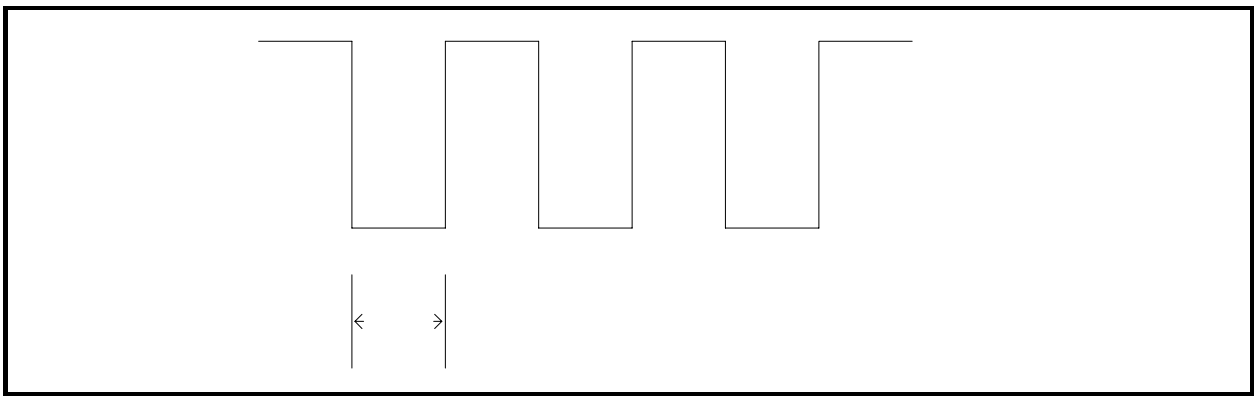
EXPANSION : DMA ACKNOWLEDGE FOR CHANNEL 2.

DESCRIPTION : This signal is generated by the DMA controller when it perform a DMA cycle in response to DRQ2.

OBSERVATION :

1) LOGIC PROBE : Low pulses during FDD operation.

2) CRO :



19. SIGNAL NAME : - TC

EXPANSION : TERMINAL COUNT SIGNAL

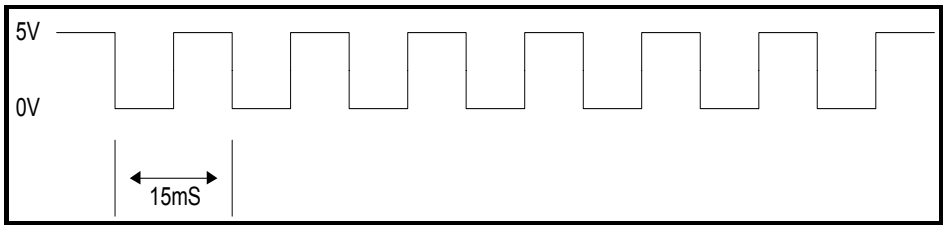
DESCRIPTION : This signal is generated by the DMA controller to indicate that the byte count for a DMA channel has become zero. The FDC reacts to the only if DACK 2 is active.

OBSERVATION :

1) LOGIC PROBE : High pulse during FDD operation.

2) CRO

:



20. SIGNAL NAME : - DMA WAIT

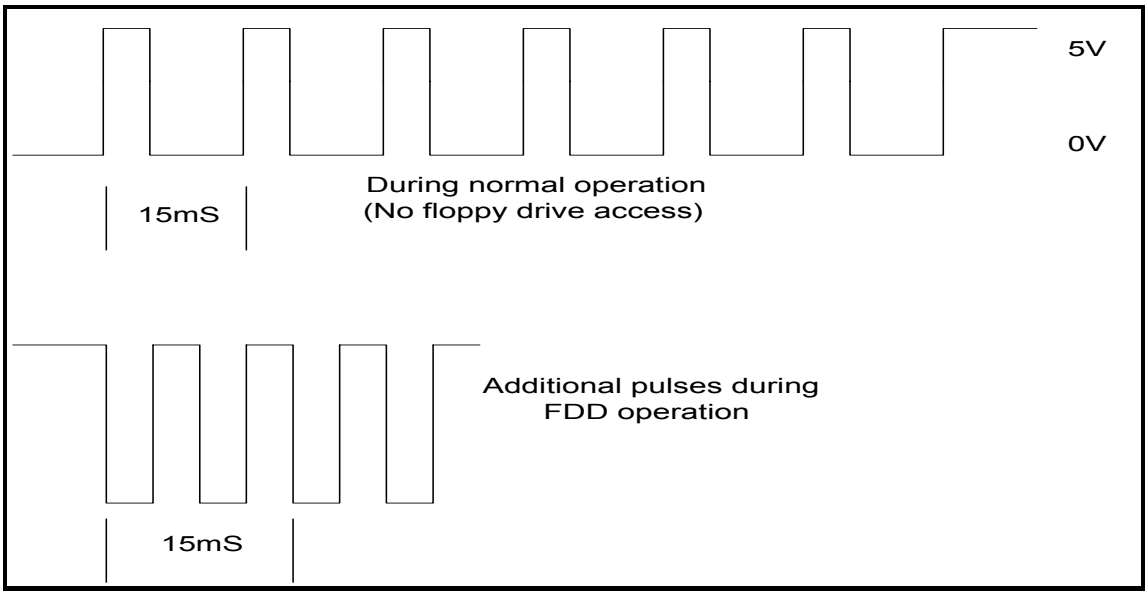
EXPANSION : DMA WAIT SIGNAL

DESCRIPTION : This signal is used to introduce wait state to the CPU during DMA cycle. It indicates that the DMA controller has been given bus grant. It is connected to RDY 1 input of clock generator.

OBSERVATION :

1) LOGIC PROBE : Low pulses during DMA operation.

2) CRO :



21. SIGNAL NAME : - AEN

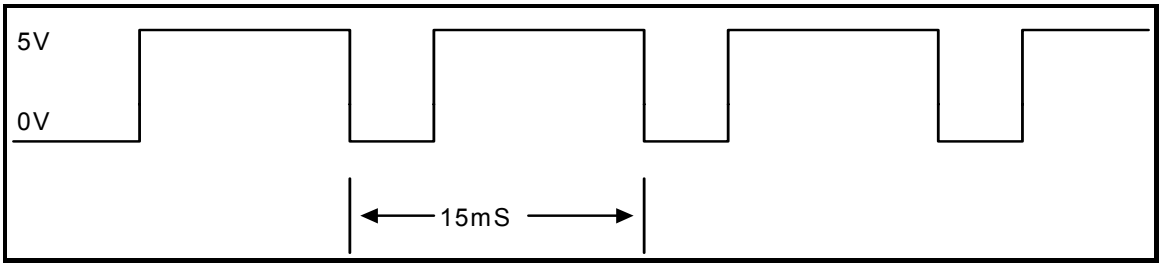
EXPANSION : ADDRESS ENABLE SIGNAL

DESCRIPTION : This signal is generated by the DMA logic. AEN is used to isolate other system bus during DMA transfer.

OBSERVATION :

1) LOGIC PROBE : High signal during DMA operation.

2) CRO :



22. SIGNAL NAME : - D0 & D7

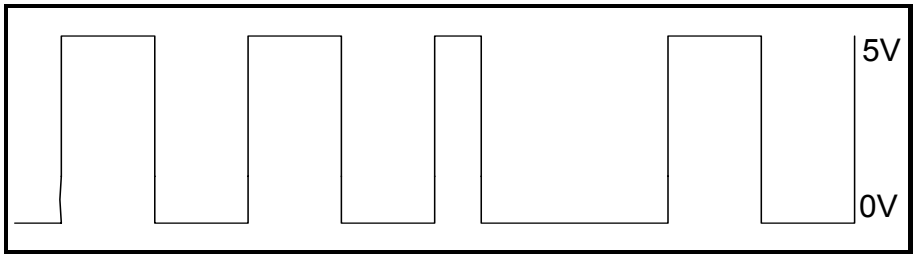
EXPANSION : DATA LINES 0 & 7.

DESCRIPTION : Data lines of the CPU. Used to transfer 8 bit data to and from CPU to other peripherals.

OBSERVATION :

1) LOGIC PROBE : Continuous irregular pulses.

2) CRO :



23. SIGNAL NAME : - A0 & A1

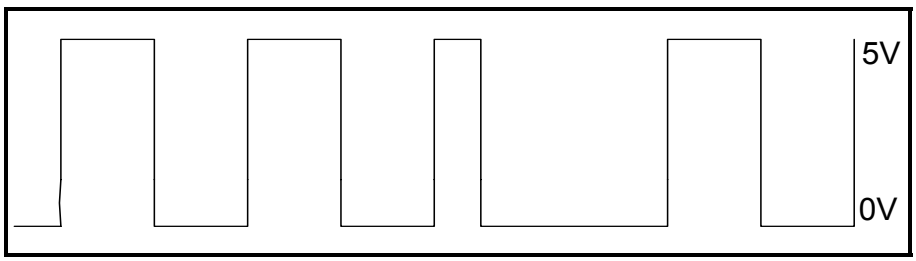
EXPANSION : ADDRESS 0 & 1.

DESCRIPTION : CPU Address lines. Used by the CPU to address peripheral devices and memory locations. The two points given is the least significant of the 20 bit address lines of 8088 CPU.

OBSERVATION :

1) LOGIC PROBE : Continuous irregular pulses.

2) CRO :



24. SIGNAL NAME : - S0, S1, S2

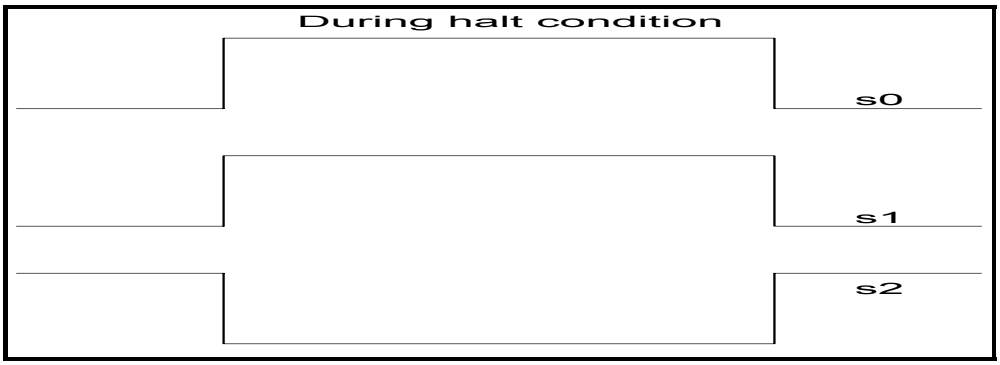
EXPANSION : BUS STATUS SIGNALS.

DESCRIPTION : These three output signals from the CPU are used to indicate the nature of the bus cycles to the bus controller.

Characteristics	$\overline{S2}$	$\overline{S1}$	$\overline{S0}$
Interrupt Acknowledge	0	0	0
Read I/O	0	0	1
Write I/O	0	1	0
Halt	0	1	1
Instruction Fetch	1	0	0
Read Data from memory	1	0	1
Write Data from memory	1	1	0
Passive (No Bus Cycle)	1	1	1

OBSERVATION :

1) LOGIC PROBE : Continuous irregular pulses.
2) CRO :



25. SIGNAL NAME : - ALE

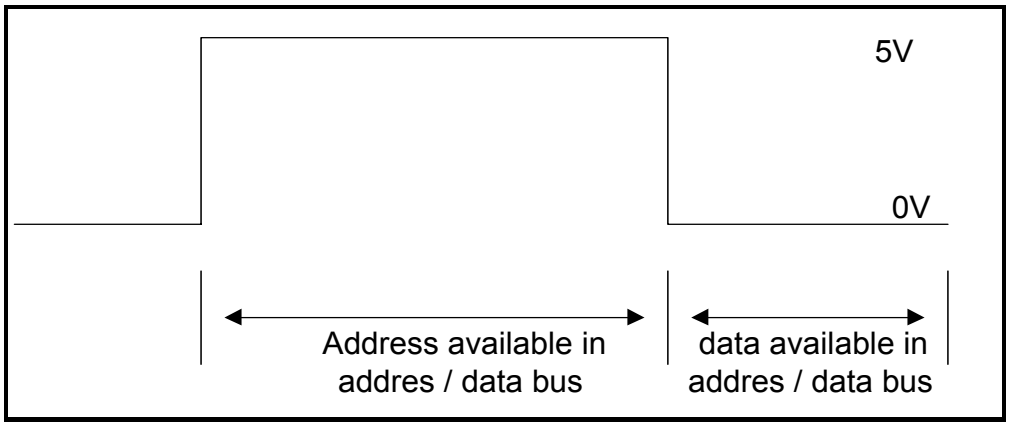
EXPANSION : ADDRESS LATCH ENABLE

DESCRIPTION : This signal is generated by the bus controller, to latch the address line from the multiplex address data line on the CPU.

OBSERVATION :

1) LOGIC PROBE : High pulses during address latching operation.

2) CRO :



26. SIGNAL NAME : - IOW

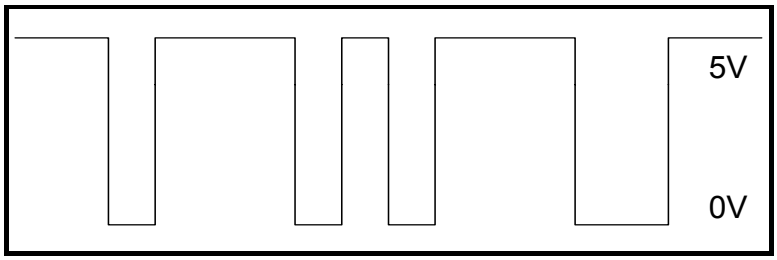
EXPANSION : I/O WRITE SIGNAL.

DESCRIPTION : This signal is generated by the bus controller or the DMA controller during write operation in I/O mapped mode.

OBSERVATION :

1) LOGIC PROBE : Low, Continuous irregular pulses.

2) CRO :



27. SIGNAL NAME : - IOR

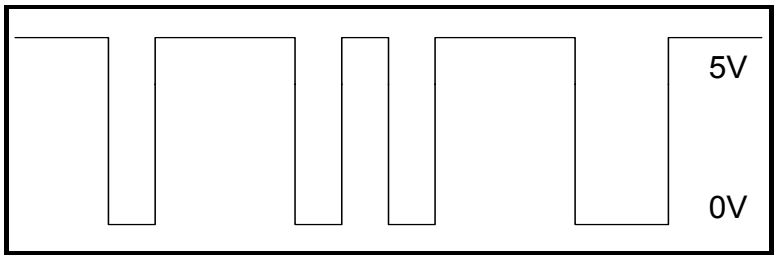
EXPANSION : I/O READ SIGNAL.

DESCRIPTION : This signal is generated by the bus controller or the DMA controller during read operation from the peripherals in I/O mapped mode.

OBSERVATION :

1) LOGIC PROBE : Continuous irregular pulses.

2) CRO :



28. SIGNAL NAME : - MEMW

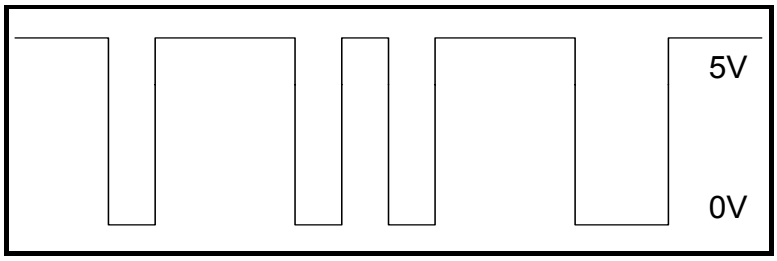
EXPANSION : MEMORY WRITE SIGNAL.

DESCRIPTION : This signal is generated by the bus controller or the DMA controller during memory write operation.

OBSERVATION :

1) LOGIC PROBE : Low, Continuous irregular pulses.

2) CRO :



29. SIGNAL NAME : - MEMR

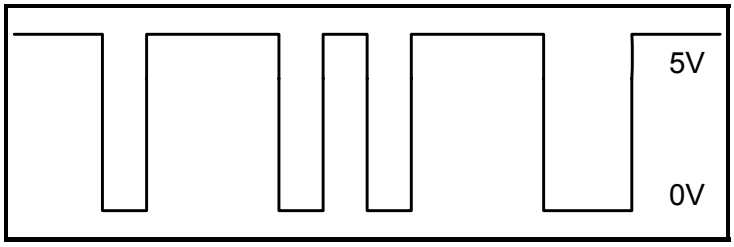
EXPANSION : MEMORY READ SIGNAL.

DESCRIPTION : This signal is generated by the bus controller or the DMA controller during memory READ operation.

OBSERVATION :

1) LOGIC PROBE : Low, Continuous irregular pulses.

2) CRO :



30. SIGNAL NAME : - IRQ 0

EXPANSION : INTERRUPT REQUEST 0

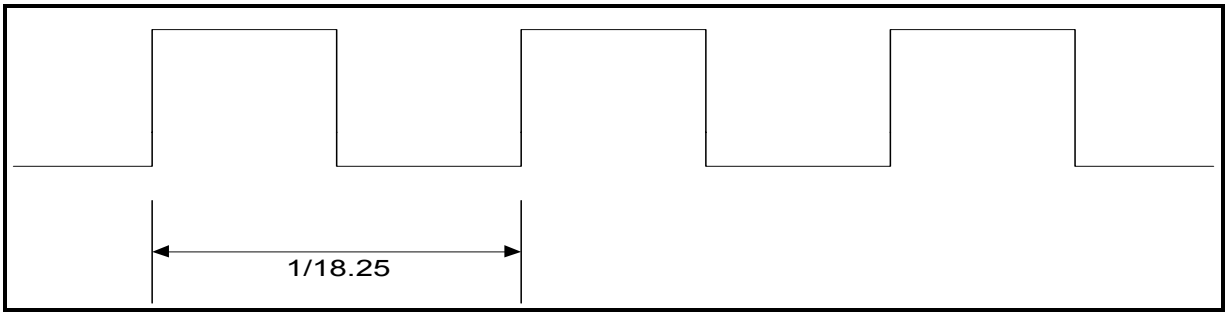
DESCRIPTION : Used by the system to update the time of day count in memory. It is used by DOS to established time of day.

This interrupt is generated by the timer 18.2 times per second.

OBSERVATION :

1) LOGIC PROBE : Continuous pulses.

2) CRO :



31. SIGNAL NAME : - IRQ 1

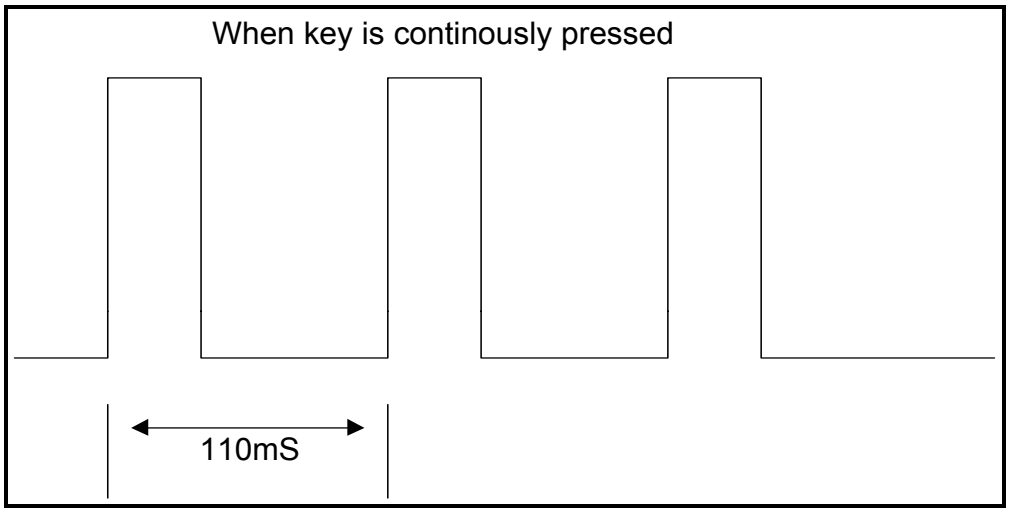
EXPANSION : INTERRUPT REQUEST 1

DESCRIPTION : This interrupt request is generated when a key is pressed in the keyboard. This signal is used by the system to receiv parallel scan codes from the keyboard.

OBSERVATION :

1) LOGIC PROBE : When key presses, two high pulses are generated.

2) CRO :



32. SIGNAL NAME : - IRQ 3

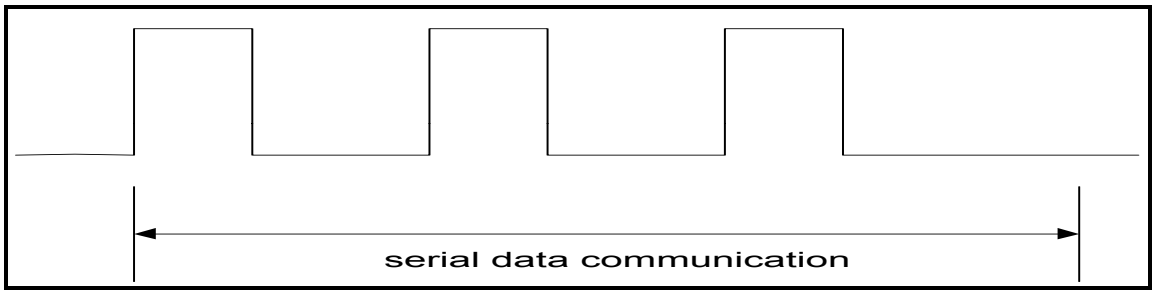
EXPANSION : INTERRUPT REQUEST 3

DESCRIPTION : This interrupt request is used by the serial port COM-2, to indicate the CPU that serial communication is to be performed.

