High Performance Computer Architecture
Prof. Ajit Pal
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Lecture - 34
Case Studies

Hello and welcome to today's lecture on Case Studies, so far we have discussed different types of architectural features, various innovations, which can be used to realize processors. So, today I shall discuss about some case studies particularly, we shall focus on evolution of Intel processors.

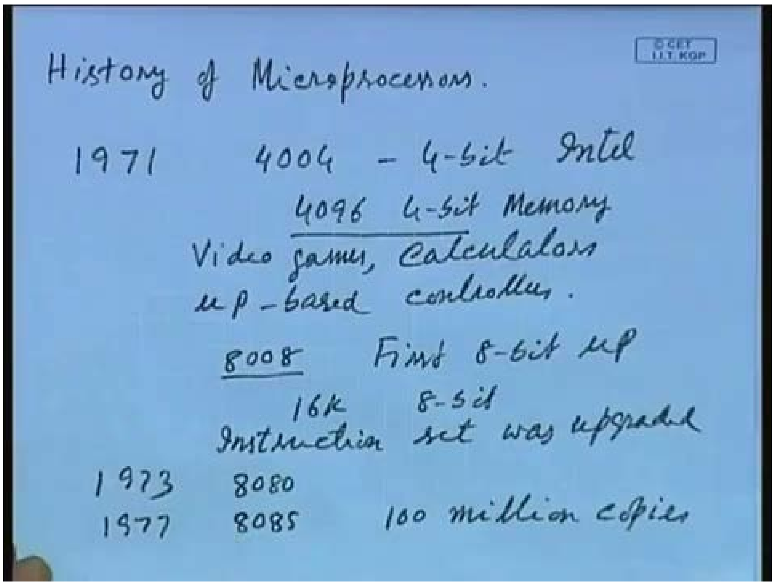
(Refer Slide Time: 01:18)

Evolution of Intel Processors

- In the subsequent slides:
  - We shall examine how Intel processors have evolved over the years.
  - The objective is to examine how various architectural innovations we discussed so far have been incorporated in the Intel processors.

So, we shall see how Intel processors have evolved over the years, and essentially the objective is to examine how various architectural innovations we have discussed, so far have been incorporated in the Intel processors.

(Refer Slide Time: 01:41)



© CET I.I.T. KGP

History of Microprocessors.

1971 4004 – 4-bit Intel

4096 4-bit Memory

Video games, Calculators

μP – based controllers.

8008 First 8-bit μP

16k 8-bit

Instruction set was upgraded

1973 8080

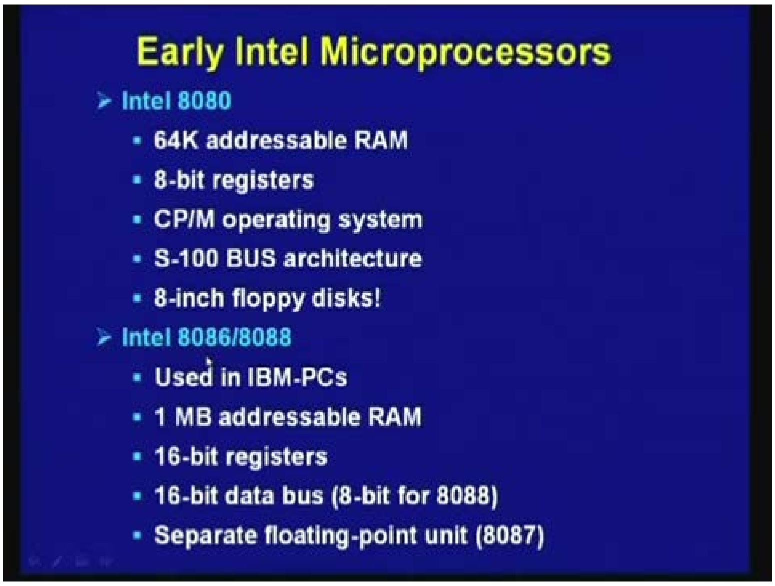
1977 8085 100 million copies

So, if we look at the history of the micro processors, we find that microprocessor was invented sometime in 1971, and the first processor was 4004, that was a 4-bit processor developed by Intel. And it was having only 4096 4-bit memory, and it has very limited processing capability and it was primarily used for video games, calculators, then microprocessor based controllers. And the instruction set was very primitive and subsequently another processor that was 8-bit was introduced, that is your 8008.

And this is the first 8-bit processor microprocessor you can say, and memory was extended from 4 k 4-bit to 16 k 8-bit and instruction set was also upgraded. Then back in 1973, another version of 8008 known as 8080, which was much better processor having many modern features that was introduced and subsequently in 1977, the one of the most powerful microprocessor that was 8085 that was proposed. So, this 8085 itself is a 8-bit processor and it has many features, like I mean instruction set was quite rich.

And 100 million copies were sold by Intel itself and not only Intel was manufacturing it, but many other companies like, Toshiba, AMD, NEC, Hitachi they got license and they manufacture the same chip.

(Refer Slide Time: 04:39)



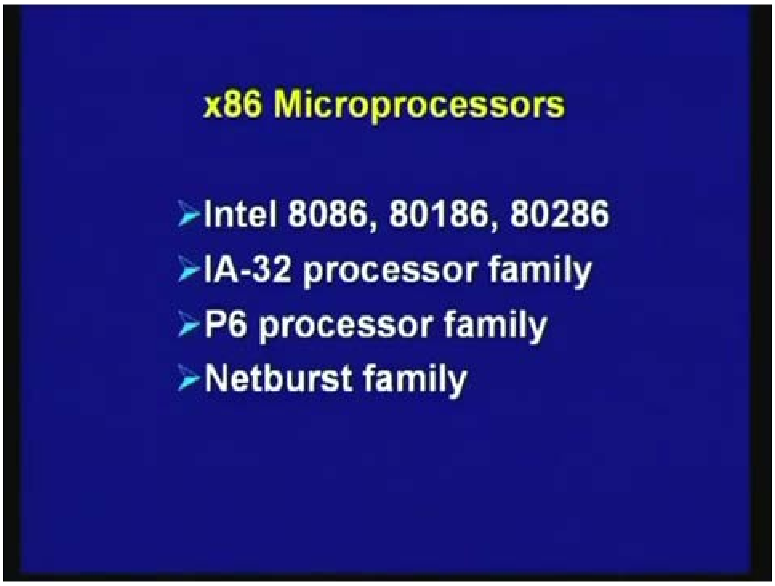
Early Intel Microprocessors

- Intel 8080
  - 64K addressable RAM
  - 8-bit registers
  - CP/M operating system
  - S-100 BUS architecture
  - 8-inch floppy disks!
- Intel 8086/8088
  - Used in IBM-PCs
  - 1 MB addressable RAM
  - 16-bit registers
  - 16-bit data bus (8-bit for 8088)
  - Separate floating-point unit (8087)

So, based on the success, the history progressed and then, subsequently from 8-bit to 16 processors were evolved, particularly 8086 and 8088, that was the 16-bit processor. And it has many modern features and has powerful processing capability of course, much lower processing capability in present context. But, because of larger memory space, 1 Megabyte addressable RAM and all the registers were 16-bit registers, 16-bit data bus.

And it was possible to have a separate floating point unit processing unit 8087, which is known as co-processor. And because of all these features the full fledged computer known as IBM, PC was developed, based on 8088 which was essentially a simplified version of 8086. That means, the bus was 8, but internal capabilities were same as 8086, and based on this success subsequent processors were developed.

(Refer Slide Time: 05:48)