OBSERVATION :

1) LOGIC PROBE : High pulses during serial data transmission.

2) CRO :



33. SIGNAL NAME : - IRQ 4

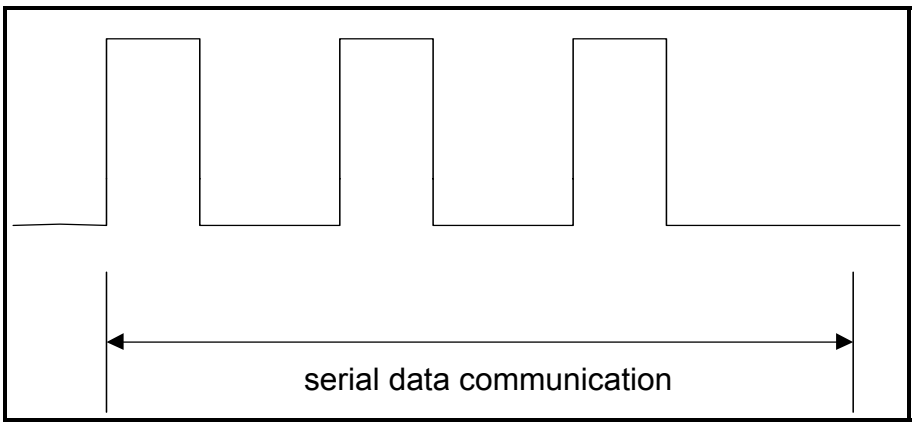
EXPANSION : INTERRUPT REQUEST 4

DESCRIPTION : This interrupt request is used by the serial port COM-1, to indicated the CPU that serial communication is to be performed.

OBSERVATION :

1) LOGIC PROBE : High pulses during serial data transmission.

2) CRO :



34. SIGNAL NAME : - IRQ 6

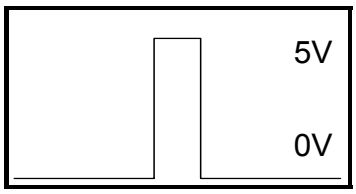
EXPANSION : INTERRUPT request 6

DESCRIPTION : The FDC generates the signal when it has completed the execution of a command. It indicates the CPU that a data transfer has been completed.

OBSERVATION :

1) LOGIC PROBE : High pulses during FDD operation.

2) CRO :



35. SIGNAL NAME : - NMI

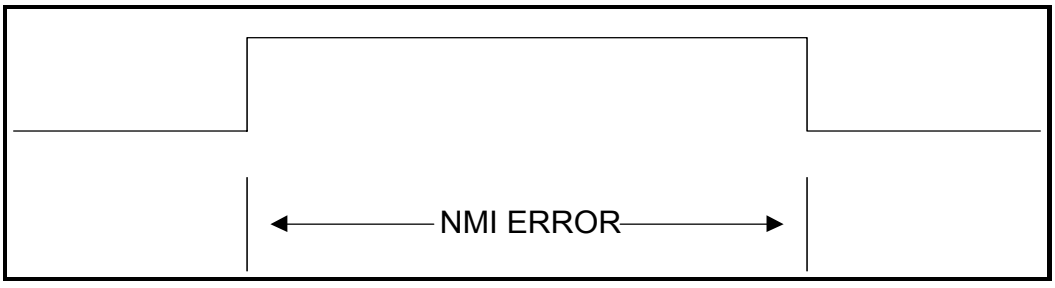
EXPANSION : NON MASKABLE INTERRUPT SIGNAL

DESCRIPTION : This is the non-maskable interrupt to the CPU generated by the NMI logic in the motherboard. It is generated for parity error in motherboard, expansion cards or co-processor error.

OBSERVATION :

1) LOGIC PROBE : One high Pulse when any error takes place.

2) CRO :



36. SIGNAL NAME : - INTR

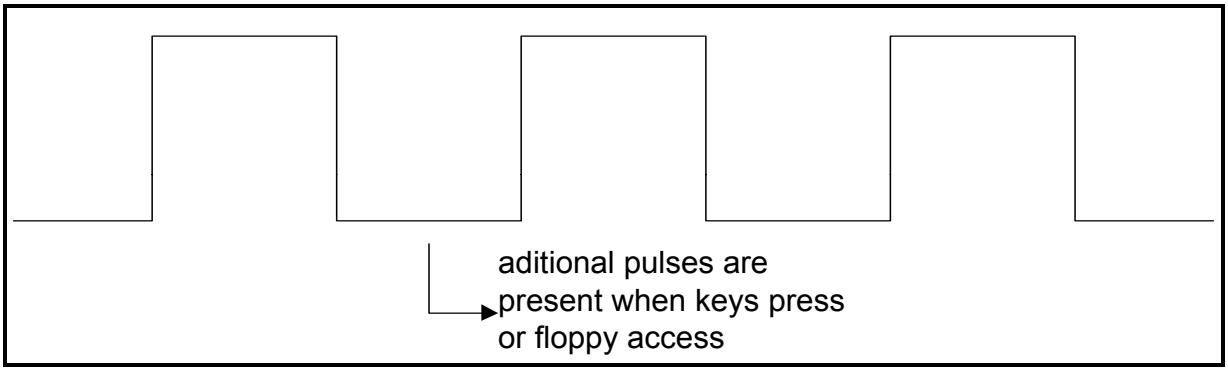
EXPANSION : INTERRUPT

DESCRIPTION : This signal INPUT to the CPU is generated when any of the interrupt request is given to the interrupt controller. This can be masked using software.

OBSERVATION :

1) LOGIC PROBE : Continuous pulses.

2) CRO :



37. SIGNAL NAME : - CLK

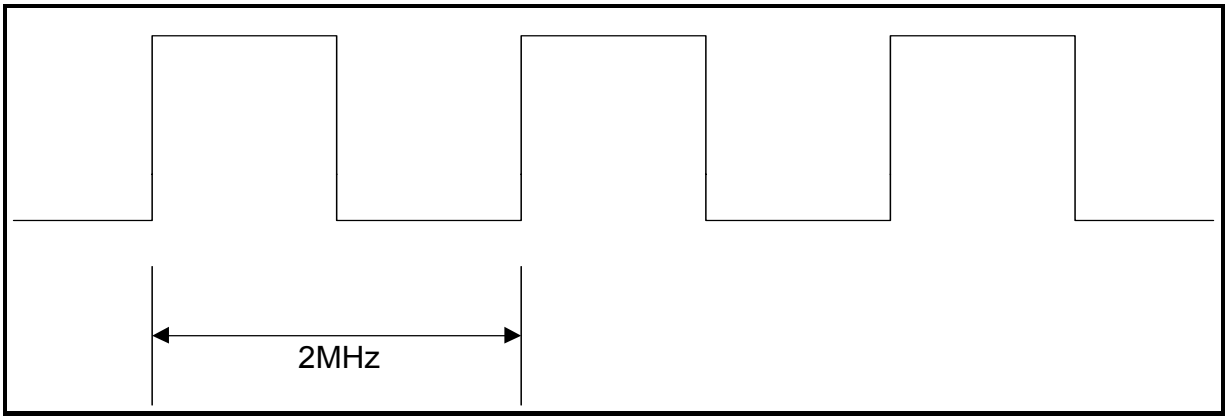
EXPANSION : clock

DESCRIPTION : Main clock signal for the CPU, co-processor, and bus controller. The frequency is 4.77MHz. Generated by the clock generator (8284) by dividing OSC clock by 3.

OBSERVATION :

1) LOGIC PROBE : Continuous uniform pulses.

2) CRO :



38. SIGNAL NAME : - PCLK

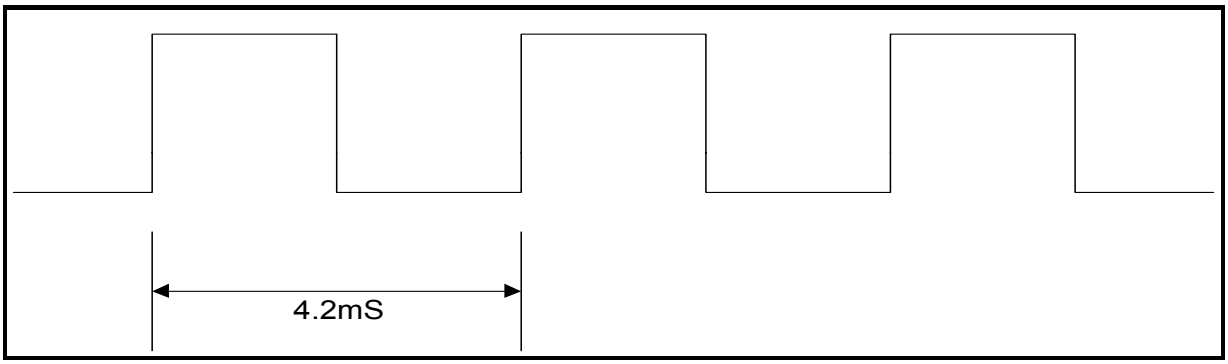
EXPANSION : Peripheral CLOCK

DESCRIPTION : This clock signal is used by slow speed peripherals. The frequency is 2.38MHz. Generated by clock generator by dividing each CLK signal by 2.

OBSERVATION :

1) LOGIC PROBE : Continuous uniform pulses.

2) CRO :



39. SIGNAL NAME : - OSC

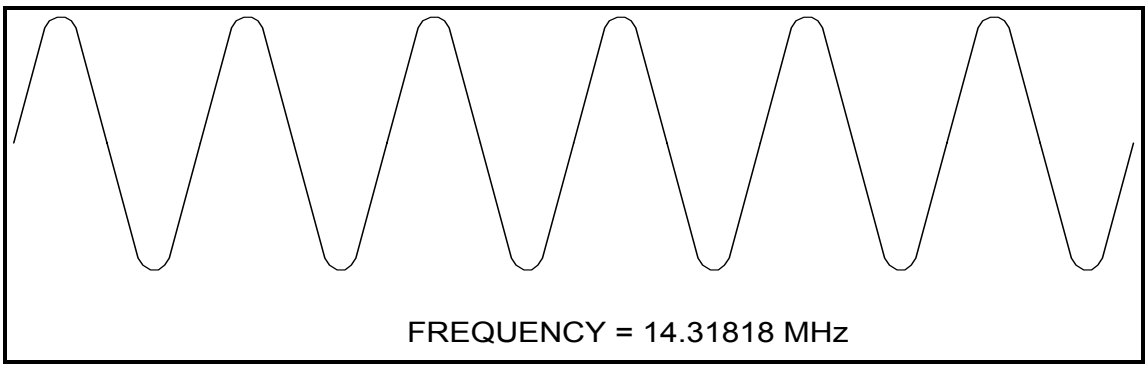
EXPANSION : oscillator CLOCK

DESCRIPTION : It is the standard TTL clock signal available in all motherboards. This signal is used to generate colour burst signal in CGA.

OBSERVATION :

1) LOGIC PROBE : Continuous uniform pulses.

2) CRO :



40. SIGNAL NAME : - READY

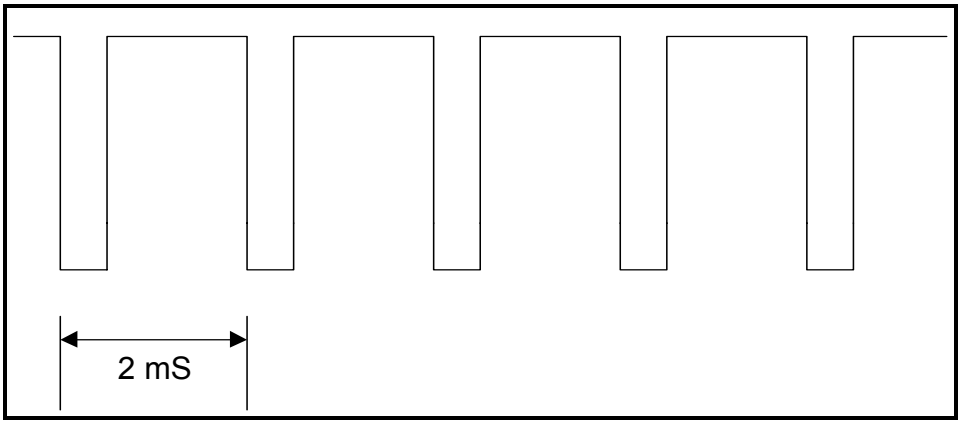
EXPANSION : READY SIGNAL

DESCRIPTION : This active high signal indicates whether the CPU has to enter into wait state. A low signal in this line before T2 cycle is a request for wait state.

OBSERVATION :

1) LOGIC PROBE : Pulsing.

2) CRO :



6.3 Observation and conclusion of faults

This section describes the troubleshooting procedure for the faults introduced through the Fault analysis points. The faults are introduced by toggling the corresponding switches in the front panel of the CPU Trainer Module.

The Board level diagnostic procedures given for the faults are common for all computers. The students can follow the procedures listed to service any computer.

The chip level diagnostic procedure given are specific to the CPU Trainer Module and is in accordance to the original PC developed by the IBM corporation. To follow the steps given here one should go through the POST Sequence.

Fault 1 -- CPU LOGIC

FAULT INTRODUCTION : Switch off the system, put 'T1' in open position and switch on the system again.

SYMPTOMS :
1. Speaker not initialized.
2. No display
3. SMPS fan rotates
4. System dead

PROBLEM CAUSE :
1. Power supply faulty
2. CPU, associated circuit may be faulty
3. BIOS EPROM, Bus controller and associated circuitry may be faulty.

BOARD LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Disconnect the SMPS from motherboard and drives. Check +5V and signal	A. +5V is present and power good present B. the voltage <4V or> 5.5V	SMPS O.K. Go to step 2 Error in SMPS rectify and proceed to Step 2
2. Disconnect all the cards from the motherboard Connect SMPS to motherboard turn on the Power.	B. +5V is not available in the I/O slot C. B. +5V is available in I/O slot. No beep.	Possible loose connection in power connector. Check and perform Step 2 again. Motherboard faulty Replace and check

CHIP LEVEL DIAGNOSTIC PRODUCER:

ACTION	OBSERVATION	CONCLUSION
1. Check +5V and power good	A. +5V & Power good present	Go to step 2

signals	B. Power supply faulty	Rectify and proceed
2. Check OSC, CLK and PCLK signals	A. Signals present B. Not present or incorrect frequency	Go to step 3 Check and replace Crystal or 8284
3. Check CPU singles S0, S1, S2, address lines, data lines	A. All signals are pulsing B. Signals not present	Go to Step 4 Check /replace CPU, address/ data buffers
4. Check MEMR, MEMW, IOW, ALE	A. All are present B. Signals not present	Go to Step 5. Check/ replace 8288
5. Arrest bus cycle. Check FFFF0 in address bus & EA in data bus	A. FFFF0 not present in address bus B. EA not present in data bus	CPU/address buffer faulty. Replace and proceed Data buffer/ EPROM faulty. Replace and proceed

Fault 2 -- DRAM LOGIC

FAULT INTRODUCTION : Switch off the system, put 'T2' in open position and switch on the system again.

SYMPTOMS :
1. Speaker not initialized or continuous beep sound. (3 each)
2. No display
3. SMPS fan rotates

PROBLEM CAUSE :
1. CPU may be faulty
2. Data buffer, address buffer may be faulty
3. Memory control section faulty.
4. Memory faulty.

BOARD LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Disconnect SMPS and check voltage level	A. +5V is present	SMPS O.K. Go to step 2
	B. Voltage is <4V or> 5.5V	Error in SMPS rectify and proceed to Step 2
2. Connect SMPS to motherboard and turn power on Check for beep sound.	A. No beep	Motherboard faulty. Replace & proceed
	B. Continuous beep	Memory failure. Replace & proceed

CHIP LEVEL DIAGNOSTIC PRODUCER:

ACTION	OBSERVATION	CONCLUSION
1. Disconnect SMPS and check voltage level	A. Voltage level Ok. B. Incorrect voltage	SMPS O.K. Go to step 2 Rectify & proceed
2. Connect SMPS and turn power on. Check diagnostic post	A. No code B. Code available	CPU logic failure Peripheral ICs failure, Proceed.
3. Check the code	A. 04 and 80 alternately display	Memory failure. As 80 is outer 7th bit fails. Replace memory or check the data buffer.

Fault 3 -- WAIT STATE LOGIC

FAULT INTRODUCTION : Switch off the system, put 'T3' in close position and switch on the system again.

SYMPTOMS :
1. Speaker not initialized.
2. No display.
3. SMPS fan rotates.

PROBLEM CAUSE :
1. CPU, associated circuitry may be faulty.
2. EPROM, bus controller may be faulty
3. power supply may be faulty.

BOARD LEVEL DIAGNOSTIC PROCEDURE:

ACTION	Article I. OBSERVATION	CONCLUSION
1. Disconnect SMPS and check voltage levels	B. Voltage level OK. C. Incorrect Voltage	SMPS Ok. Go to step 2 Rectify SMPS and proceed.
2. Disconnect all cards from the motherboard. Connect SMPS and speaker to motherboard. Turn on the power	A. +5V not available at the I/O slot. B. +5V available at I/O slot. No beep C. Speaker initializes. Beep sound heard.	Possible loose connection in the power connector. Motherboard problem. Replace and proceed. The problem may be due to some add on card.
3. Connect the, add on cards one by one and repeat step 2.	A. Problem repeated when adding a particular card.	The card may be faulty. Replace and proceed.

CHIP LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Check +5V in SMPS	A. Voltage not present B. Voltage is OK	Rectify SMPS and proceed. Go to step 2.
2. Check CLK, OSC and PLK signals	A. Signals present B. Not present or incorrect frequency	Go to step 3. Check / Replace XTAL or 8284.
3. Check CPU signals, address, data status lines.	A. Signals present C. All lines are dead. i. e. high or low permanently	Proceeds as given for fault 1. CPU or control circuitry may be faulty. Proceed.
4. Check the signals, RESET READY etc.	A. Reset and ready o.k. B. Reset not coming C. Ready always low	CPU faulty replaces. Check reset logic. Wait state logic problem, proceed to step 5.
5. Check 10CHRDY signals in I/O slot.	A. Signal always low.	Possibility of 10CHRDY getting shorted with ground. Check and remove.

Fault 4 -- FDC / DMA LOGIC

FAULT INTRODUCTION : Switch off the system, put 'T4' in open position and switch on the system again.

SYMPTOMS :
1. POST takes places.
2. Memory test completed.
3. Flopy drive initialized.
4. System not booting.

PROBLEM CAUSE :
1. BOOT diskette may be faulty
2. Floppy drive cable may be faulty
3. FDD may be faulty.
4. FDC may be faulty.
5. DMA logic in the motherboard may be faulty.

BOARD LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Reboot the system with a good boot diskette	A. System boots B. System not booting	Problem with boot diskette solved Go to step 2.
2. Clean the drive head using a cleaning diskette	A. System boots B. System not booting	Dust problem in read/write head solved Go to step 3 3. Replace the FDD cable and reboot the system
3. Replace the FDD cable and reboot the system	A. System boots B. System not booting	Problem with cable solved. Go to step 4
4. Replace drive and reboot the system	A. System boots B. System not booting	Problem with FDD rectify and solve Problem is in motherboard or FDC card. Replace and Proceed.

CHIP LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Check the FDC interface signals index, Motor on, step direction etc	A. signals OK B. Singals not OK Go to step 2 Check/replace FDD or 8398	1. Check the FDC interface signals
2. Check the signals AEN, DRQ2, DACK2 in the I/O slot.	A. Signals OK B. Signals not OK	Problem may be in the DMA or associtated circuitry. Check & rectify Go to step 3.
3. Check the signals	A. AEN always high or low B. DRQ2 is not pulsing during booting C. DACK2 is not coming during booting.	Bus arbitration logic problem. rectify and proceed. FDC 8398 failure, replace DMA or associated circuitry failure. Check and rectify.

Fault 5 -- FDC LOGIC

FAULT INTRODUCTION : Switch off the system, put 'T5' in open position and switch on the system again.

SYMPTOMS :
1. POST takes places.
2. Memory test completes.
3. Flopy drive not getting initialized.

PROBLEM CAUSE :
1. Floppy disk drive may be faulty
2. Floppy drive cable may be faulty
3. FDC may be faulty.

BOARD LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Check the floppy drive cable and replace. Reboot the system	A. Problem solved B. Not solved	Problem with cable Replace with good one. Go to step 2.
2. Replace the drive and check again	A. Problem solved B. Not solved	Floppy drive fault Service the drive. Floppy drive controller faulty. Replace board and repaet step 2.

CHIP LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. 1. Check the signals TRACK00, SIDE, DIR, MOTOR - A - ON and STEP of 8398.	A. Signals OK B. TRACK00 not coming C. SIDE, STEP, DIR or MOTOR - A - ON signal not coming	8398 OK. Problem may be due to CPU interface in Motherboard. Trace and rectify. Floppy drive faulty. Rectify and Check.

		8398 or associated circuitry faulty. Trace and rectify.
2. Check IOR, IOW, AEN in 8398	A. Signals not proper	Check the signal sources and rectify

Fault 6 -- SERIAL COMMUNICATION LOGIC

FAULT INTRODUCTION : Switch off the system, put 'T6' in open position and switch on the system again.

SYMPTOMS : 1. Serial port does not receive or transmit any information to or from its subsystem or interface units.