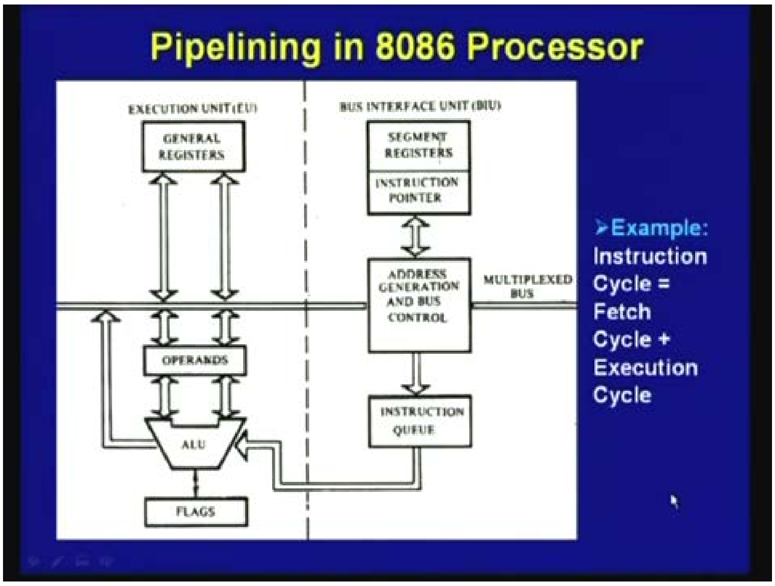


x86 Microprocessors

- Intel 8086, 80186, 80286
- IA-32 processor family
- P6 processor family
- Netburst family

As we can see, we got X 86 series of micro processors for example, Intel 8086 that was evolved into 80186, 80286, then these were all 16-bit processors subsequently 32-bit processors, IA-32 processor family were introduced by Intel. Then P6, Pentium 6 processor family were introduced, and subsequently net burst family of processors were introduced, which we shall discuss briefly in today and in the next lecture.

(Refer Slide Time: 06:29)



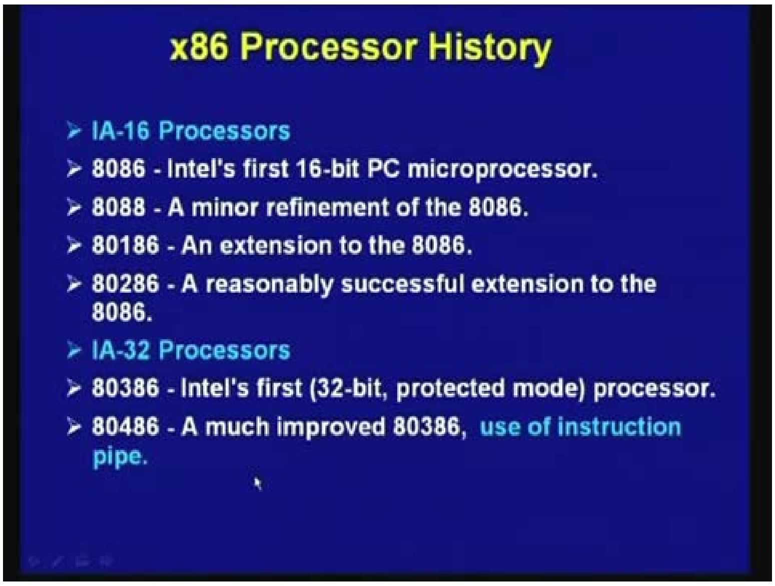
Pipelining in 8086 Processor

- Example: Instruction Cycle = Fetch Cycle + Execution Cycle

So, this is the internal architecture 8086 processor, as you can see it has got two distinct units, one is known as execution unit another is bus interface unit. And these two units are essentially work in a over left manner, these are they used in pipeline manner, so pipelining was introduced with this processor. And this bus interface unit was accessing memory and fetching instructions, and so address generation and bus control all these things were generated with the help of this bus interface unit. And of course, there are certain internal registers like segment registers, instruction pointer which are used for generating the effective address.

And after fetching the instructions, the instructions were stored in a queue known as instruction queue and from this instruction queue, instructions were taken by the execution unit, for processing the instructions. And the instruction unit comprises the ALU registers, various registers like general purpose registers and other registers where you can store operand flag bits and so on. So, this is how the pipelining was introduced in Intel processor, and 8086 was a beginning of that.

(Refer Slide Time: 08:05)

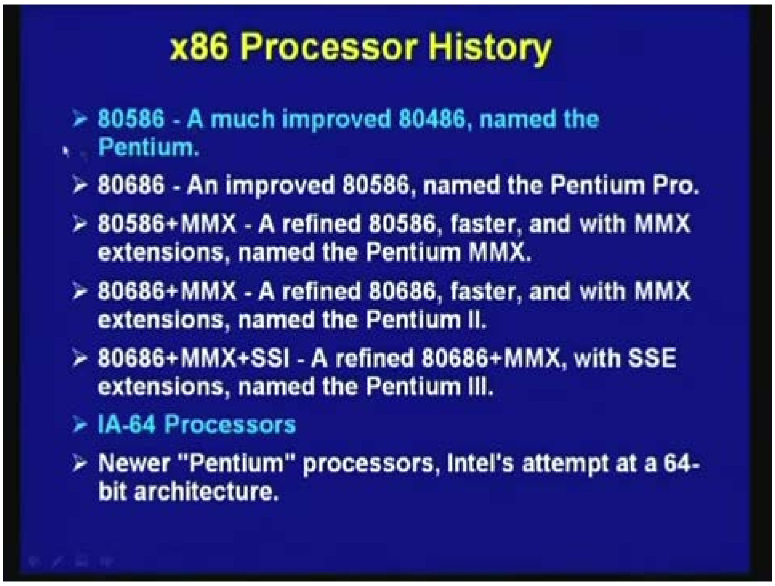


x86 Processor History

- IA-16 Processors
- 8086 - Intel's first 16-bit PC microprocessor.
- 8088 - A minor refinement of the 8086.
- 80186 - An extension to the 8086.
- 80286 - A reasonably successful extension to the 8086.
- IA-32 Processors
- 80386 - Intel's first (32-bit, protected mode) processor.
- 80486 - A much improved 80386, use of instruction pipe.

And as I mentioned 8088 a minor refinement of a 8086 was introduced and that became very popular in their IBM, PC computer. And then, 80186 is an extension of 8086 and 80286 which is little bit powerful than 8086, a reasonably successful extension of 8086. Then, that IA-32 that 32-bit processors 80386, Intel first 32-bit processor that was introduced and 80486 a much improved version of 80386 was proposed and that which used instruction pipe, we shall discuss about their internal architecture in more detail.

(Refer Slide Time: 08:58)



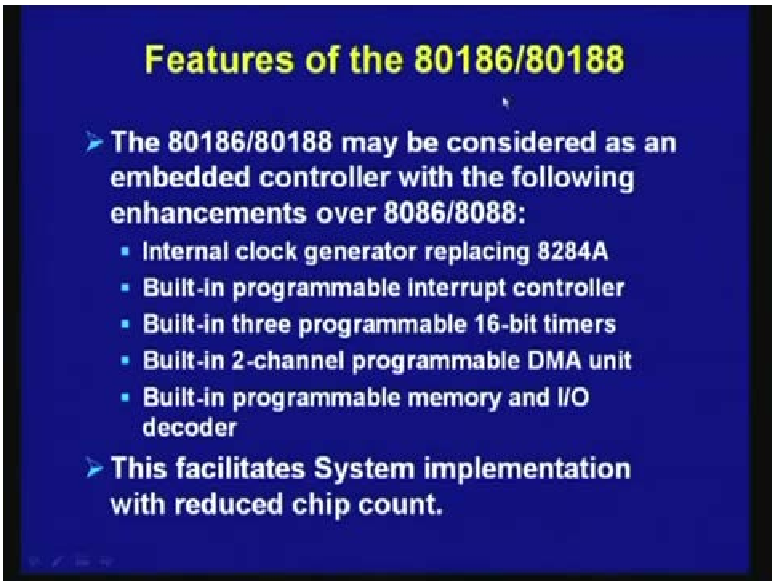
x86 Processor History

- 80586 - A much improved 80486, named the Pentium.
- 80686 - An improved 80586, named the Pentium Pro.
- 80586+MMX - A refined 80586, faster, and with MMX extensions, named the Pentium MMX.
- 80686+MMX - A refined 80686, faster, and with MMX extensions, named the Pentium II.
- 80686+MMX+SSI - A refined 80686+MMX, with SSE extensions, named the Pentium III.
- IA-64 Processors
- Newer "Pentium" processors, Intel's attempt at a 64-bit architecture.

Then 80586 a much improved 8048 version of 80486 was introduced, that was named as Pentium and subsequently some up gradations of 80586 were proposed 80686, and improved of 80586, named as Pentium Pro. Then 80586 MMX refined 80586 faster and with MMX extensions, that is for multimedia processing, named as Pentium MMX. Then 80686 MMX, a refinement of 80686 a faster with MMX extensions, multimedia extensions named as Pentium II and 80686 plus MMX plus SSI.