PROBLEM CAUSE : 1. Serial port cable may be faulty

2. Serial communication section may be faulty

3. The Voltage path (+12V, -12V) for serial port section may be faulty.

4. The Voltage levels may not be present from SMPS.

BOARD LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Check the serial cable	A. Cable connection OK B. Cable faulty	Goto step 2 Rectify & proceed
2. Check the voltage levels of +12V, 12V in I/O slot	A. Voltage level OK B. Voltage level not proper Go to step 3 Check and rectify the SMPS	2. Check the voltage levels of +12V,
3. Replace the serial port card and check.	A. Problem solved B. Problem not solved	Problem with serial card. Rectify & proceed Problem may be with the interface signals at the I/O slot. Check with another motherboard.

CHIP LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Short 2 and 3 of the serial port connector using debug check the loop back	A. Data send is received. B. Loop back error	Problem is due to the external RS232 cable. Go to step 2
2. Check the signals SEROUT, SERIN of 8250.	A. Signals present during serial transmission B. Signal not present	Go to step 3 8250 and the interface circuitry faulty. Check and rectify.
3. Check the output of 1488, 1489 TXD, RXD	A. Signal present during serial transmission B. No signal	Problem may be with interface cable. 1488, 1489 or the voltage sources problem. Go to step 4
4. Check voltage levels +12V, 12V at the I/O slots.	A. Voltages present B. Voltages not present. Problem with1488, 1489. Replace. Check the SMPS and rectify.	Problem with1488, 1489. Replace. Check the SMPS and rectify.

Fault 7 -- PARALLEL COMMUNICATION LOGIC

FAULT INTRODUCTION : Switch off the system, put 'T7' in open position and switch on the system again.

SYMPTOMS : 1. Printing junk characters.

PROBLEM CAUSE : 1. Parallel port section may be faulty
2. Parallel cable may be faulty
3. Printer may be faulty.

BOARD LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Connect another printer (if possible) and check.	A. Junk printing B. No Problem	Go to step 2 Problem may be with the receiving side of the printer.
2. Connect another cable and check	A. Junk printing B. No problem	Card problem, replace and proceed Centronic cable fault. Replace

CHIP LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Check the parallel port using internal loop back technique.	A. Loop back failure B. Loop back OK	1. Check the parallel port using internal loop back technique.

Fault 8 -- KEYBOARD LOGIC

FAULT INTRODUCTION : Switch off the system, put 'T8' in open position and switch on the system again.

SYMPTOMS :
1. POST starts.
2. Displays keyboard error or 301.

PROBLEM CAUSE :
1. Keyboard may be faulty
2. Switches in the keyboard may be struck.
3. Keyboard logic in motherboard may be faulty.

BOARD LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
3. Check whether XT/AT switch in keyboard is in proper position	A. XT / AT switch is correct B. XT / AT switch not correct	Go to step 2 Put the switch in the proper position
4. Check the keyboard for any struck keys. Also check the keyboard connector	A. Struck keys found B. No struck Keys.	Rectify & proceed Go to step 3
5. Check the system with another keyboard.	A. Problem solved B. Problem exists.	The keyboard is faulty. Rectify The problem may be due to the motherboard check and rectify.

Fault 9 -- VIDEO LOGIC

FAULT INTRODUCTION : Switch off the system, put 'T9' in open position and switch on the system again.

SYMPTOMS : 1. Display scrolls horizontally.

PROBLEM CAUSE : 1. The monitor may be faulty.
2. The signal cable may be struck.
3. The display adapter may be faulty.

BOARD LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
6. Check for any loose connection in the signal cable.	A. Cable O.K. B. Cable loose	Goto step 2 Rectify & proceed.
7. Check with another cable	A. Problem solved. B. Problem not solved	Rectify the display card and proceed. Problem may be due to the monitor rectify and proceed.

CHIP LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Check for any soldering in the display connector	A. Dry soldering present B. No dry soldering	Rectify and check Goto step 2
2. Check for the signals HYSYNC at the connector	A. HSYNC coming B. HSYNC not coming	Problem may be due to the cable or monitor Check the path of the signal and rectify.

Fault 10 -- VIDEO MEMORY LOGIC

FAULT INTRODUCTION : Switch off the system, put 'T10' in open position and switch on the system again.

SYMPTOMS :
1. Display error beep may be coming.
2. Junk displays on the screen.

PROBLEM CAUSE :
1. The system I/O slot data lines may be faulty.
2. The video memory or data line may be faulty.
3. The character generator be faulty.

BOARD LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Check the display connector	A. Loose connection B. No problem	Solve and proceed Go to step 2
2. Replace the video memory ICs and check	A. Problem solved B. Not solved	Problem is due to memory failure. Replace. The problem is due to fault in display card. Replace & proceed.

CHIP LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Check the display card data & address lines	A. Lines are proper B. Signals not	Goto step 2 Check its circuit path and rectify
2. Check the video address and data lines	A. Signals normal B. Signals not normal	Check the I/O slot signals Check its circuit path and rectify

Fault 11 -- TIME OF DAY LOGIC

FAULT INTRODUCTION : Switch off the system, put 'T11' in open position and switch on the system again.

SYMPTOMS : 1. Display 101 code in the monitor.
2. System time is not update.

PROBLEM CAUSE : 1. Timer IC (8253) may be faulty
2. The interrupt logic may be struck.

BOARD LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. The problem is in the motherboard. Replace and check	A. Problem rectified	Service and use the old motherboard.

CHIP LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Check the signal OUT0 of 8253	A. If signal coming B. No signal	Goto step 2 Check the 8253 and associated signals.
2. Check the 8259 IRQ0, INT	A. IRQ0 coming, but INT not generated B. IRQ0 not coming C. IRQ0 and INT generated	8259 problem, replace. Trace the path and rectify. Problem may be due to CPU or control section.

Fault 12 -- TIMER LOGIC

FAULT INTRODUCTION : Switch off the system, put 'T12' in open position and switch on the system again.

SYMPTOMS :
1. No display.
2. Speaker Initialized.
3. SMPS fan rotates.

PROBLEM CAUSE :
1. Power supply may be faulty
2. CPU, associated circuit may be struck.
3. BIOS EPROM, Bus controller and associated circuitry may be faulty.

BOARD LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Disconnect the SMPS from motherboard and drives. Check +5V and power good.	A. +5V is present and power good (+5V) present. B. The voltage is <4V or > 5. 5V	SMPS O.K. Goto step 2 Error in SMPS. Rectify and proceed to step 2.
2. Disconnect all the cards from the motherboard. Connect SMPS to motherboard turn on the power	A. +5V is not available in the I/O slot B. +5V is available in I/O slot. No beep.	Possible loose connection in power connector. Check and perform step 2 again Motherboard faulty. Replace and check.

CHIP LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Check +5V in I/O slot.	A. Voltage level O. K. B. Incorrect Voltage.	Goto step 2. Rectify the SMPS.
2. Check the diagnostic post	A. 02 displayed.	Problem is due to the failure of 8253 IC. Check the IC and associated circuit.

Fault 13 -- DMA LOGIC

FAULT INTRODUCTION : Switch off the system, put 'T13' in open position and switch on the system again.

SYMPTOMS :
1. No display.
2. Speaker initialized.
3. SMPS fan rotates.

PROBLEM CAUSE :
1. SMPS may be faulty
2. CPU and associated circuitry may be struck.
3. Any other peripheral ICs may be faulty.

BOARD LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Disconnect the SMPS from motherboard and drives. Check +5V and power good.	A. +5V is present and power good (+5V) present B. The voltage is <4V or > 5.5V	SMPS O.K. Goto step 2 Error in SMPS. Rectify and proceed to step 2.
2. Disconnect all the cards from the motherboard. Connect SMPS to motherboard. Turn on the power.	A. +5V is not available in the I/O slot. B. +5V is available in I/O slot. No beep.	Possible loose connection in power connector. Check and perform step 2 again. Motherboard faulty. Replace and check.

CHIP LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Check +5V from SMPS.	A. Voltage level O.K. B. Incorrect voltage level.	Goto step 2. Rectify SMPS and proceed.
2. Check the output of diagnostic port.	A. 03 displayed	Problem may be due to DMA failure
3. Check the signals DRQ0, DAK0 of 8237 and OUT 0 of 8253.	A. DRQ0 and DACK0 not coming but OUT 0 comes. B. DRQ0 and DACK0 not coming.	DMA chip problem (8237) Associated circuitry of DMA may be faulty. Trace and rectify.

Fault 14 -- PARITY GENERATION LOGIC

FAULT INTRODUCTION : Switch off the system, put 'T14' in open position and switch on the system again.

SYMPTOMS :
1. System displaying parity error / no display.
2. Speaker initializes/Beep sound for memory error.
3. SMPS fan rotates.

PROBLEM CAUSE :
1. Memory ICs may be faulty
2. Parity generation/checking section may be faulty.

BOARD LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Replace the memory ICs (SIMM) and check	A. System works normally B. Problem persists	Problem with memory module. Replace it. Problem is with the motherboard. Replace & proceed.

CHIP LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Check the diagnostic port output.	A. 04 and 00 alternately displayed in the diagnostic port.	Parity bit mismatch occurred.
2. Replace the memory IC (SIMM) and check.	A. Problem persists.	Goto step 3 Problem with the memory

	B. Problem solved.	module. Replace.
3. Check the signals PCK.	A. A. PCK going high after reset. B. PCK always low.	Problem may be with the 74280, 7474 ICs. Trace and rectify. Check 8255 and interrupt (NMI) logic and rectify.

Fault 15 -- CLOCK GENERATION LOGIC

FAULT INTRODUCTION : Switch off the system, put 'T15' in open position and switch on the system again.

SYMPTOMS :
1. No display.
2. Speaker initialized.
3. System dead.
4. SMPS fan rotates.

PROBLEM CAUSE :
1. CPU or associated circuit may be faulty
2. BIOS EPROM or associated circuitry may be faulty.
3. Power supply may be faulty.

BOARD LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Disconnect the SMPS from motherboard and drives. Check +5V and power good.	A. +5V is present and power good (+5V) present B. The voltage is <4V or > 5.5V	SMPS O.K. Goto step 2 Error in SMPS. Rectify and proceed to step 2.
2. Disconnect all the cards from the motherboard. Connect SMPS to motherboard. Turn on the power.	A. +5V is not available in the I/O slot. B. +5V is available in I/O slot. No beep.	Possible loose connection in power connector. Check and perform step 2 again. Motherboard faulty. Replace and check.

CHIP LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Check +5V and power good	A. +5V power good present. B. Power supply faulty.	Goto step 2. Rectify and proceed.
2. Check OSC, CLK and PCLK signals.	A. Signals present. B. Not present or incorrect frequency.	Goto step 3. Check and replace crystal or 8284.

Fault 16 -- FLOPPY DRIVE LOGIC

FAULT INTRODUCTION : Switch off the system, put 'T16' in open position and switch on the system again.

SYMPTOMS :
1. POST takes place.
2. Memory test over.
3. Floppy drive initialized.
4. System not booting.

PROBLEM CAUSE :
1. The booting diskette may be faulty
2. Floppy drive may be faulty.
3. Floppy drive cable may be faulty.
4. The floppy disk controller section may be faulty.

BOARD LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Check the floppy drive cable and replace. Reboot the system.	A. Problem solved. B. Not solved.	Problem with cable. Replace with good cable. Goto step 2.
2. Replace the drive and check again.	A. Problem solved. B. Not solved.	Floppy drive faulty. Service the drive. Floppy drive controller faulty. Replace board and repeat step 2.

CHIP LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Check the signals TRACK00, SIDE, DIR, INDEX, MOTOR-A-ON and STEP of 8398	A. Signals OK	8398 O.K. Problem may be due to CPU interface in motherboard. Trace and rectify.
	B. Track00 or INDEX signal not coming.	Floppy drive faulty. Rectify and check.
	C. SIDE, STEP, DIR or MOTOR-A-ON signal not coming.	8398 or associated circuitry faulty. Trace and rectify.
2. Check IOR, IOW, AEN in 8398.	A. Signals not proper.	Check the signal source and rectify.

Fault 17 -- FDC BIOS LOGIC

FAULT INTRODUCTION : Switch off the system, put 'T17' in open position and switch on the system again.

SYMPTOMS :
1. POST takes place.
2. FDC BIOS not loaded.
3. System not booting from 1.2Mb / 1.44 Mb drive.

PROBLEM CAUSE :
1. FDC BIOS EPROM may be faulty
2. The control logic may be faulty.
3. System BIOS may be faulty.

BOARD LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Replace the FDD controller and reset the system.	A. FDC BIOS loading B. Problem Persists	Problem with controller card. goto step 2. Problem due to control circuitry in the motherboard/ BIOS. Replace and rectify.
2. Replace the FDC / BIOS EPROM and check.	A. FDC BIOS loading B. Problem persists	Faulty BIOS EPROM. Control circuit of EPROM faulty. Replace card and proceed.

CHIP LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Check the address / data lines of FDC BIOS EPROM.	A. Signals are coming. B. Address or data line not coming	Goto step 2. Trace the path and rectify.
2. Check the chip select signal of EPROM.	A. Chip select always high. B. Chip select coming.	Problem with system BIOS or chip select logic. Trace and rectify. FDC BIOS EPROM faulty. Replace and rectify.

Fault 18 -- RTC LOGIC

FAULT INTRODUCTION : Switch off the system, put 'T18' in open position and switch on the system again.

SYMPTOMS :
1. Setup not stored.
2. System continuously rebooting.

PROBLEM CAUSE :
1. RTC chip may be faulty
2. Battery may be week.
3. Power down circuitry in RTC may be faulty.

BOARD LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Turn off the system and check the voltage level in the battery.	A. Approximately 3.6 V available	Faulty section in motherboard. Replace and rectify.
	B. Battery weak.	Replace battery and proceed.
2. Replace the FDC / BIOS EPROM and check.	A. FDC BIOS loading	Faulty BIOS EPROM.
	B. Problem persists	Control circuit of EPROM faulty. Replace card and proceed.

CHIP LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Check the address / data lines of IC 58167.	A. Address / data line coming B. Address or data line not coming	Goto step 2. Trace the path and rectify.
2. Turn off the system and check for 3.6V at the Vcc pin of 58167.	A. Voltage available B. Voltage not coming	58167 IC faulty. Replace and proceed. Trace the voltage path and rectify.

Fault 19 -- CONTROL LOGIC

FAULT INTRODUCTION : Switch off the system, put 'T19' in open position and switch on the system again.

SYMPTOMS :
1. Speaker initialized or continuous beep sound (3 each).
2. No Display.
3. SMPS fan rotates.

PROBLEM CAUSE :
1. CPU may be faulty
2. Data buffer, address buffer may be faulty.
3. Memory control section may be faulty.
4. Memory may be faulty.

BOARD LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Disconnect SMPS and Check voltage level	A. +5V is present. B. Voltage is < 4V or > 5.5V	SMPS O.K. go to step 2 Error in SMPS. Rectify and proceed to step 2.
2. Connect SMPS to motherboard and turn power on. Check for beep sound.	A. No beep. B. Continuous beep	Motherboard faulty. Replace and proceed. Memory failure. Replace and check.

CHIP LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Disconnect SMPS and check voltage level.	A. Voltage level O.K. B. Incorrect voltage.	SMPS O.K. Goto step 2. Rectify and proceed.
2. Connect SMPS and turn power on. Check the diagnostic port.	A. No code B. Code available	CPU logic failure Peripheral ICs. Proceed.
3. Check the code	A. 04 and FF alternately displayed	All bits fails in memory.
4. Replace the memory and Check.	A. Problem persists B. System working normally.	Goto step 5 Fault with the memory module. Replace.
5. Check the signals MSMR, MEMW, RAS, CAS.	A. MEMR, MEMW, RAS, or CAS not generated. B. All the signals are coming.	Trace the circuit path and rectify Check the address / data lines, Multiplexed and rectify.

Fault 20 -- BUS ARBITRATION LOGIC

FAULT INTRODUCTION : Switch off the system, put 'T20' in open position and switch on the system again.

SYMPTOMS :
1. Speaker initialized.
2. No Display.
3. One long and two short beeps for display adapter failure.

PROBLEM CAUSE :
1. Display adapter may be faulty
2. Interface circuitry in motherboard may be faulty.
3. Bus controller or associated circuitry may be faulty.

BOARD LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Check the display adapter for any loose connection. Clean the card and reinsert.	A. System working. B. Problem presists.	Loose connection solved. Goto step 2.
2. Replace the display card and check.	A. System working B. Problem presists.	Display card faulty. Check the jumper setting in the card or service. Problem due to motherboard. Replace and check.

CHIP LEVEL DIAGNOSTIC PROCEDURE:

ACTION	OBSERVATION	CONCLUSION
1. Check the address / data lines in the I/O slot.	A. All signals are O.K. B. Address or data line not coming	Goto step 2. Check the buffers / drivers and rectify.
2. Check the signals AEN, OSC, CLK at I/O slot.	A. AEN signal not coming. B. OSC, CLK not coming.	Check the bus controller signals and associated control circuitry. Trace the circuit and rectify.

CHAPTER 7

EXPERIMENTAL

7.1 Introduction:

In computing a benchmark is the result of running a computer program, or a set of programs, in order to assess the relative performance of an object, by running a number of standard tests and trials against it. The term, benchmark, is also commonly used for specially-designed benchmarking programs themselves. Benchmarking is usually associated with assessing performance characteristics of computer hardware, e.g. the floating point operation performance of CPU, but there are circumstances when the technique is also applicable to software. Software benchmarks are, for example, run against compilers or database management systems

Benchmarks provide a method of comparing the performance of various subsystems across different chip / system architectures. Once the province of a limited number of technicians and engineers, benchmarking has exploded in popularity in recent years. It seems impossible of late to see anyone talking about their MOTHERBOARD without also quoting benchmarks. There also seems to be a new benchmarking program released every month. The proliferation of benchmarking software and benchmarking comparisons on the Internet has also, unfortunately, led to benchmarking increasingly being used incorrectly and even with intent to deceive. Let us look at benchmarking in the modern PC world and discusse some of the pitfalls of using benchmarks inappropriately

As motherboard architecture advanced, it becomes more and more difficult to compare the performance of various motherboard simply by looking at their specification. Therefore, tests were developed that could be performed on different systems, allowing the results from these tests to be compared across different architectures. For example, Intel Pentium 4 processors have a higher hertz rating than AMD Athlon XP processors for the same computational speed, in other words a ‘slower’ AMD processors could be as fast on benchmark tests as a higher hertz rated Intel processors. Motherboard architecture is the theory behind the design of a computer. The Pentium 4 is a seventh-generation x86 architecture microprocessor produced by Intel and is their first all-new CPU design since the Pentium Pro of 1995. Athlon is the brand name applied to a series of different x86 processors designed and manufactured by AMD. The original Athlon, or Athlon Classic was the first seventh generation x86 processor and, in

a first, retained the initial performance lead it had over Intel's competing processors for a significant.

Benchmarks are designed to mimic a particular type of workload on a component or system. "Synthetic" benchmarks do this by specially-created programs that impose the workload on the component. "Application" benchmarks, instead, run actual real-world programs on the system. Whilst application benchmarks usually give a much better measure of real-world performance on a given system, synthetic benchmarks still have their use for testing out individual components, like a hard disk or networking device, Typical hard drives of the mid-1990s.

Motherboard manufacturers have a long history of trying to set up their systems to give unrealistically high performance or benchmark tests that is not replicated In real usage. For instance, during the 1980s some compilers could detect a specific mathematical operation used in a well-known floating-point benchmark and replace the operation with a mathematically-equivalent operation that was much faster. However, such a transformation benchmarks (or aspects of benchmarks) show their products in the best light. They also have been known to mis-represent the significance of benchmarks, again to show their products in the best possible light. Taken together, these practices are called bench marketing.

7.2 Benchmarks software

Computer benchmarking was originally the realm of computer scientists and systems engineers. By running tests designed specifically to test the operation of a particular component, researchers could measure how well a new design compared to previous designs in terms of speed or throughput. The values were given as numbers due to the fact that digital computations were being measured. By understanding the component and benchmark design, intelligent conclusions could be made about the effects of various features and design changes. Later, marketing departments discovered that benchmarks could be used to "prove" that their product was "faster" than their competitors', usually by choosing the benchmark tests and results that were the most favorable for their purposes.

In an attempt to provide standardization, several organizations were formed that developed their own benchmark tests. Some of these were formed as non-profit organizations that received funding from manufacturers for the purpose of developing these benchmark programs. Most of these programs were offered as source code to be compiled by each manufacturer, making them more platform- and operating-system-independent. It didn't take very long before manufacturers realized that they could optimize their product or drivers for these benchmarks, and even use special compiler flags to take advantage of their particular hardware features.

In order to combat the "bench marketing" performed by manufacturers, new versions of these benchmarks are released on a periodic basis, though this does not completely eliminate the criticisms. Many of these programs are offered for "free" (though sometimes with an "administrative" fee), while some are run only by the organization itself to ensure the results are not skewed. Since the inception of the World Wide Web on the Internet, the free benchmark programs have become available to anyone, causing a virtual explosion of benchmark activity by end users.

If you have spent any amount of time on the hardware-related Usenet newsgroups, you will see numerous benchmarks posted by users, usually to find out if their system compares favorably to others running similar configurations. Most of these users have little or no knowledge of what the programs are really testing, nor how to evaluate the results. This has caused a lot of confusion and misinformation to be spread. Users regularly post questions about why their system-level benchmark score was lower than someone else's when they both had the same CPU. These users are usually completely confused when told that their video card or hard drive (or other component) could be affecting the result.