SSI essentially provides you more powerful processing capability and leading to Pentium III, then IA-64 processors, the newer Pentium processors that is Intel attempt of 64-bit architecture, which I shall discuss briefly one after the other.

(Refer Slide Time: 10:17)



Features of the 80186/80188

- The 80186/80188 may be considered as an embedded controller with the following enhancements over 8086/8088:
  - Internal clock generator replacing 8284A
  - Built-in programmable interrupt controller
  - Built-in three programmable 16-bit timers
  - Built-in 2-channel programmable DMA unit
  - Built-in programmable memory and I/O decoder
- This facilitates System implementation with reduced chip count.

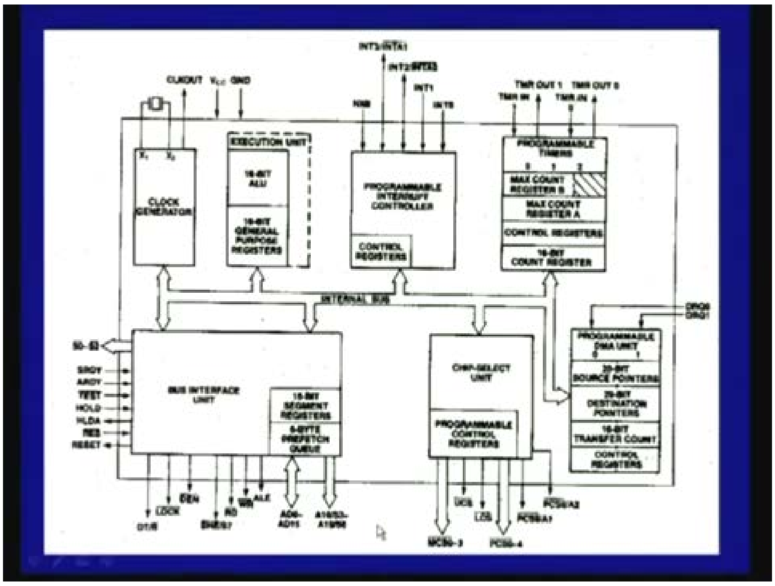
First let us focus on 80186 and 80188, just like 8086 and 8088 this 80186 and 80188 they introduced two processors. And this 80186 and 80188 may be considered as an embedded controller, with the following enhancements over 8066 and 8088. You see Intel processors get targeted, their processors in two directions, one is for embedded applications. Embedded applications means which should require minimum number of chips, minimum number of IC's and with the help of that a complete system can be build. And which will go in various embedded systems, like printer, various communication equipments and so on.

So, what was done the 80186 incorporated that internal clock generator, earlier in 8086 that clock generator were outside the processors. So, 8084 A was the clock generator IC that was externally at a connected to 8086, that was included as part of the chip, and built in programmable interrupt controller. Interrupt controller was also outside the chip in 8086, so that was also put inside the chip and built in three programmable 16-bit timers, that was incorporated as a part of the chip and built in 2 channel programmable DMA controller.

So, DMA controller is necessary outside the chip in case of 8086, but here as long as the DMA you are satisfied with 2 channel DMA interface, then 2 channel programmable DMA unit is built in part of 80186. And built in programmable memory and IO decoder, whenever you built a complete system you will require some decoder IC's, those decoder

IC's will select memory devices, IO devices and so on. So, without using these decoder IC's what was done in 80186 itself, some of these I mean decoder and IC's were incorporated. So, all these features facilitated system implementation with reduced chip count, which is very important requirement for embedded applications.