Computer hardware sites are becoming more prevalent, with many providing recommendations on what component is the "best of class", usually based upon the results of some limited benchmark testing. Most run a commonly used benchmark simply for the purpose of having a benchmark result to display. Usually these sites have not clearly identified exactly what they are trying to test, nor do they provide all of the

pertinent details about the system being tested. This is like asking the question of which car is faster, then running them for a quarter mile and choosing a winner. If you wanted to know which one was faster on mountain roads, you cannot come to a reasonable conclusion because you have not run the right test to answer that question.

There are even sites whose entire purpose is to gather user benchmark results, ostensibly so users can determine if their system is running optimally. Some of these sites do not go verification of results, and only require a limited amount of information about the system that was tested. When you then consider that users may have differing levels of drivers and widely varying BIOS settings, you can easily see why making any kind of comparisons from the results can be somewhat misleading. The more professional benchmarking sites, such as SPEC, require detailed hardware/software information and have a validation process in an attempt to prevent fraudulent scores from being submitted.

The most prevalent misuse of a benchmark score is the fascination so many have with Quake frame rate results. The Quake benchmark is the absolute best benchmark for testing the Quake performance of a system. It also is useful for determining performance of other games that use the Quake engine on a given machine. The problem is that the benchmark is also being used as a measure for floating point performance (which it is not designed to do), and even the overall "value" of a processor or system.

It is also interesting to note that for 3D games, an important metric is the *slowest* frame rate because that determines the performance during the most graphics intensive parts of the application. The Quake benchmark does not provide this information, but instead gives only the average. It is fairly obvious that two systems could have the same average, yet have different ranges of frame rates. The net result would be that the system with the smallest variance between frame rates would be the one with the best performance in actual game play.

Some of the popular benchmark softwares are as follows

- HINT
- Fhourstones
- Dhrystone
- Whetstone
- Standards Performance Evaluation Corporation Performance-Evaluation-Corporation (SPEC)
- BA Pco
- 3Dmark
- Quake
- Khornerstone
- TPC
- Linpack
- Aquamark
- John the Ripper

CHAPTER 8

RESULT AND DISCUSSION

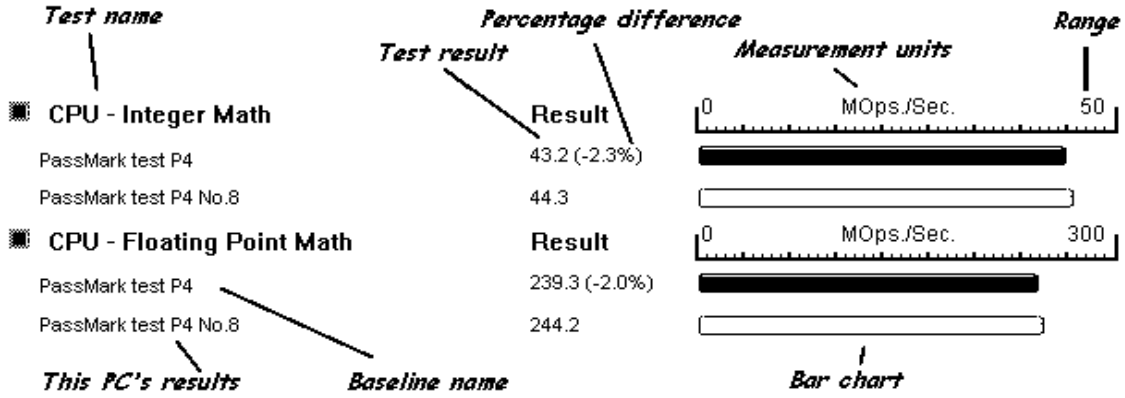
8.1 IMPORTANT FEATURES OF THE BENCH MARK SOFTWARE

Performance Test is a fast, easy to use software benchmarking tool that allows everybody to quickly assess the performance of their PC and compare it to a number of standard 'baseline' computer systems.

PerformanceTest allows you to,

- Find out if your PC is performing at its best.
- Compare the performance of your machine to similar machines.
- Measure the effect of configuration changes and upgrades.
- Avoid paying big bucks for poor performance.
- Make objective independent measurements on which to base your purchasing decision.

8.2 Interpreting the result



Measurement Unit

The result of each test is presented using a particular unit of measurement. In each case the higher the number, the higher the performance of the computer. For some tests there is a differentiation between integer and floating point numbers. Integers are whole numbers such as 23, 459532, -26. Floating point numbers contain a fractional part, eg. 1.003, 98394.2. These two types of numbers are treated quite differently inside a computer, which is why Performance Test differentiates between the two.

MBytes / Sec Megabytes transferred per second

- Used to measure the amount of information transferred to and from a disk or memory.

Add’s / Sec Millions of additions per second

- Used to measure the number of integer or floating point mathematical operations.

Subs's / Sec Millions of subtractions per second

- Used to measure the number of integer or floating point mathematical operations.

Mult’s / Sec Millions of multiplications per second

- Used to measure the number of integer or floating point mathematical operations.

Div’s / Sec Millions of divisions per second

- Used to measure the number of integer or floating point mathematical operations.

Lines / Sec Thousands of lines per second

- Used to measure the number of lines that can be drawn in a window per second. This is a measure of the 2D graphics performance.

Images / Sec Thousands of bitmap images per second

- Used to measure the number of bitmap images that can be drawn in a window per second. This is a measure of the 2D graphics performance.

Shapes / Sec Thousands of shapes per second

- Used to measure the number of shapes that can be drawn in a window per second. This is a measure of the 2D graphics performance.

Frames / Sec Number of different image frames displayed per second

- Used to measure the performance of animation using 3D graphics. High frame rates produce a smoother, more realistic picture.

MegaFLOPS (or FLOPS)

FLOPS is the number of floating point operations that can be performed per second. As this figure is normally in the millions the term MegaFLOPS is used. 1,000,000 FLOPS = 1 MegaFLOPS. This is an industry standard measurement, that is used to compare the performance of diverse computer systems.

- Composite Average

This is an extra heading used for the summary section. It doesn't really represent a particular unit of measurement as it is a combination (weighted average) of several other measurements.

8. 3 Discussion about different test

CPU test suit

Test suite – CPU:- This suite contains multi-process CPU tests. The number of CPU test processes is configurable in Preferences.

The following tests make up the suite.

- Integer (32-bit and 64-bit addition, subtraction, multiplication and division)
- Floating Point (32-bit and 64-bit addition, subtraction, multiplication and division)
- SSE (128-bit SSE operations such as addition, subtraction and multiplication) or 3Dnow! (3D transformation of an array of 32-bit vertices). The SSE test will run if SSE is supported, if not, the 3Dnow! test will run if 3Dnow! is supported. The 3Dnow! test is not available on 64-bit Windows platforms.
- Comperision
- Encryption
- Image Rotation (Rotate image co-ordinates in memory)
- Random String Sorting
- Find Prime numbers

For each of the integer tests, a large array of random 32 bit and 64 bit integers is processed using the particular mathematical operations listed below. For the floating point tests, single precision floating point numbers (32 bit and 64 bit) are used. Figure 8.1 to Figure 8.8 shows CPU test suit performance chart and comparison for different types of CPUs.

2D graphics test suit

Test suite – Graphics: - This suite contains a number of tests that exercise the standard Windows graphics functions. The results from this suite depend on the speed at which the video card can carry out 2D graphics operations and the color depth currently in use.

Line Drawing: - Lines are drawn into window. The color is changed every 500 lines to enable the lines to be seen in the test window.

Painting Bitmaps: - A bitmap image is painted into a window as quickly as possible. It should be noted that due to the queuing of display requests by hardware accelerated video cards, the duration of this test might be longer than expected.

Outline shapes: - Two different shapes are drawn into a window

- An ellipse
- A square with rounded corners

The color is changed every 64 shapes to enable the shapes to be seen in the test window.

Fonts and Text: - Tests the performance of the graphics card with the typical rendering of Fonts and Text.

GUI: - Test the performance of the graphics card and the Windows Display settings for interacting with the Graphical User Interface. The test includes the performance measurement of Common GUI controls: treeview, listview, sliders and edit boxes, as well as window movement and resizing. Figure 8.9 to Figure 8.13 shows 2D graphics test suit performance chart and comparison for different types of CPUs.

3D graphics test suit

Test suite – 3D Graphics: - This test suite attempts to measure the performance of the 3D graphics hardware installed in the machine. This test suite makes full use of version 9 of the Microsoft DirectX 3D graphics library. Without DirectX 9.0 or above it will not be possible to run these tests.

Three standard tests have been defined: Simple, Medium, and Complex. Ultimately all these performances depend on the architecture and performance of the motherboard because all devices are connected internally through motherboard. So, this is very simple way to test the motherboard as per the requirement. Figure 8.14 to 8.16 shows 3D graphics test suit performance chart and comparison for different type of CPUs.

Memory test suit

Test suite - Memory: - This suite contains a number of tests that exercise the memory sub-system of the computer (Random Access Memory- RAM). All tests use a combination of 32-bit and 64-bit data when reading or writing from or to RAM.

Memory - Allocate small block: - This test measures the time taken to allocate & free small zeroed memory blocks (around 100KB block size).

Memory – Cached: - This test measures the time taken to read a small block of memory. The block is small enough to be held entirely in cache (if one is present).

Memory - UnCached: - This test measures the time taken to read a large block of memory. The block is too large to be held in cache.

Memory - Write: - This test measures the time taken to write information into memory.

Memory – Large RAM: - This test measures the ability to allocate very large amounts of RAM and the time taken to read this RAM. The test is designed to measure the ability of the system to support applications that use very large amounts of RAM.

More detailed results: - By using the advanced memory testing window it is possible to examine memory performance in more detail. Figure 8.17 to 8.21 shows memory test suit performance chart and comperision for different types of CPUs.

Disk test suit

Test suite – Disk: - This suite contains a number of tests that exercise the mass storage units (hard disk or otherwise) connected to the computer. By default drive C: is used but this can be changed from the Preferences Dialog.

For each test a file is created in the root directory of the selected disk. The file size needs to be large in order to get an accurate measurement. The test file size is 200MB and the read or writes block sizes used are 16KB. Under Windows 2000 and above, each test uses uncached asynchronous file operations (with an IO queue length of 20). Under Windows 98 and Windows ME, each test uses uncached synchronous file operations. Each test runs for at least 20 seconds. See advanced disk testing window for descriptions of terms.

There are a few issues to aware of when interpreting the results of the disk test. These are covered in the precautions section.

Disk Sequential Read: - A large test file is created on the disk under test. The file is read sequentially from start to end.

Disk Sequential Write: - A large file is written to the disk under test. The file is written sequentially from start to end.

Disk Random Seek RW: - A large test file is created on the disk under test. The file is read randomly; a seek is performed to move the file pointer to a random position in the file, a 16KB block is read or written then another seek is performed. The amount of data actually transferred is highly dependent on the disk seek time. Figure 8.22 to 8.24 shows disk test suit performance chart and comperision for different type of CPUs.

CD /DVD test suit: - The CD/DVD test suite is made up of a single test, which reads data from the CD drive selected in the preferences window.

To use this test you need to have a CD in the CD Drive. The CD should, if possible, have mostly large files on the disc. Having many small files on the disc will lower the result, as the time required to seek between files is included in the result. Figure 8.25 shows CD/DVD test suit performance chart and comperision for different types of CPUs.

More detailed results: - By using the advanced disk testing window it is possible to examine the performance of the disk in more detail. Figure 8.26 to 8.32 shows advance disk test suit performance chart and comperision for different types of CPUs.

8.4 Performance Chart & Comparison of Different Motherboards:-

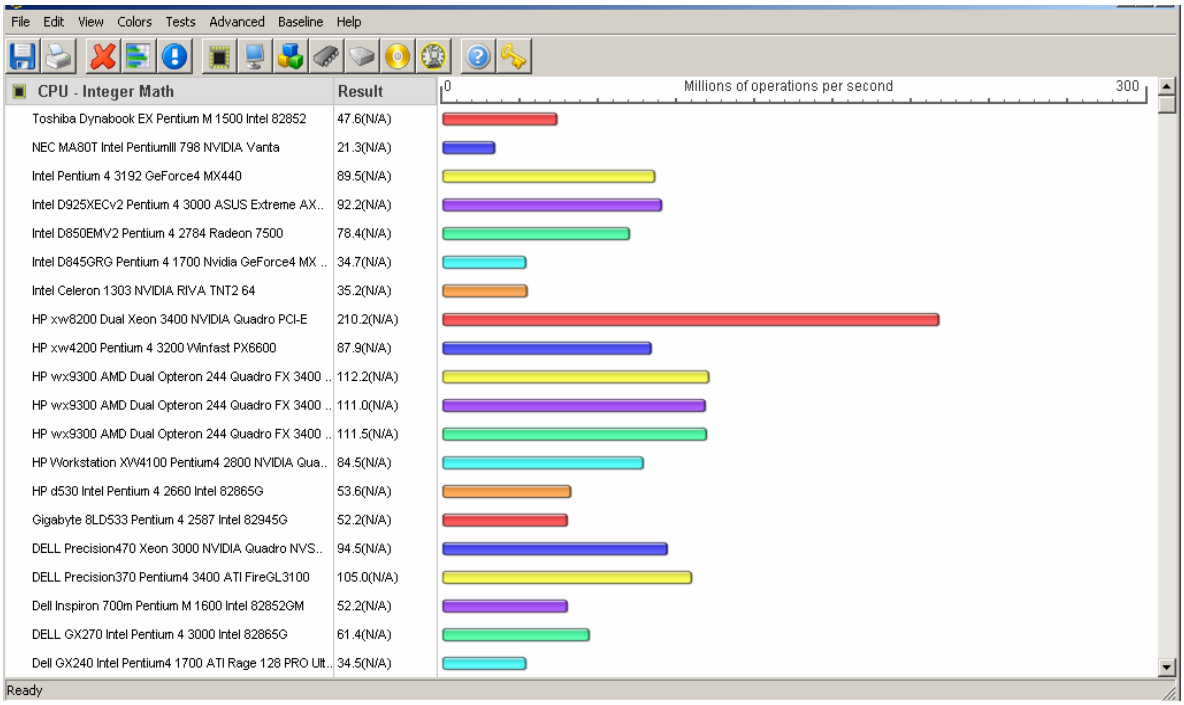


Figure 8.1 CPU PERFOMANCE FOR INTEGER MATHS

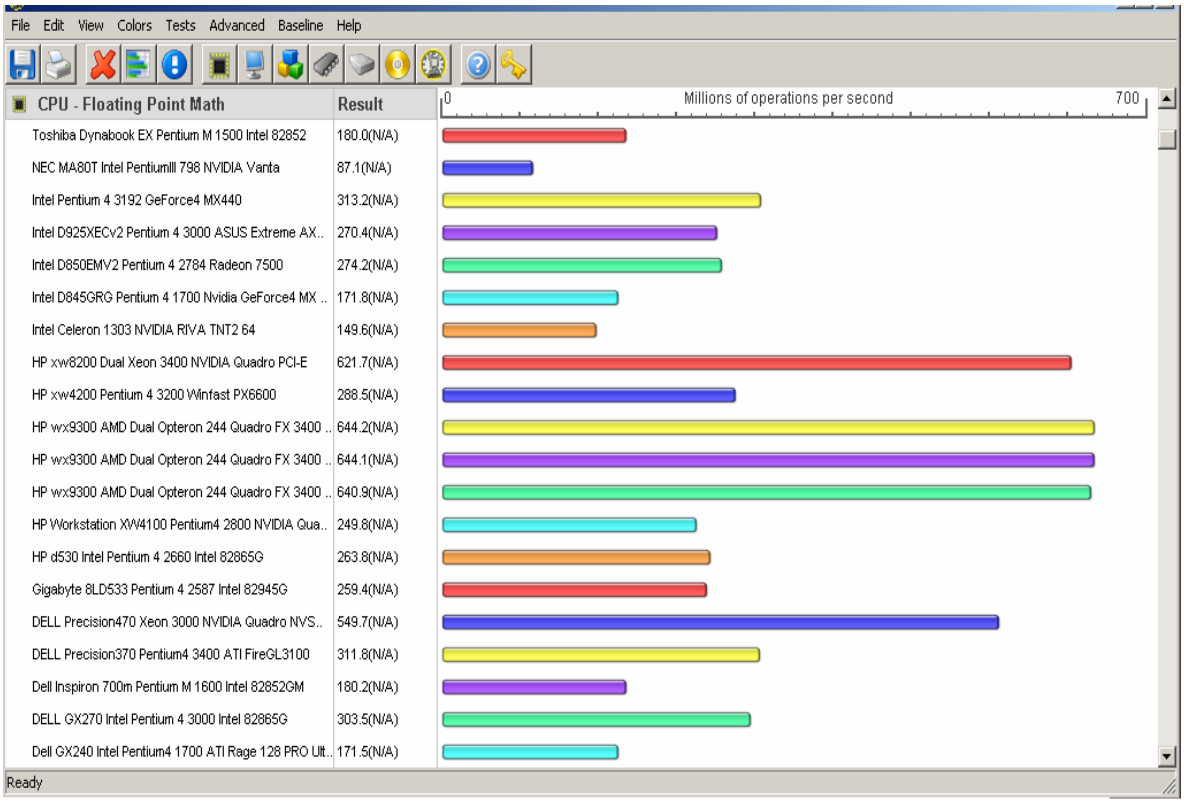


Figure 8.2 CPU PERFOMANCE FOR FLOATING POINT MATHS

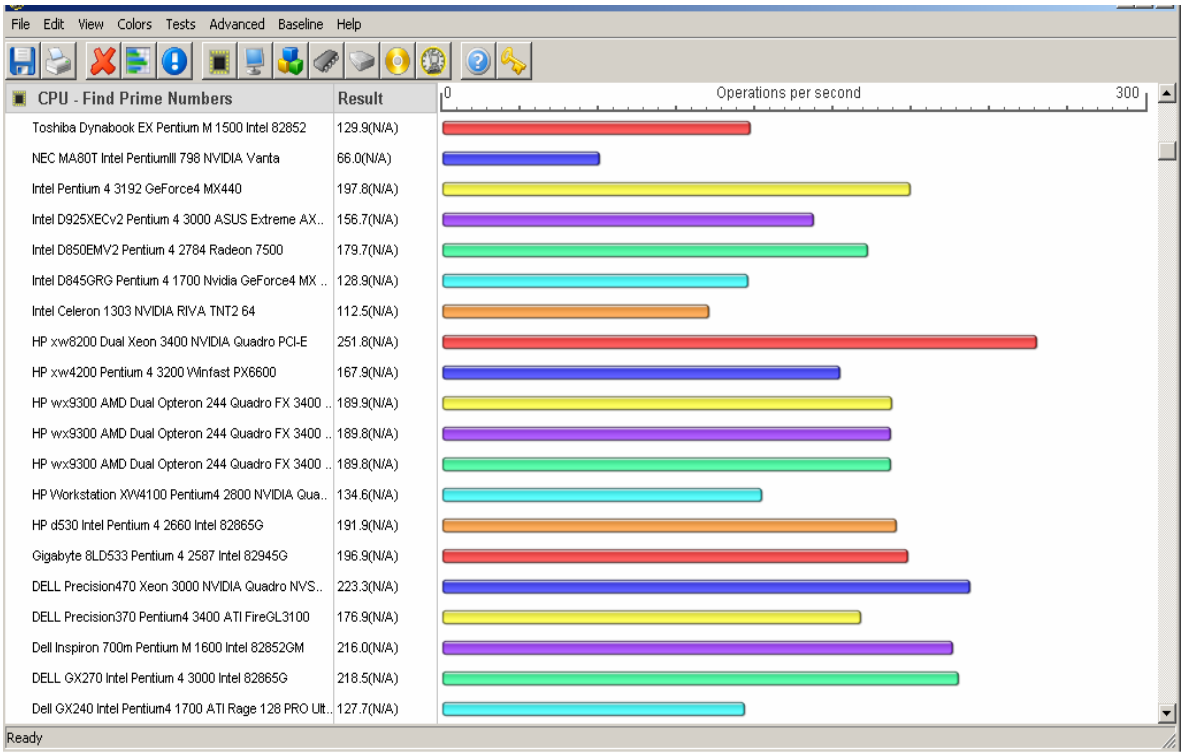


Figure 8.3 CPU PERFOMANCE TO FIND PRIME NUMBERS

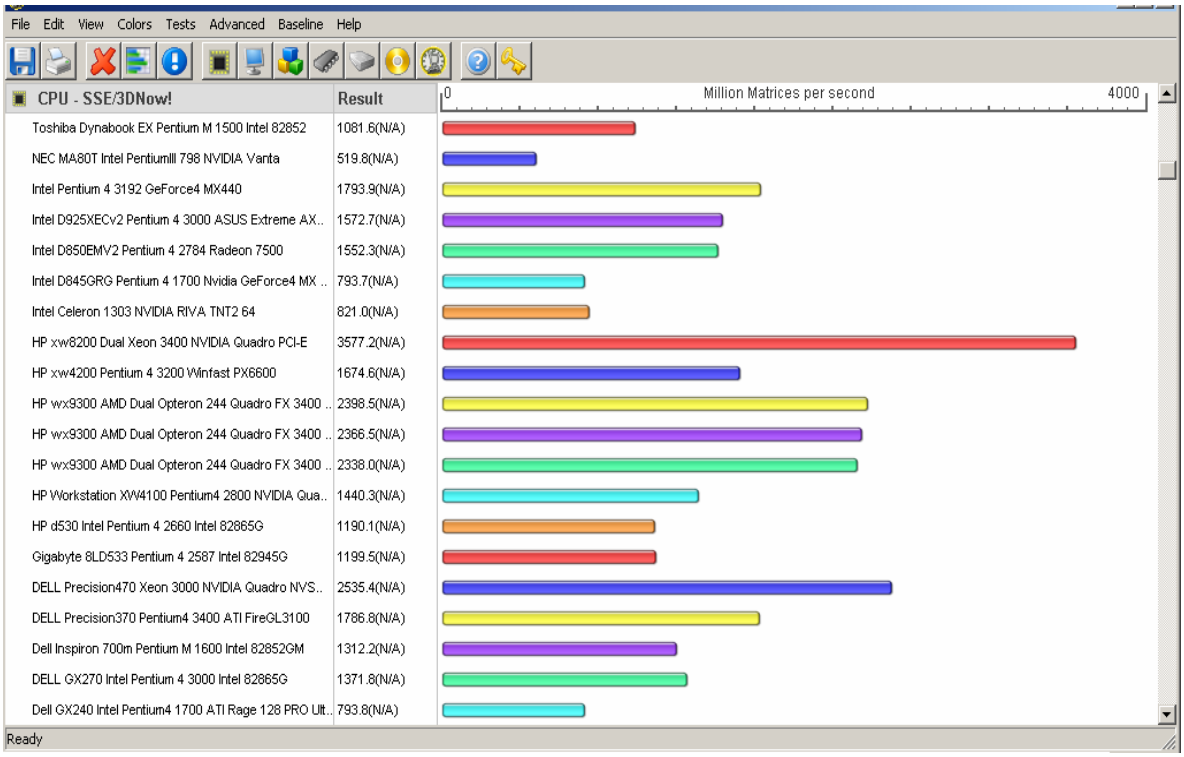


Figure 8.4 CPU PERFOMANCE FOR SSE/3DNow!

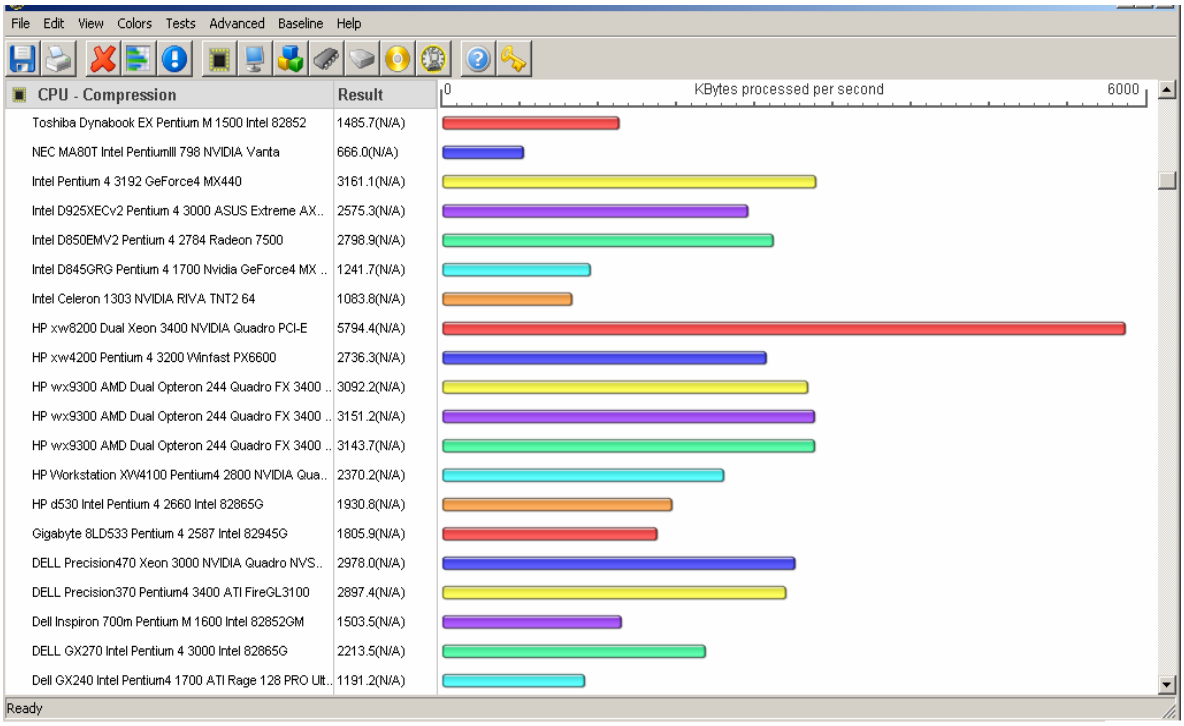


Figure 8.5 CPU PERFOMANCE FOR COMPERISION

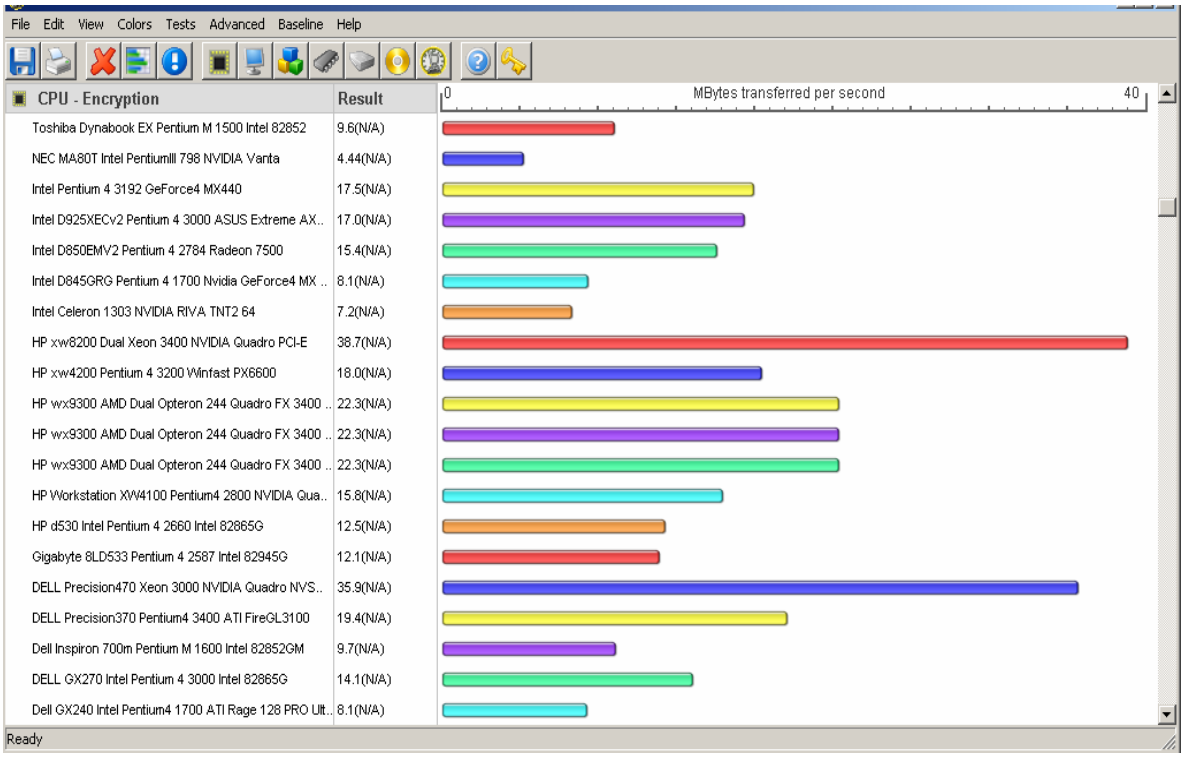


Figure 8.6 CPU PERFOMANCE FOR ENCRYPTION

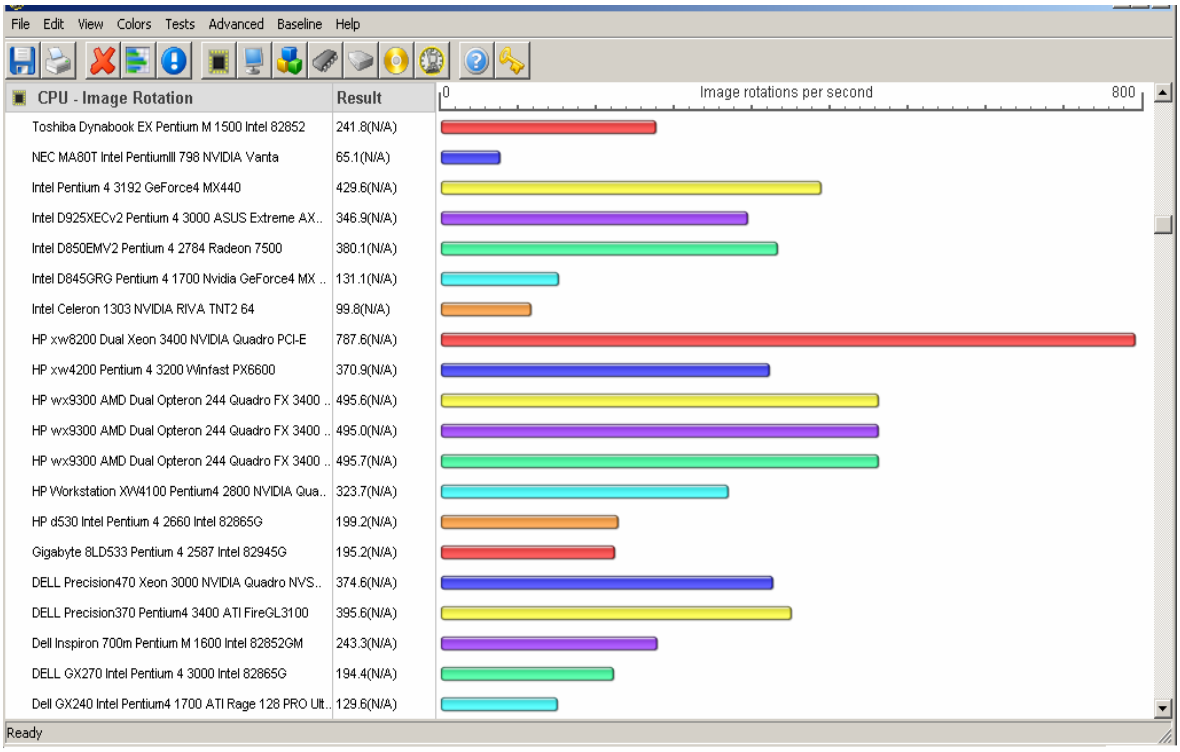


Figure 8.7 CPU PERFOMANCE FOR IMAGE ROTATION

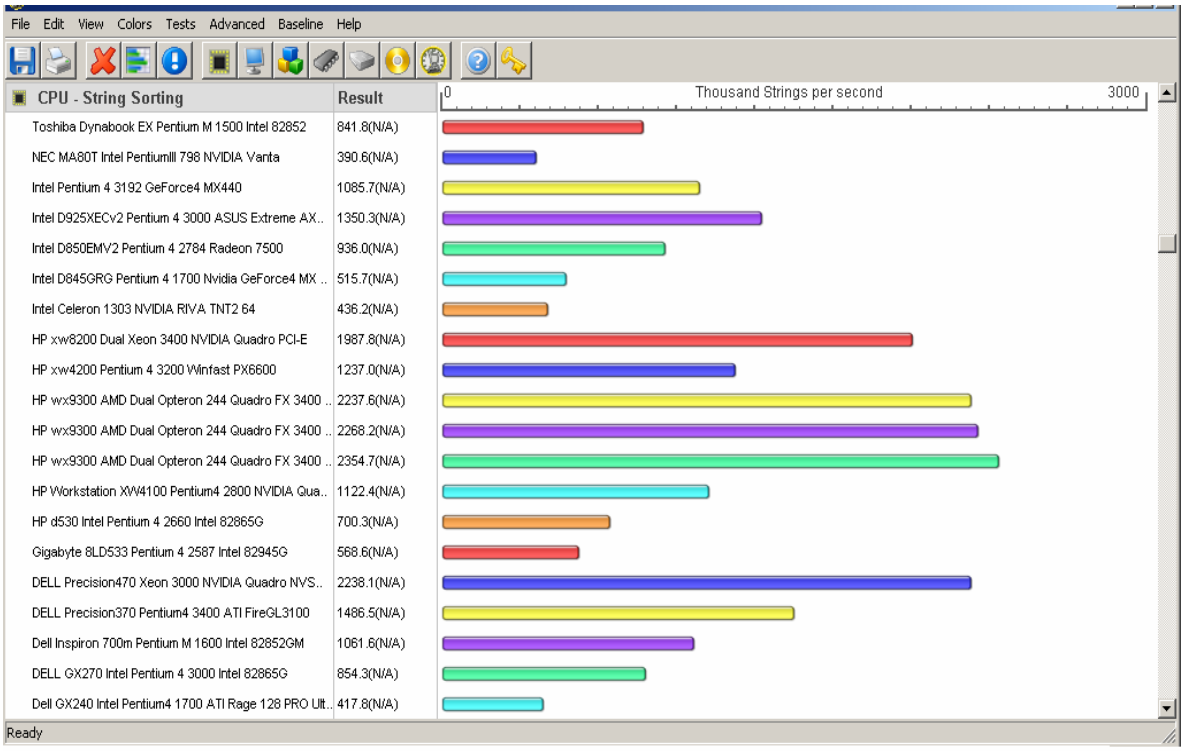


Figure 8.8 CPU PERFOMANCE FOR STRING SORTING

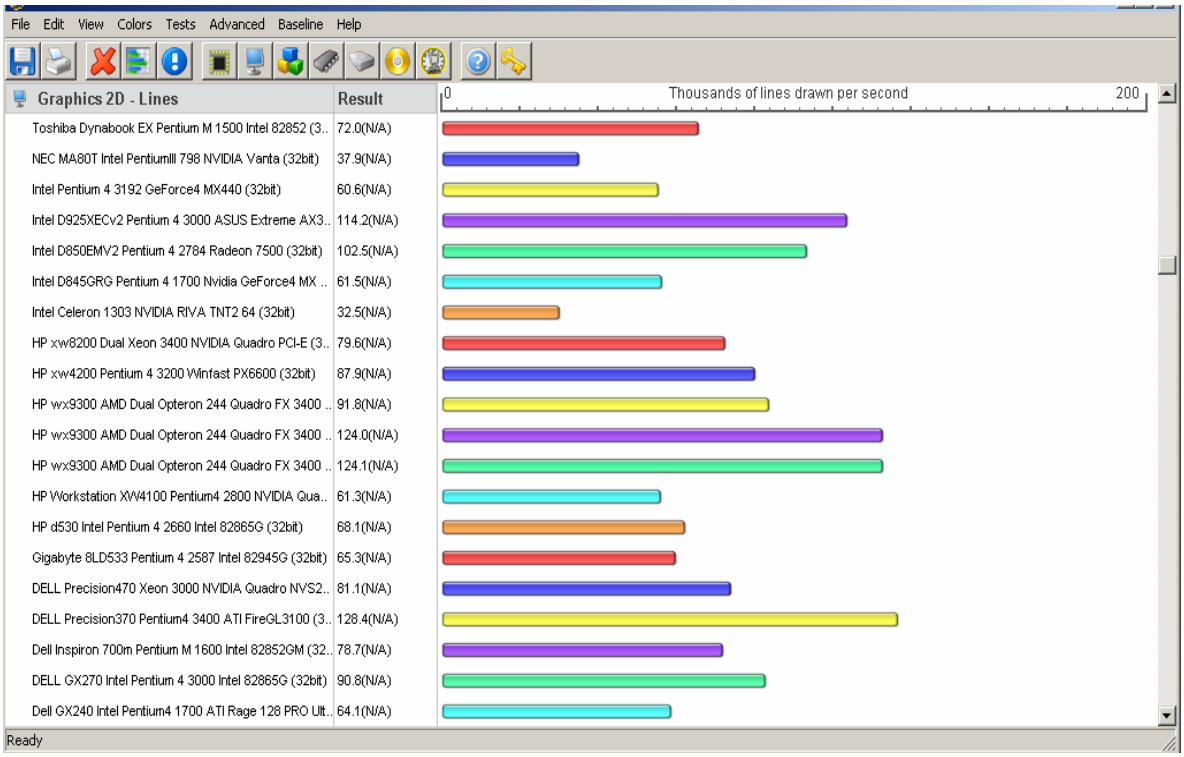


Figure 8.9 CPU PERFOMANCE FOR GRAPHICS 2D-LINES

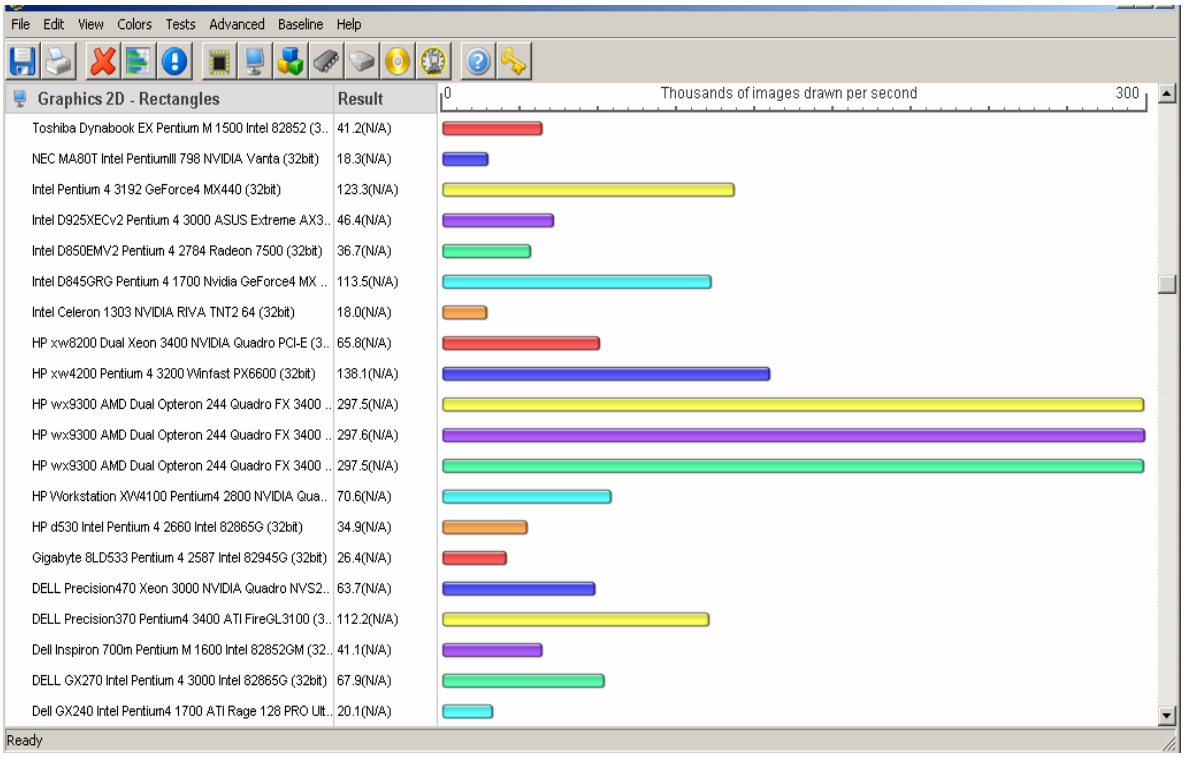


Figure 8.10 CPU PERFOMANCE FOR GRAPHICS 2D-RECTANGLE

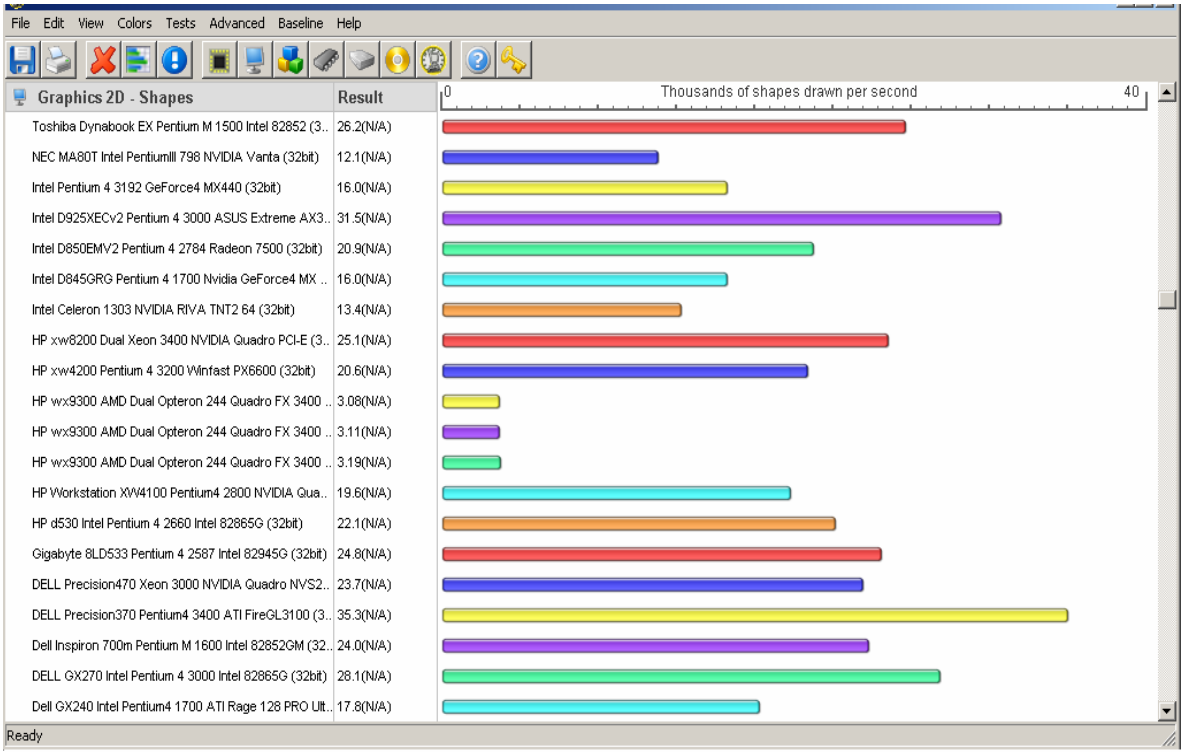


Figure 8.11 CPU PERFOMANCE FOR GRAPHICS 2D-SHAPES

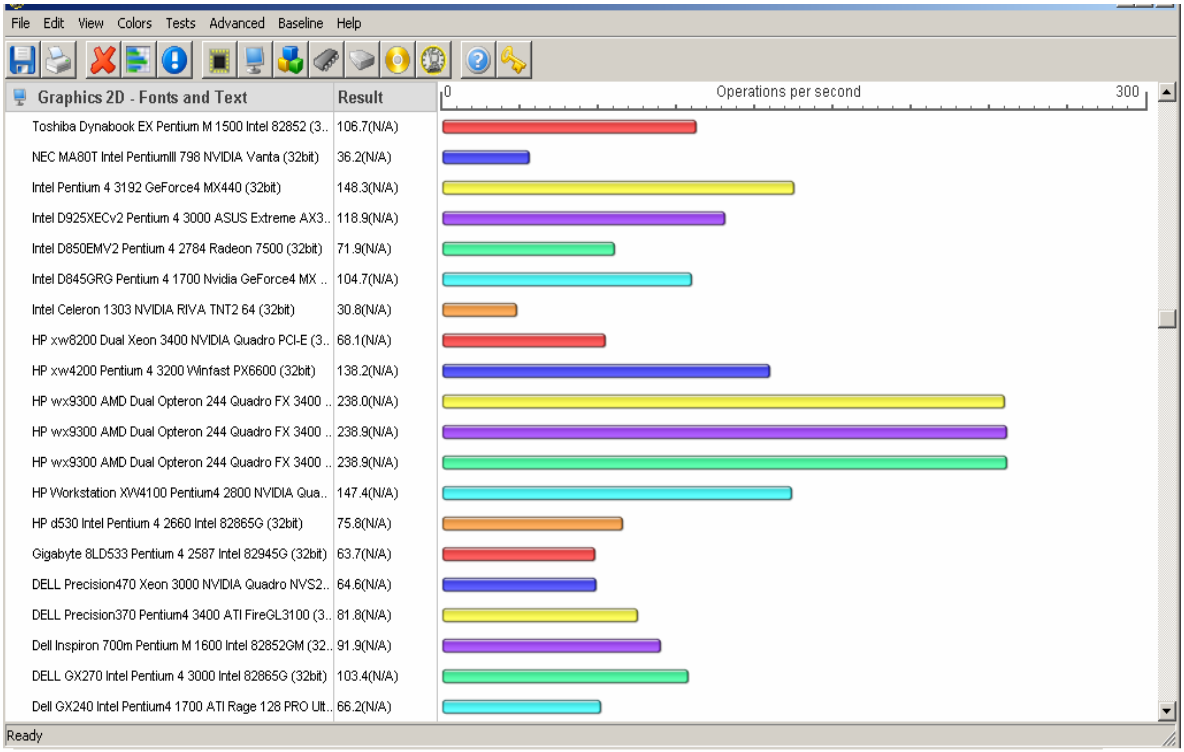


Figure 8.12 CPU PERFOMANCE FOR GRAPHICS 2D-FONTS AND TEXT

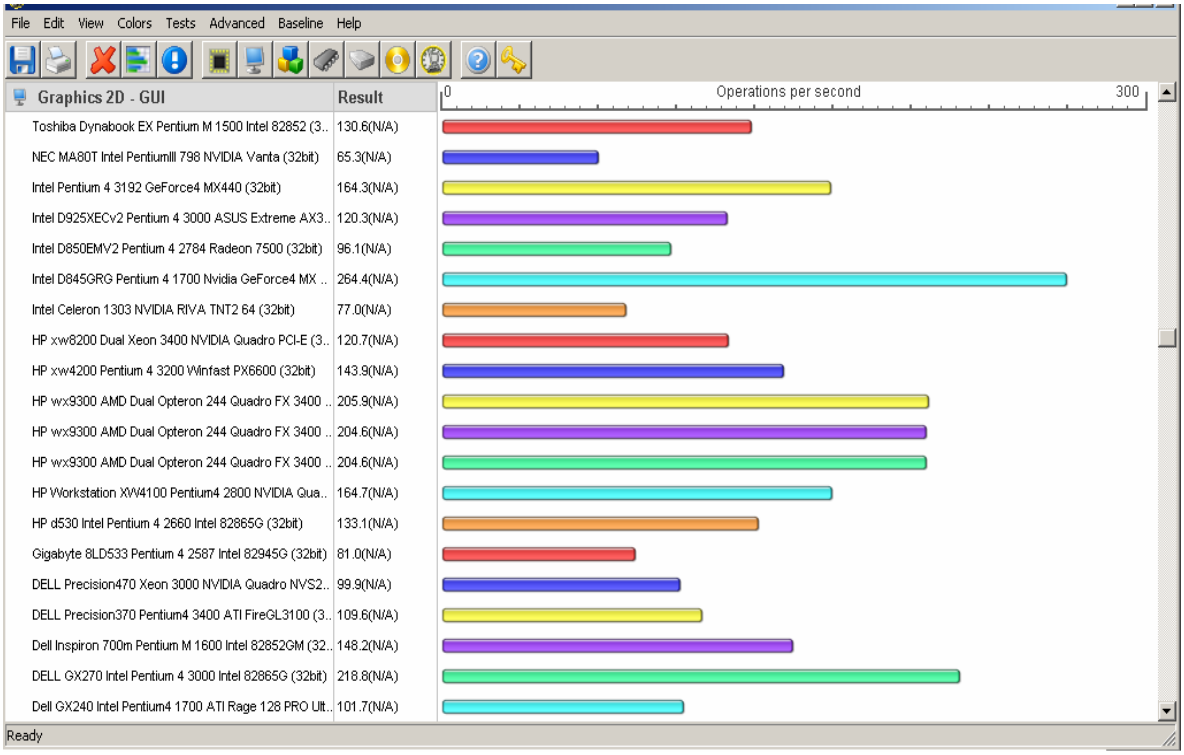


Figure 8.13 CPU PERFOMANCE FOR GRAPHICS 2D-GUI

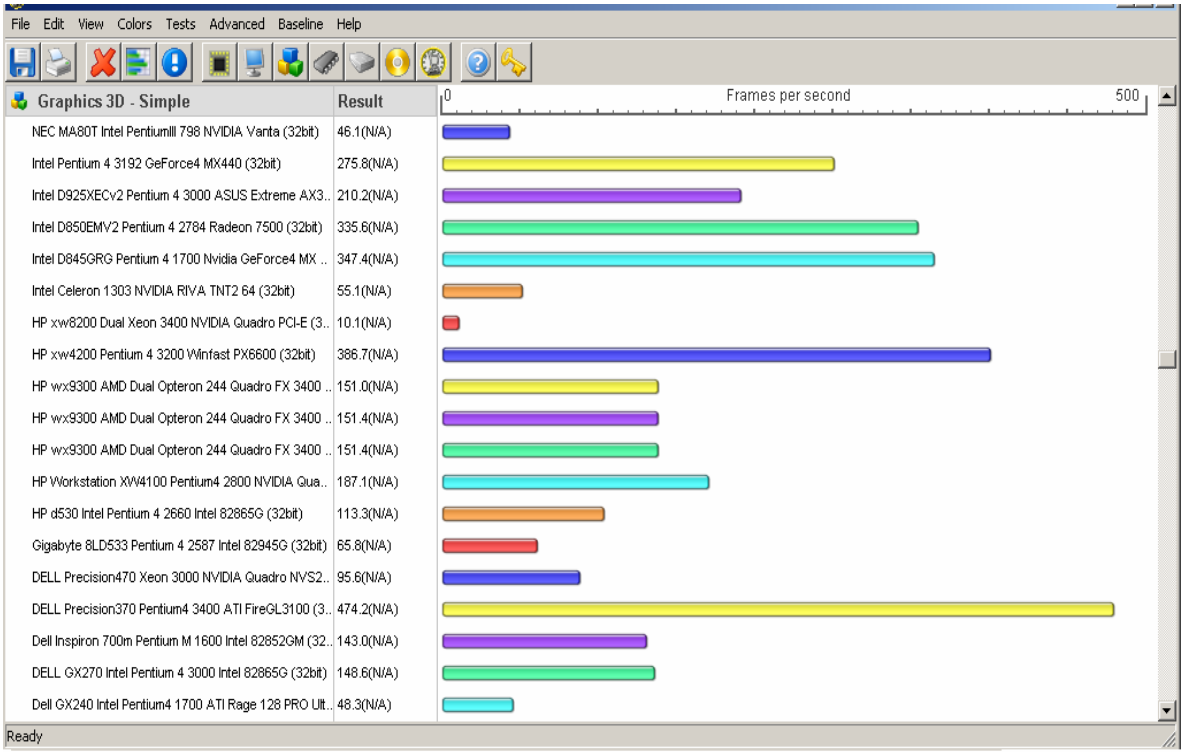


Figure 8.14 CPU PERFOMANCE FOR GRAPHICS 3D-SIMPLE

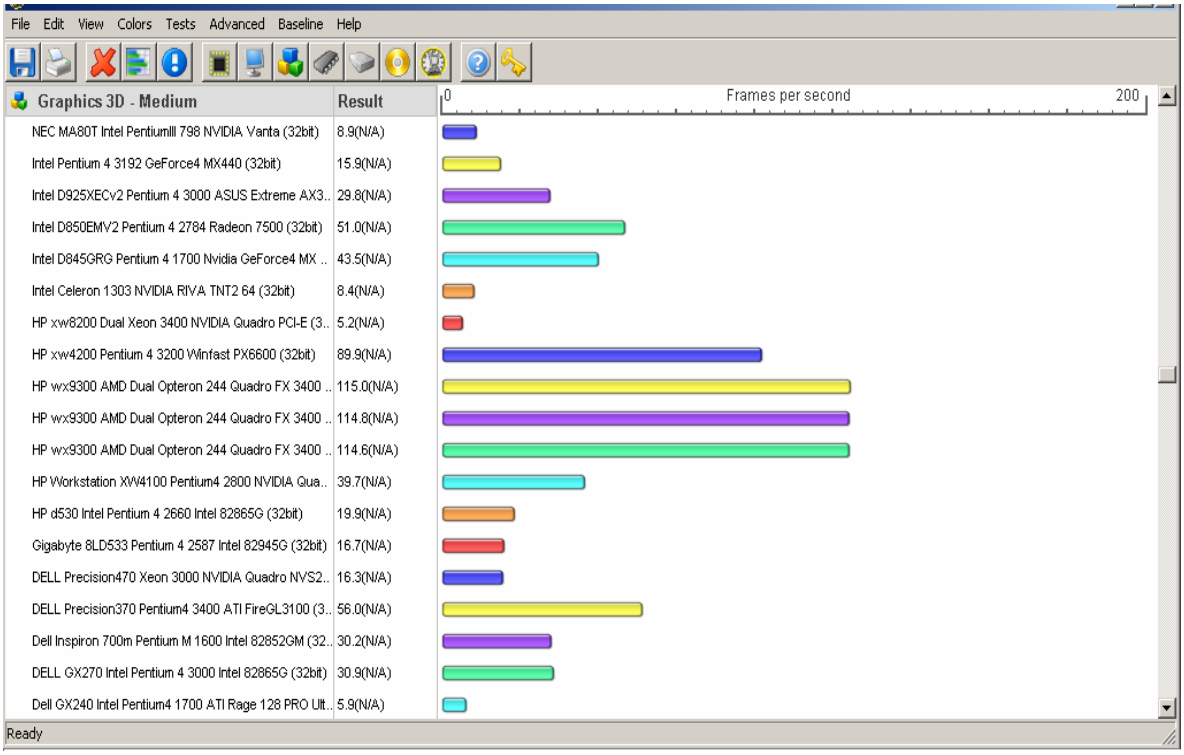


Figure 8.15 CPU PERFOMANCE FOR GRAPHICS 3D-MEDIUM

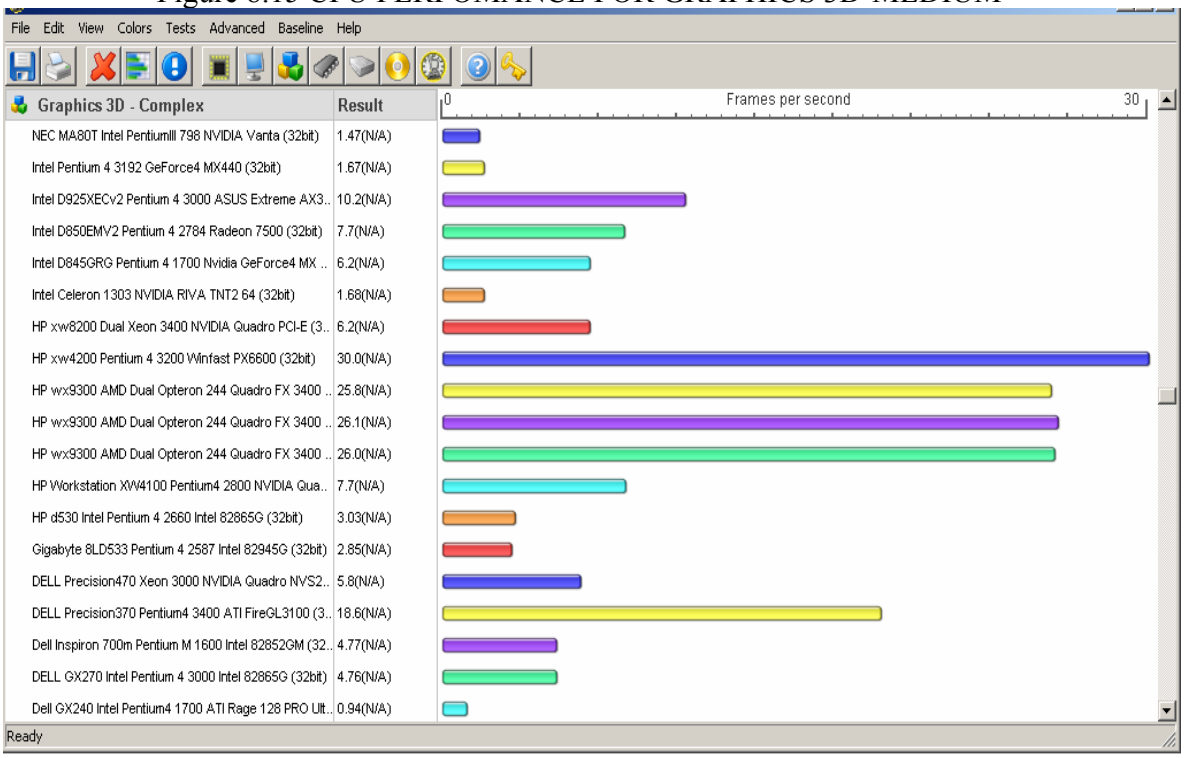


Figure 8.16 CPU PERFOMANCE FOR GRAPHICS 3D-COMPLEX

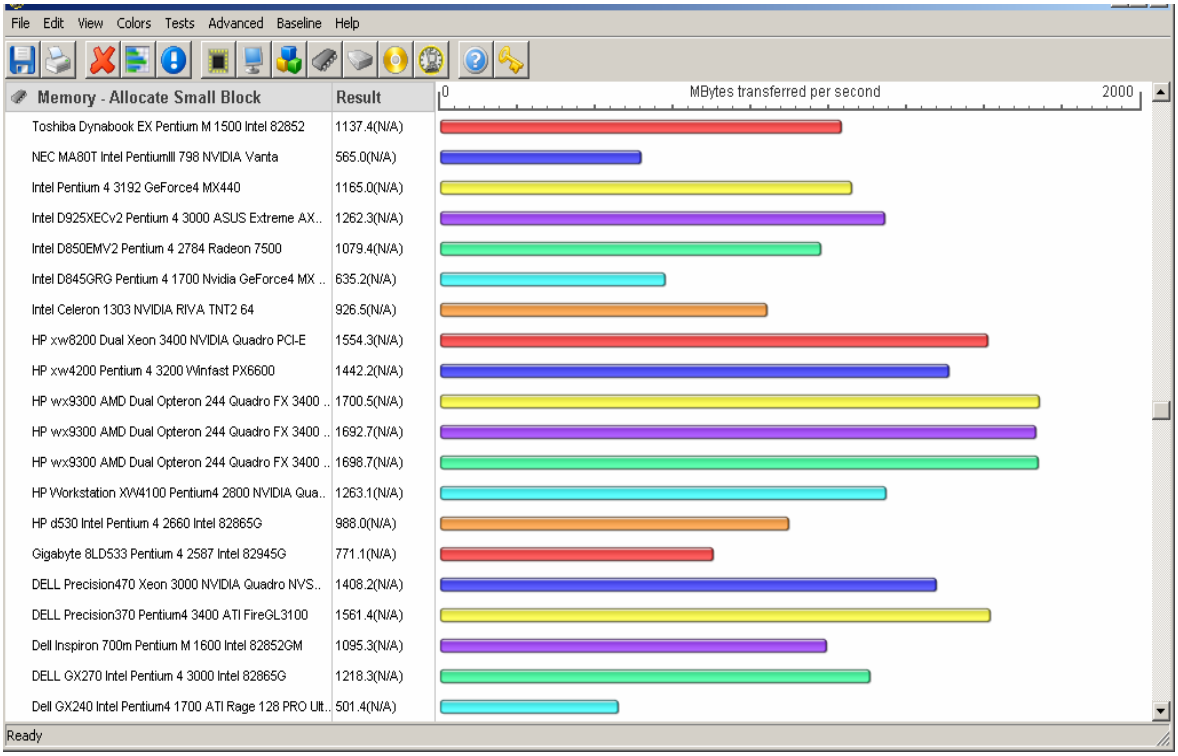


Figure 8.17 CPU PERFOMANCE FOR MEMORY TO ALLOCATE SMALL BLOCKS

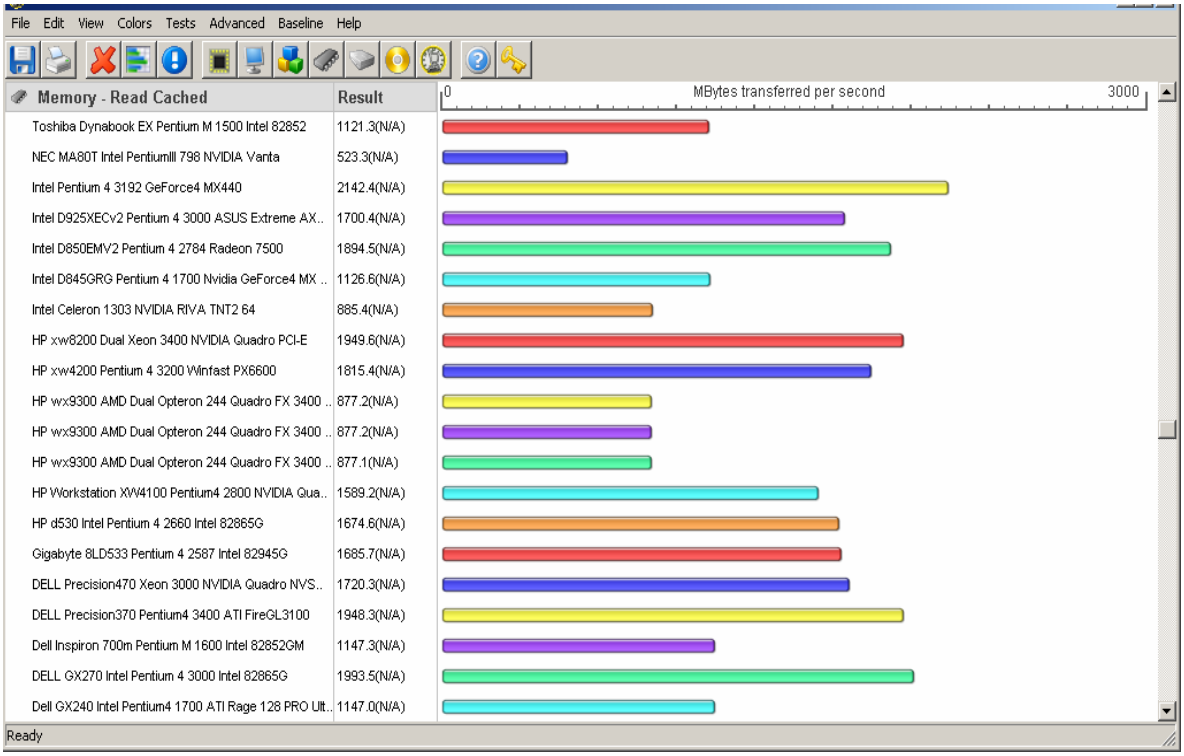


Figure 8.18 CPU PERFOMANCE FOR MEMORY READ CACHED

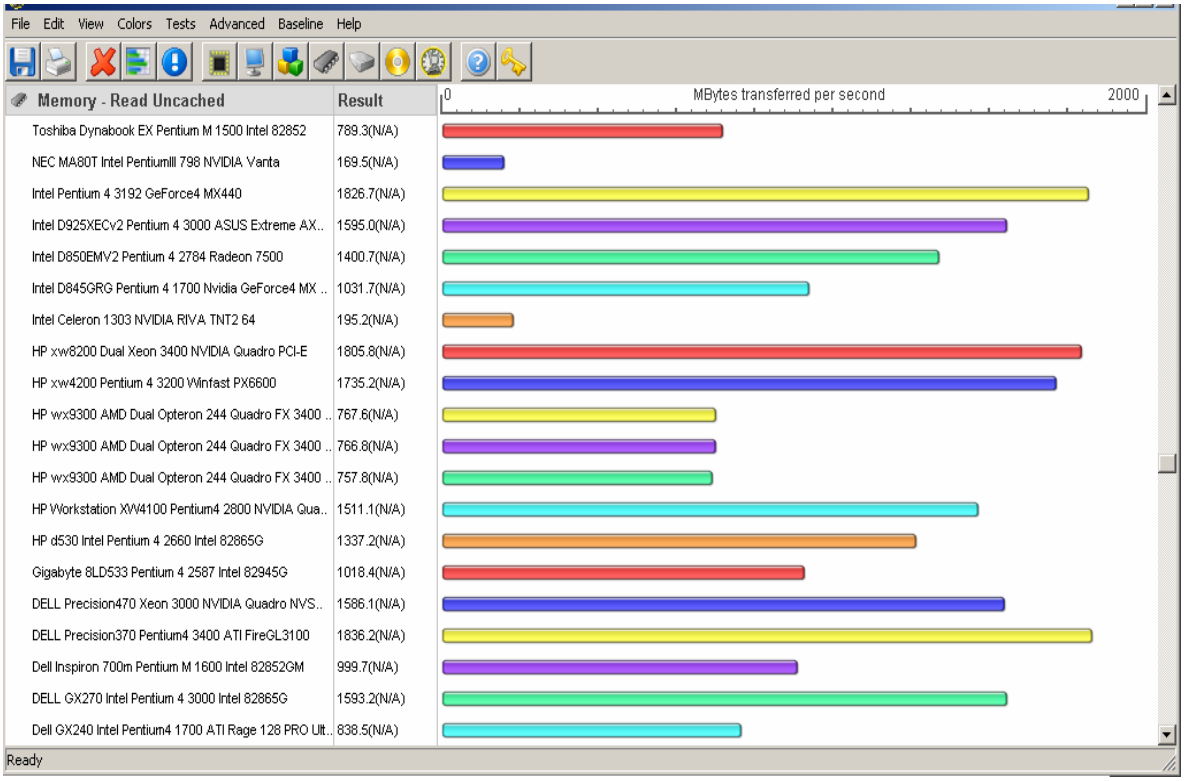