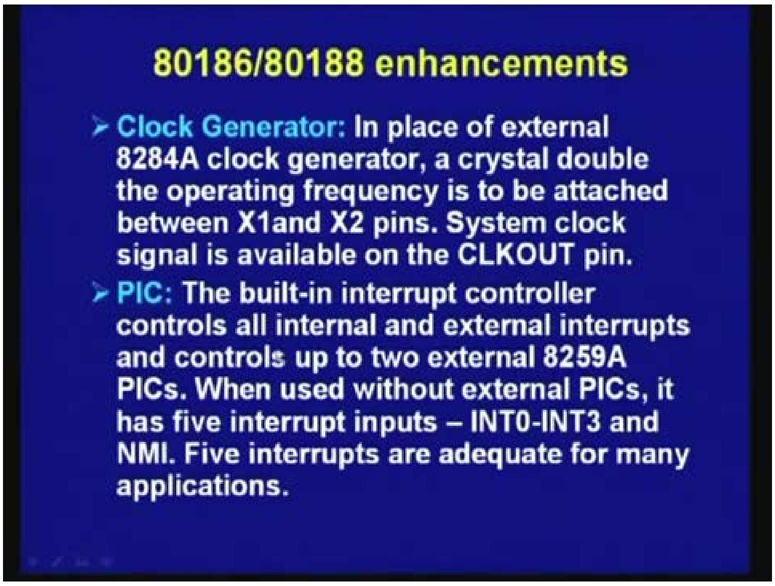
(Refer Slide Time: 13:00)



So, this is the internal diagram as you can see, apart from the processor this is the processor execution unit, it has got clock generator, programmable interrupt controller, programmable timer counter which are needed in many applications to generate time delays. And different types of counters, in different embedded applications, then DMA this is that 2 channel DMA controller, and this is the programmable chips select unit, as I mentioned it you will require a decoders for selecting memory and IO devices.

This performs the function of providing the chips select signals, then the bus interface unit which was present in earlier 8086 that is also present.

(Refer Slide Time: 13:58)

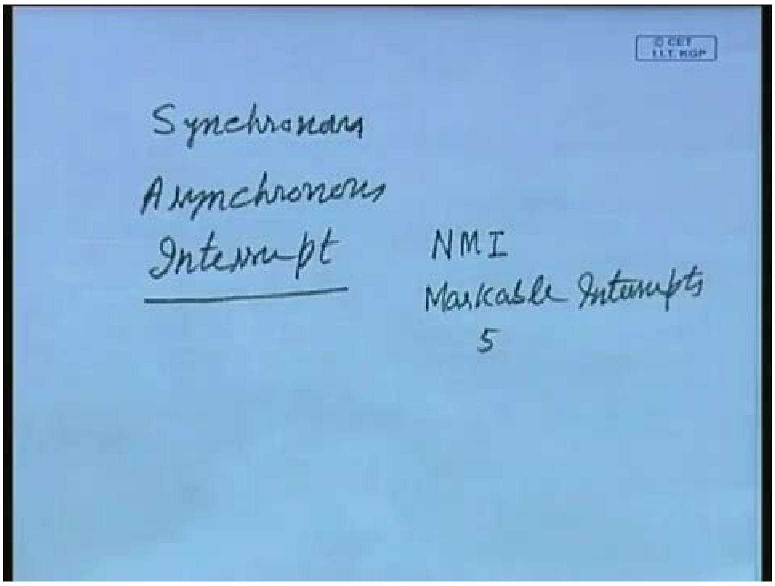


80186/80188 enhancements

- **Clock Generator:** In place of external 8284A clock generator, a crystal double the operating frequency is to be attached between X1and X2 pins. System clock signal is available on the CLKOUT pin.
- **PIC:** The built-in interrupt controller controls all internal and external interrupts and controls up to two external 8259A PICs. When used without external PICs, it has five interrupt inputs – INT0-INT3 and NMI. Five interrupts are adequate for many applications.

So, as I already mentioned clock generator is replaced by a internal clock generator, similarly programmable interrupt controller, the built in programmable interrupt controller, controls all internal and external interrupts, and controls up to two external 8259 A PIC. You see, normally large number of input IO devices are operated in the interrupt given mode, the data transfer can take place using three distinct modes.