Figure 8.19 CPU PERFOMANCE FOR MEMORY READ UNCACHED

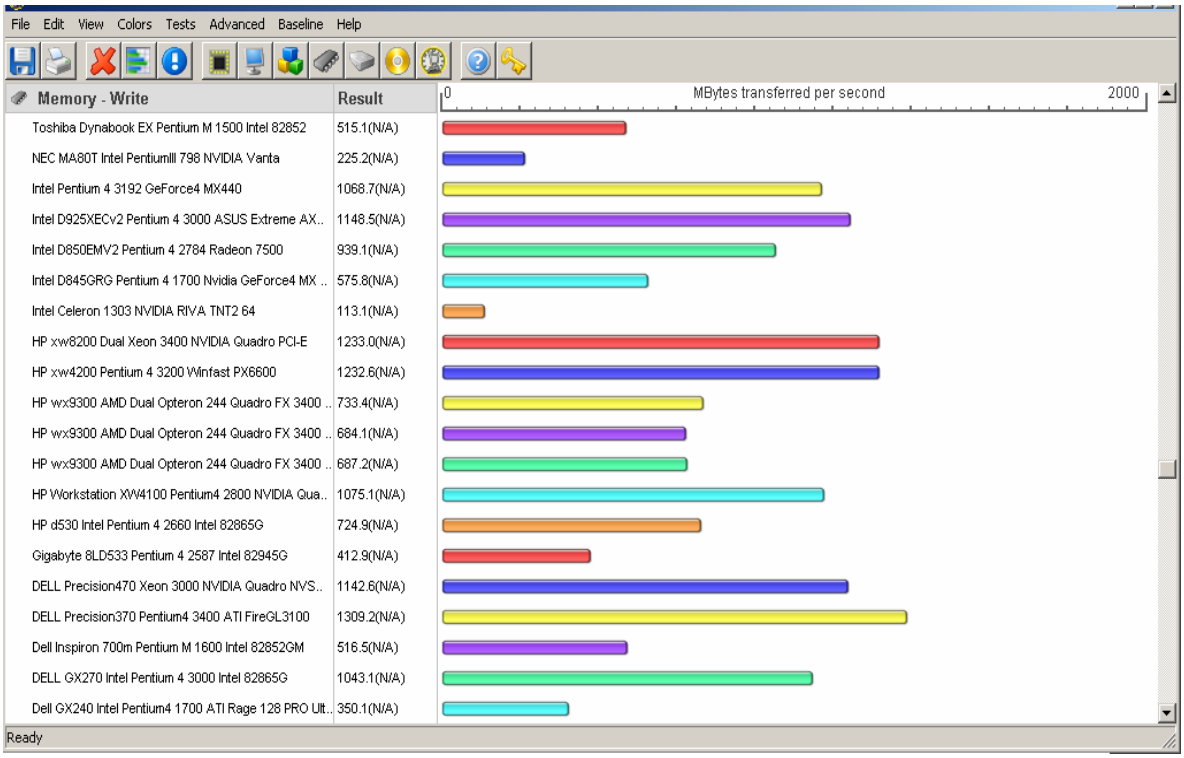


Figure 8.20 CPU PERFOMANCE FOR MEMORY WRITE

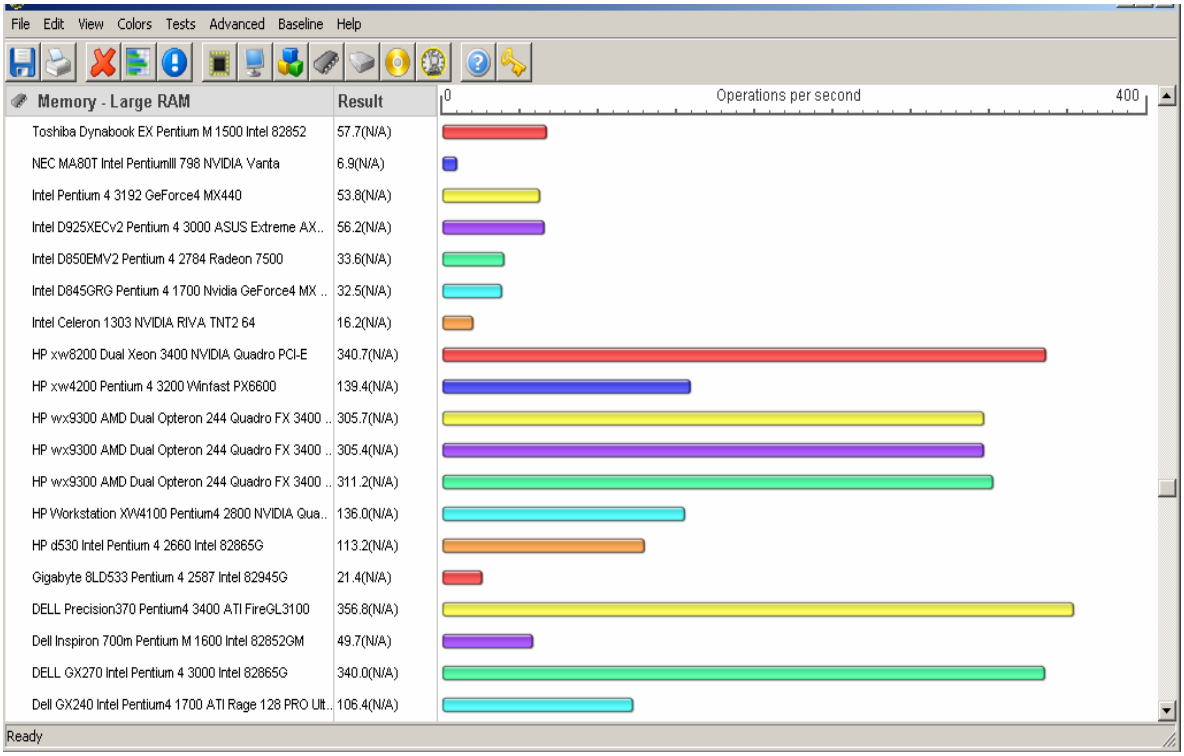


Figure 8.21 CPU PERFOMANCE FOR MEMORY LARGE RAM

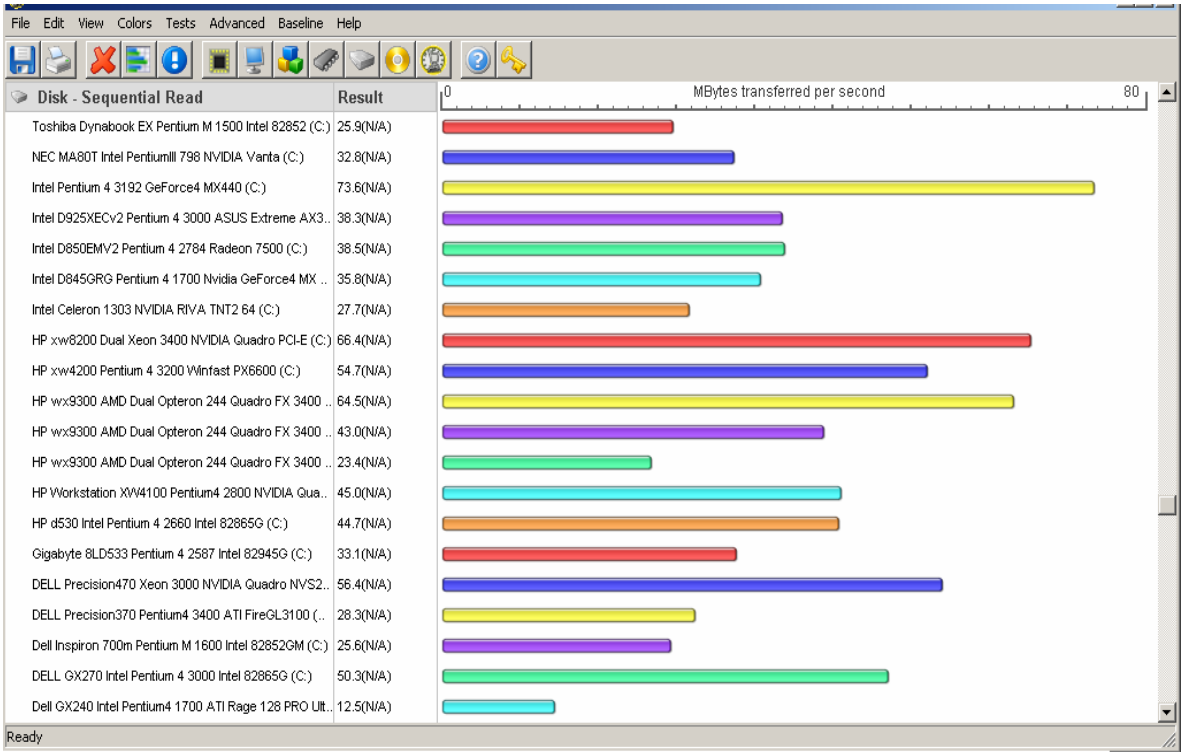


Figure 8.22 CPU PERFOMANCE FOR DISK SEQUENTIAL READ

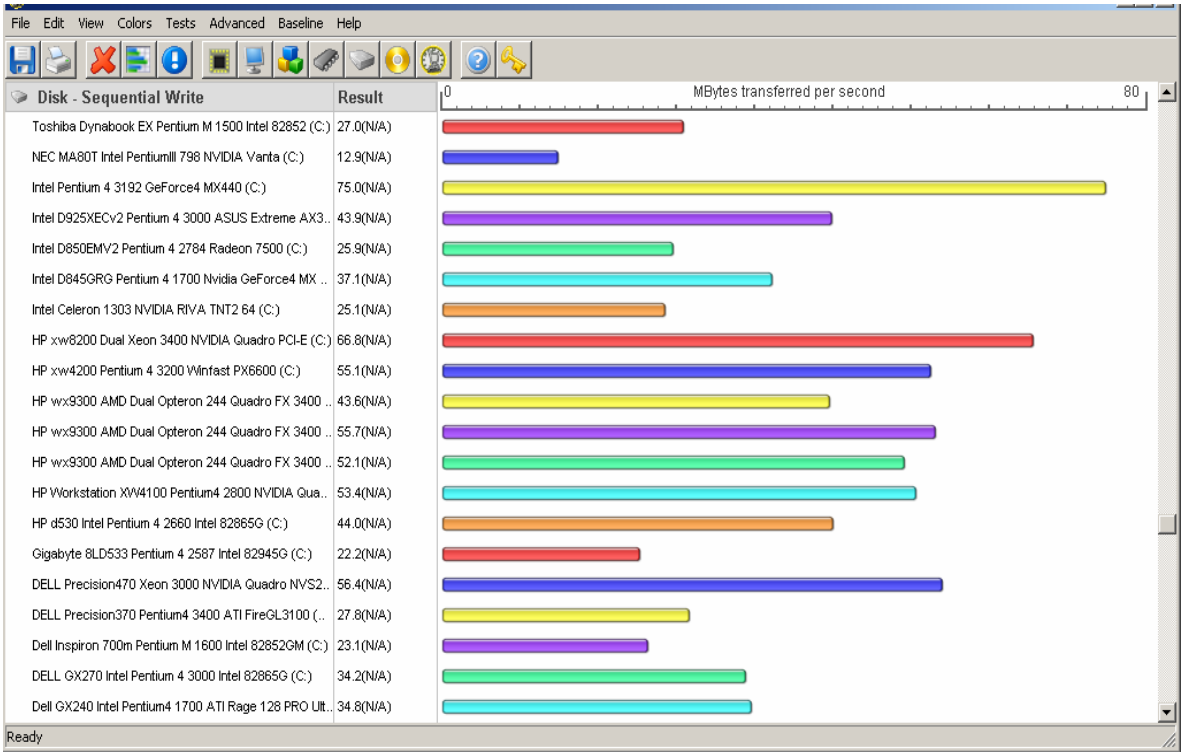


Figure 8.23 CPU PERFOMANCE FOR DISK SEQUENTIAL WRITE

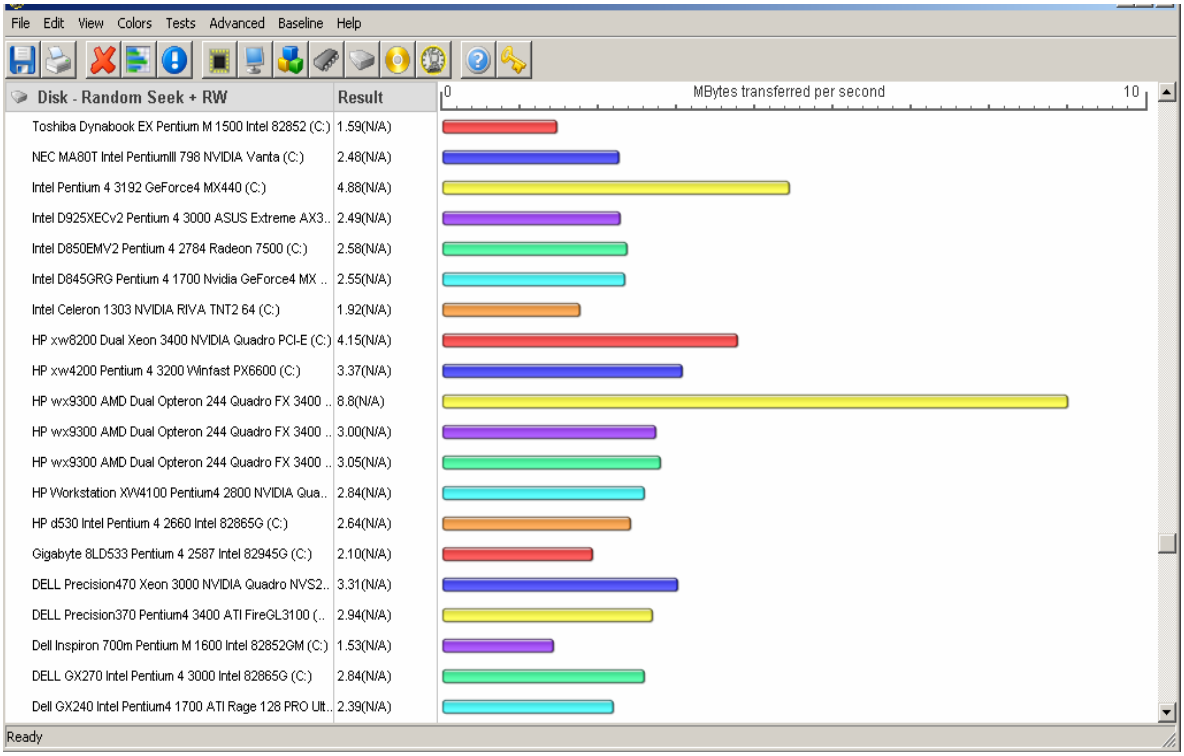


Figure 8.24 CPU PERFOMANCE FOR DISK RANDOM SEEK + READ/WRITE

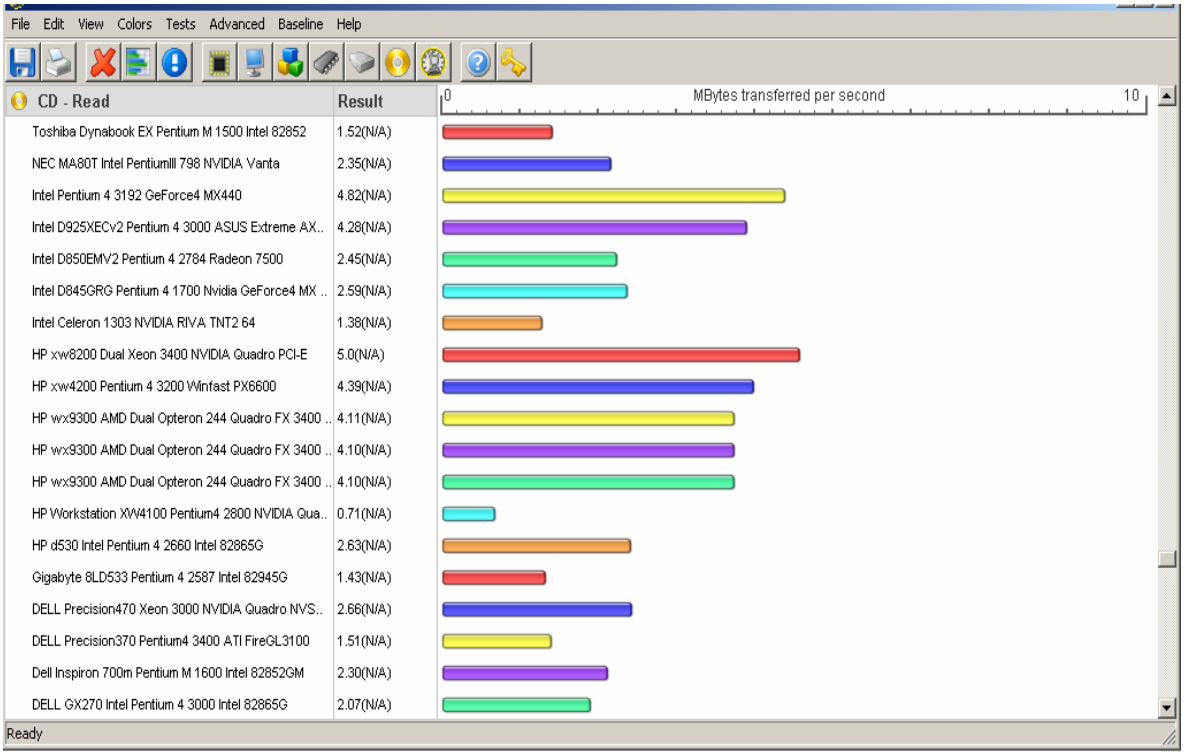


Figure 8.25 CPU PERFOMANCE FOR CD READ

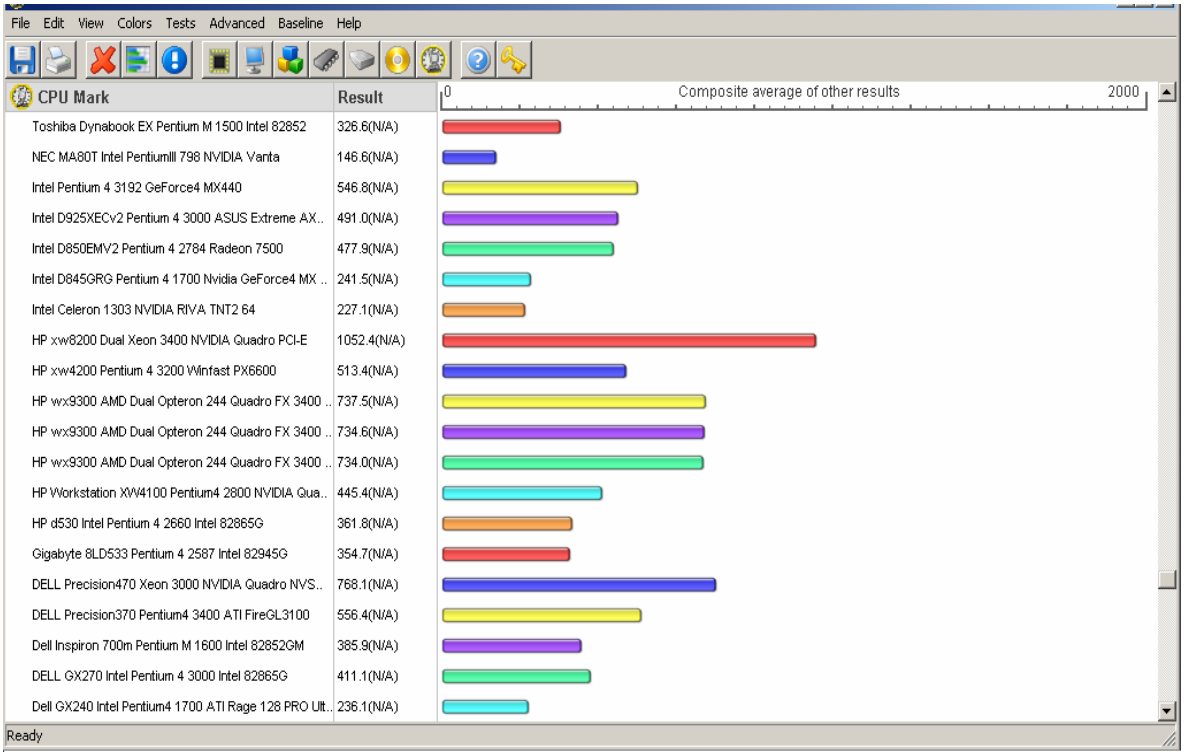


Figure 8.26 CPU PERFOMANCE FOR CPU MARK

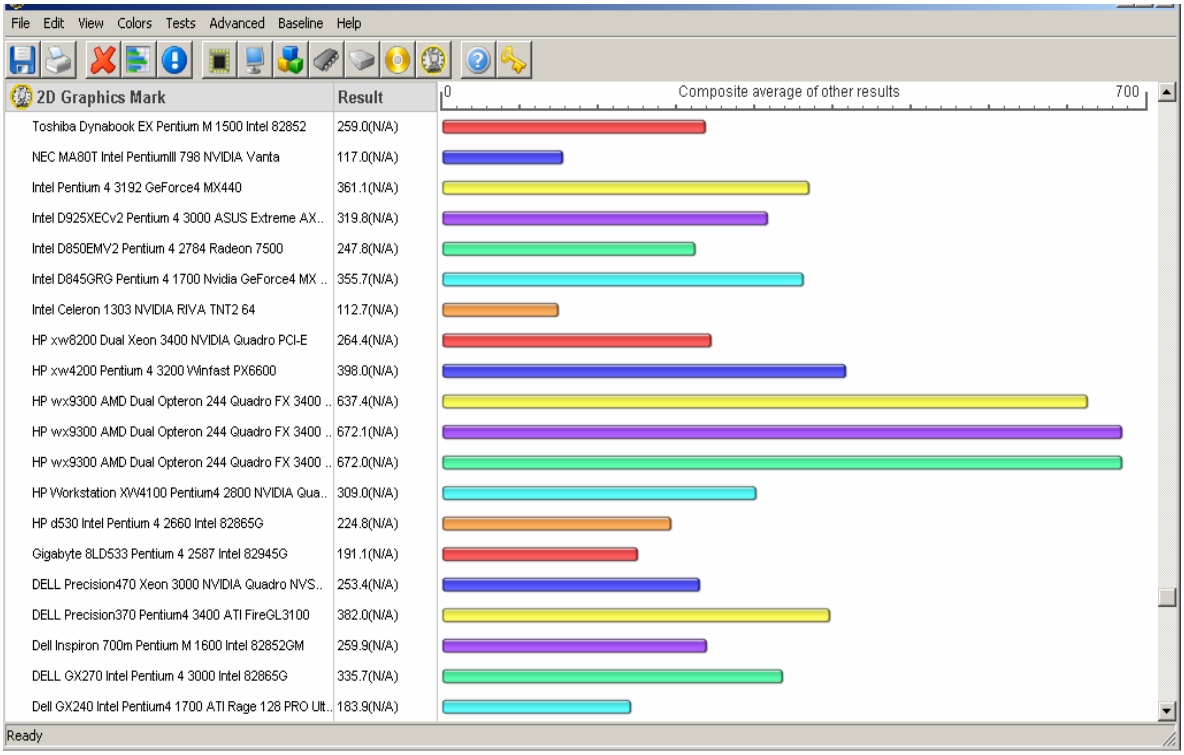


Figure 8.27 CPU PERFOMANCE FOR 2D GRAPHICS MARK

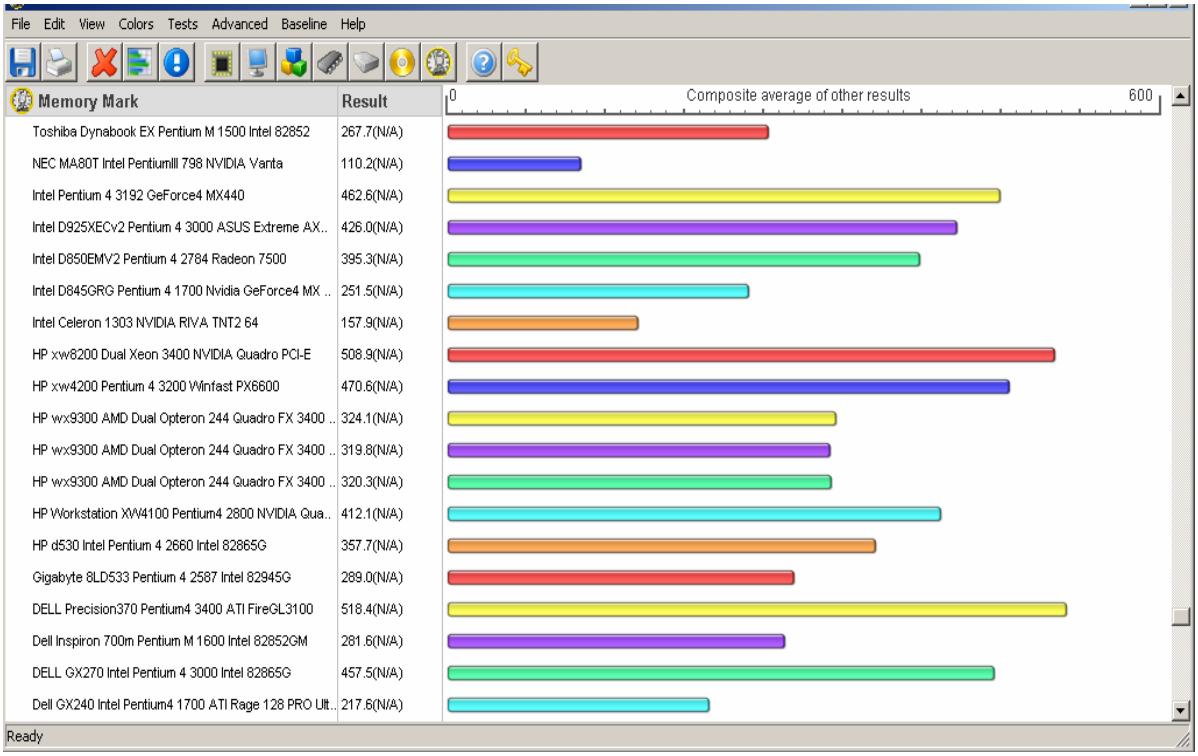


Figure 8.28 CPU PERFOMANCE FOR MEMORY MARK

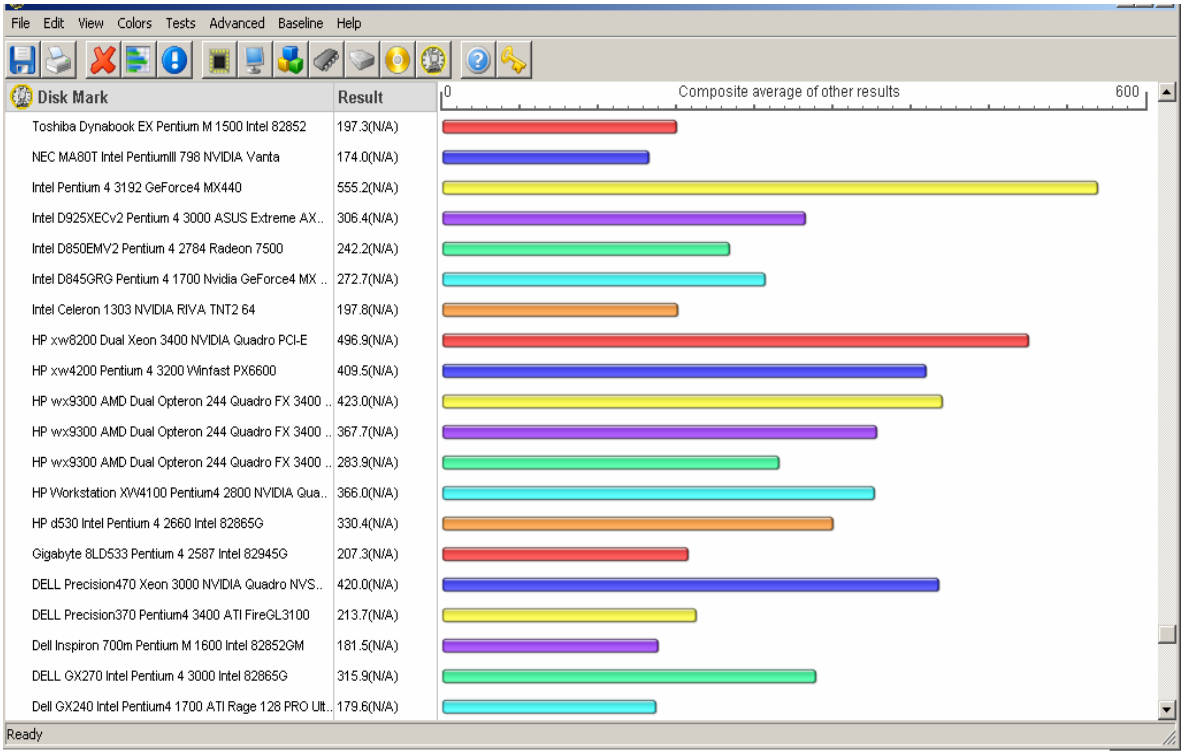


Figure 8.29 CPU PERFOMANCE FOR DISK MARK

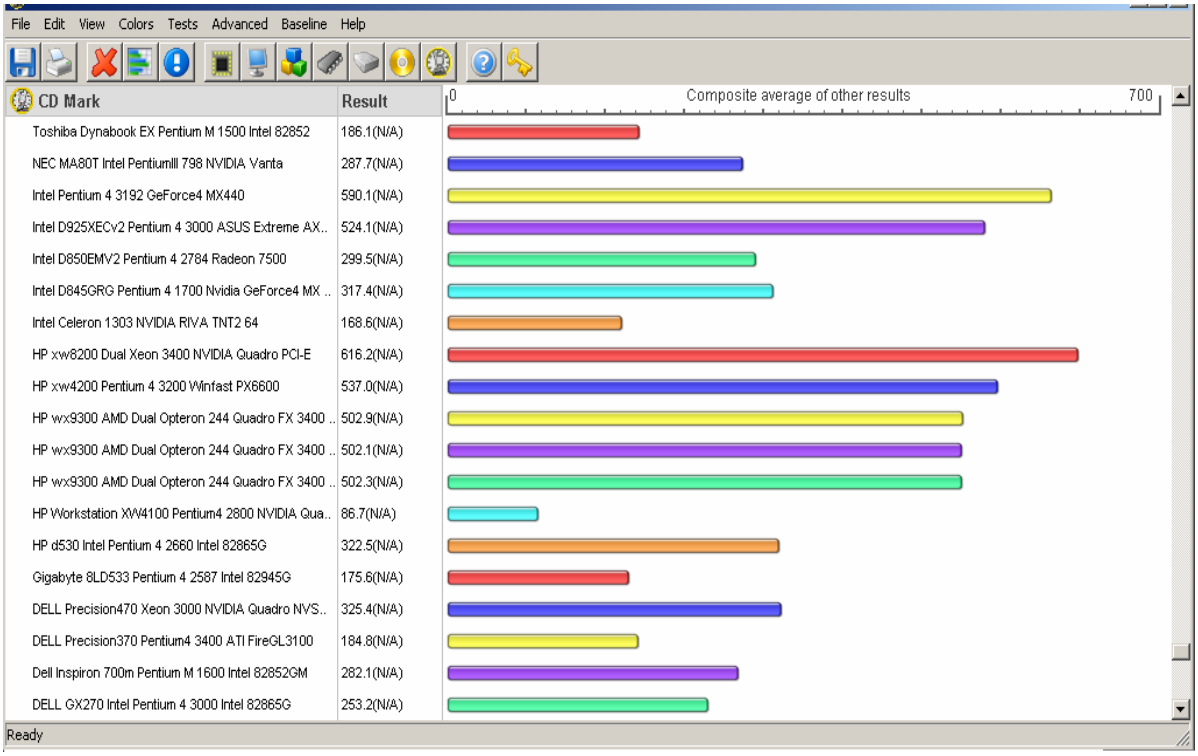


Figure 8.30 CPU PERFOMANCE FOR CD MARK

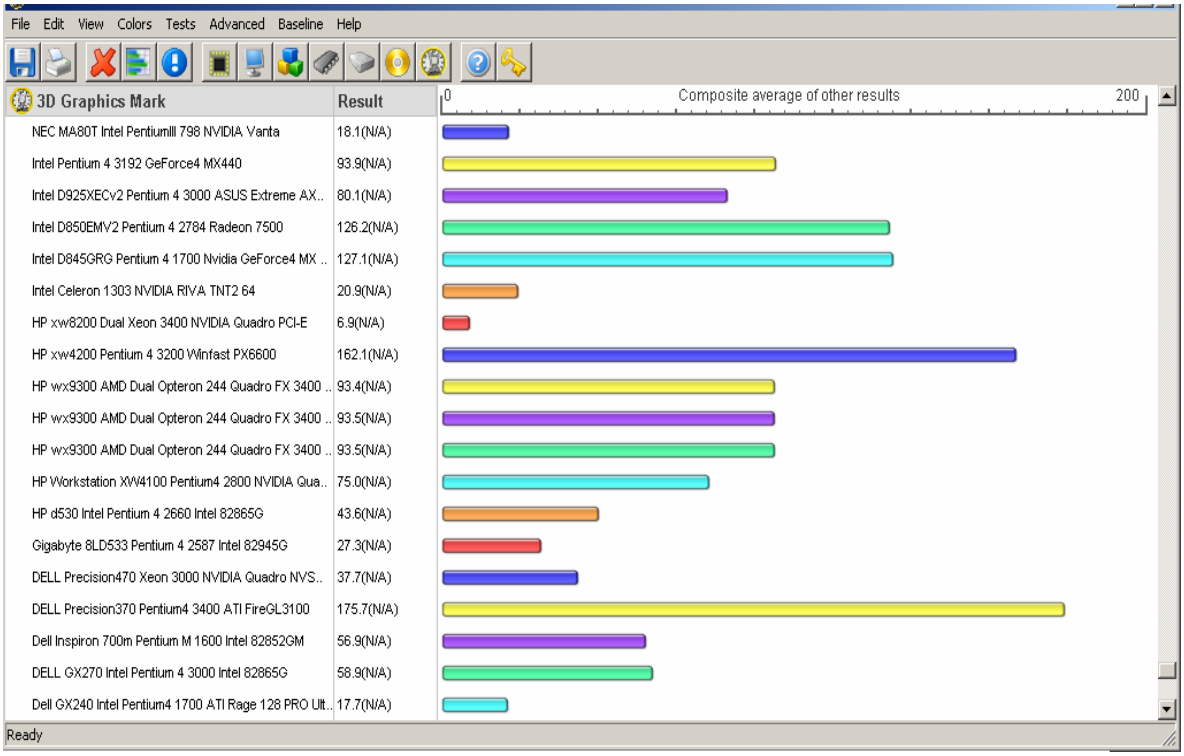


Figure 8.31 CPU PERFOMANCE FOR 3D GRAPHICS MARK

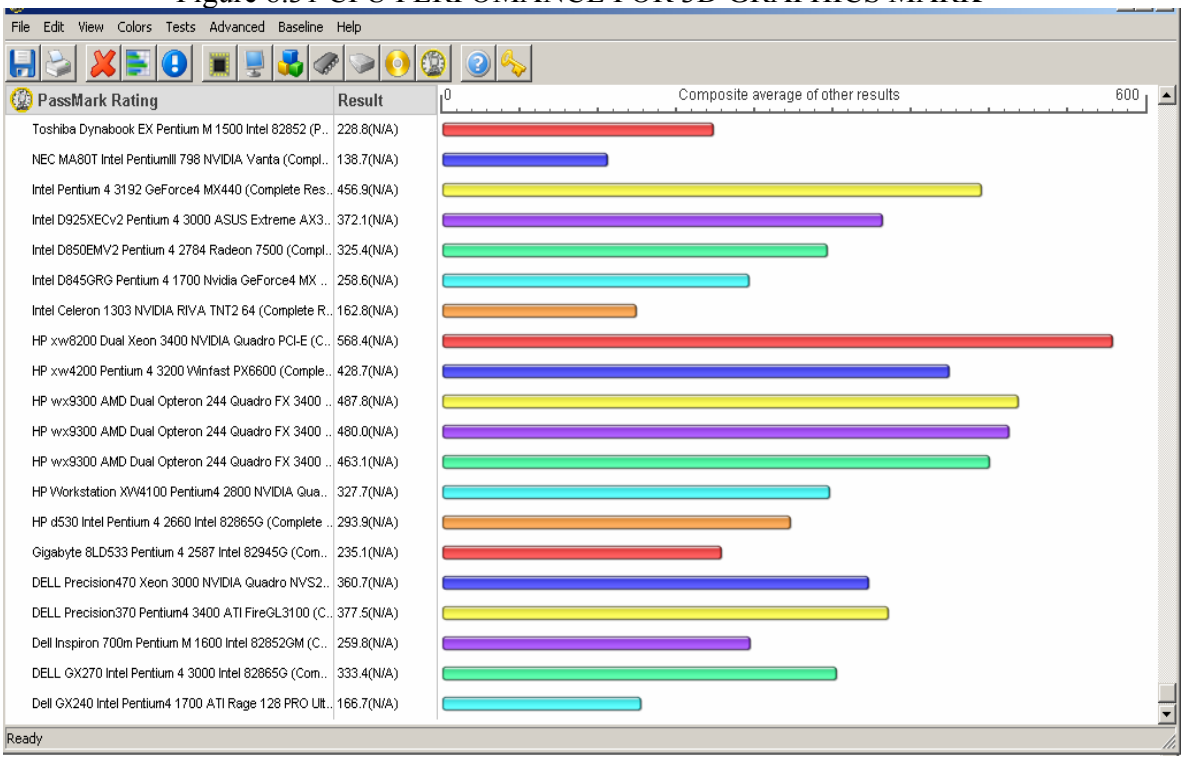


Figure 8.32 CPU PERFOMANCE FOR PASSMARK RATING

CHAPTER 9

CONCLUSION