(Refer Slide Time: 14:34)



Synchronous

Asynchronous

Interrupt

NMI

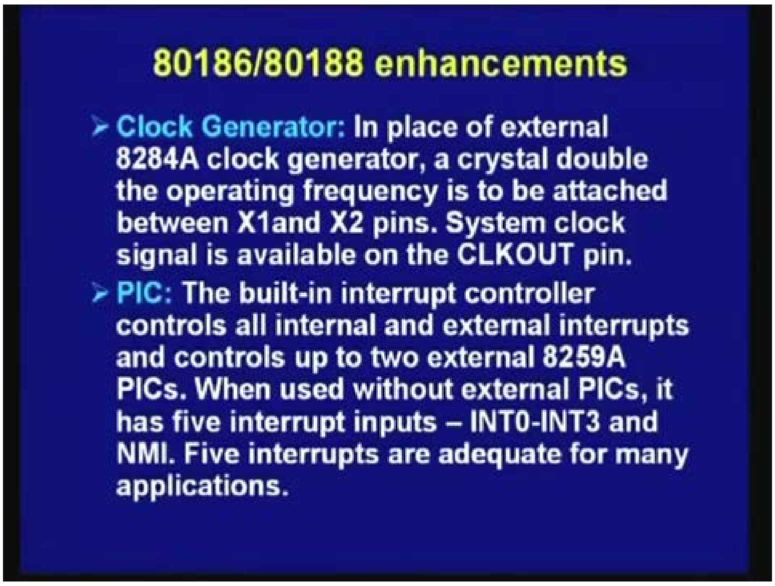
Maskable Interrupts

5

One is known as synchronous mode, another is asynchronous mode and third one is interrupt given mode. Interrupt given mode is very suitable, when the speed of the IO devices is not compatible with that of the processor. And large number of IO devices like say mouse and various types of IO devices that you attach in modern processors, which can be interfaced with the help of this interrupt given transfer technique. And since large number of IO devices are connected, you require external programmable interrupt controller like 8259 A.

So, this programmable interrupt controller is built in can be used to interface to additional external programmable interrupt controller, like 82598 A. And without this external interrupt controller, it provides you five interrupt inputs, INT 0 to INT 3 and NMI, NMI stands for non maskable interrupts. So, interrupt inputs can be broadly divided into two groups, one is known as non maskable interrupts at least one has to be provided, particularly for emergency situations, which cannot be disabled. On the other hand, you may have a number of maskable interrupts, and the number of maskable interrupts provided is 5 in case of 8086,4 INT 0 to INT 3.

(Refer Slide Time: 16:18)

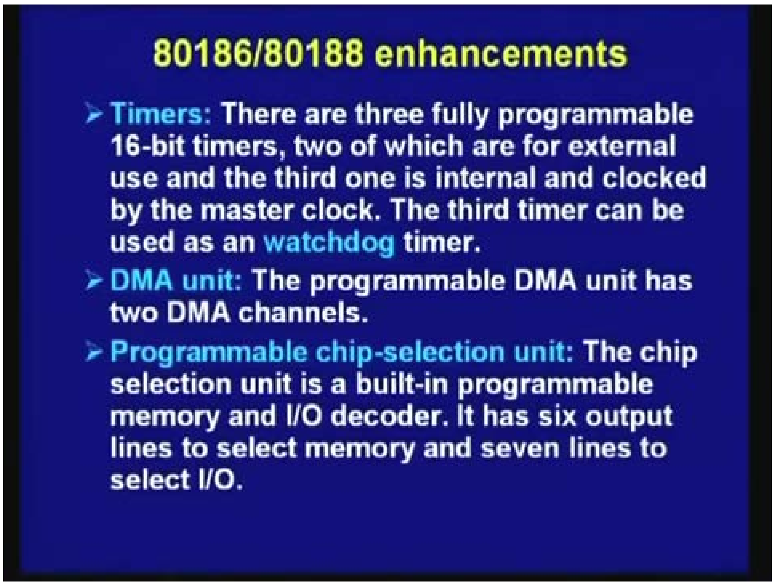


80186/80188 enhancements

- Clock Generator: In place of external 8284A clock generator, a crystal double the operating frequency is to be attached between X1and X2 pins. System clock signal is available on the CLKOUT pin.
- PIC: The built-in interrupt controller controls all internal and external interrupts and controls up to two external 8259A PICs. When used without external PICs, it has five interrupt inputs – INT0-INT3 and NMI. Five interrupts are adequate for many applications.

And five interrupts have been found to be adequate for many applications and however, when it is not sufficient we can use external interrupts.