When I started my research on motherboards, I started just overall literature survey; I found that motherboard is very big issue for designers, manufacturers, professionals and researchers of motherboards. It affects a lot on the global business of computer world because motherboard is certainly a complicated and important component in the computer system.

Then I started study right from the scratch as per very wise guidance from Dr. H. N. Pandya. We developed 8088 motherboard, wrote BIOS, studied various processors and chipsets, and finally did performance tests on recently available few motherboards.

This study helps to develop specific motherboards for specific users. Dell has launched new system, XPS, which is specific for gaming purpose. This way manufacturer can design specific systems for kids, young students, professionals, home users and researchers.

This research would also be helpful for students, as I have covered all general aspects of the motherboard. Especially troubleshooting part would help who are interested to study further deep for motherboard issue, study of various parts of motherboard, to develop expansion card of motherboard or to write technical manual for specific motherboard. Students can also start study on specific signals from particular part of the motherboard.

This study will also be helpful to end-users because the choice should not be made lightly, but it should also not be too difficult. It is more important to choose the proper CPU, memory and I/O devices (video cards, hard drives, etc.) and then find the motherboard that supports them. It is very important to know every nuance of every chipset. It seems very hard for the general users to get in-depth knowledge of the chipset, but it is a good idea to have general information about the major features. It *is* very important to know particular needs and requirements are, in terms of hardware support, and what applications will be running on particular system. It is also important to purchase a motherboard based upon more than just price, since a poorly functioning or defective motherboard will render the entire system next to useless.

One should not let anyone else's recommendations determine what motherboard is best for them unless they know that his or her requirements match. Computers are very personal items, and every person uses his or hers slightly differently than everyone else, and has different performance requirements as well (remember, performance isn't just about speed). A search of the various hardware newsgroups will show that there are very happy users of motherboards the hardware sites (and many "expert" users) have rated extremely poorly - because the board satisfied *their* needs.