(Refer Slide Time: 16:34)

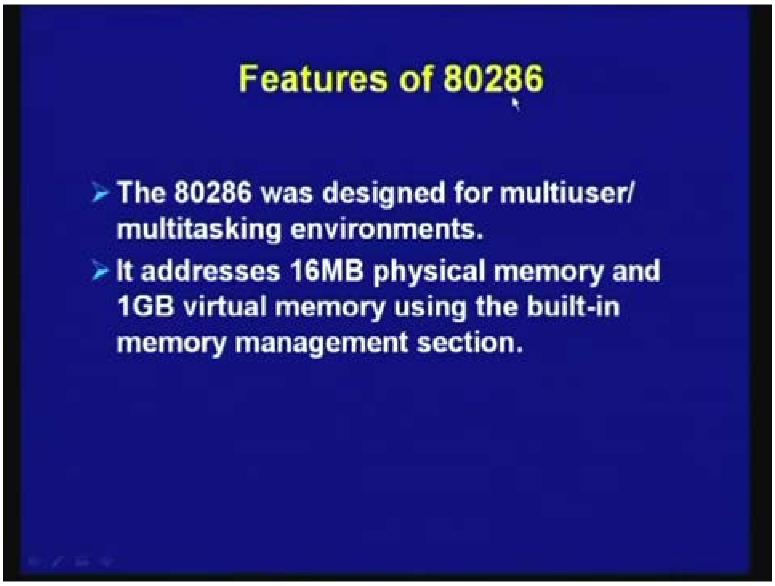


80186/80188 enhancements

- Timers: There are three fully programmable 16-bit timers, two of which are for external use and the third one is internal and clocked by the master clock. The third timer can be used as an watchdog timer.
- DMA unit: The programmable DMA unit has two DMA channels.
- Programmable chip-selection unit: The chip selection unit is a built-in programmable memory and I/O decoder. It has six output lines to select memory and seven lines to select I/O.

Then timer counters as I said, which is required in many applications which can be also be used to implement watchdog timers, which are necessary in embedded systems. If a embed systems fails and if does not response with some duration, then watchdog timer will automatically reset the processors, sometimes the processor is infinite loop or is not able to perform the task for which it is designed. In such a case watchdog timers helps and that is also provided in 80186, then DMA unit as I said it provides 2 channel and then programmable chip select unit, for selecting six output lines to select memory and seven lines to select IO. So, you can interface external memory and IO devices without the need for additional decoder IC's.

(Refer Slide Time: 17:39)

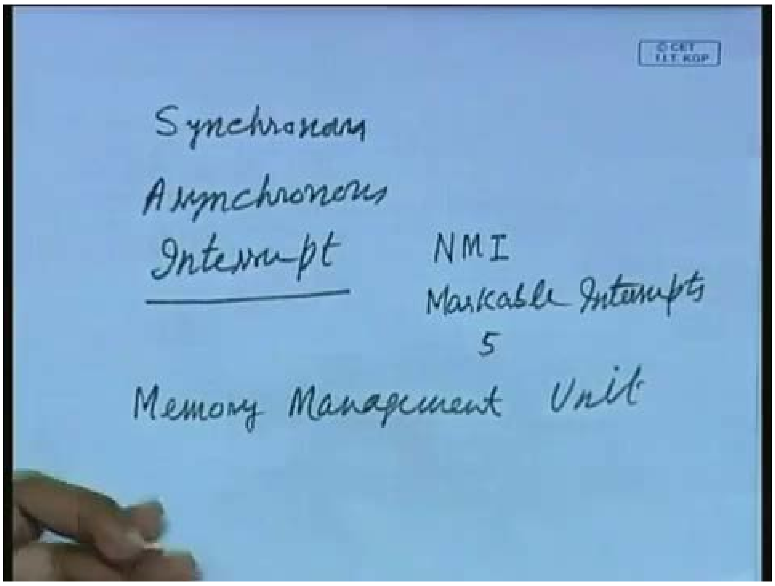


Features of 80286

- The 80286 was designed for multiuser/ multitasking environments.
- It addresses 16MB physical memory and 1GB virtual memory using the built-in memory management section.

Then coming to 80286, you see the 80186 was meant for embedded applications, on the other hand 80286 is an extension of 8086 that was primarily used for making computers. So, the processing capability were enhanced in 80286 and another new feature was incorporated that is memory management unit.

(Refer Slide Time: 18:04)



Synchronous

Asynchronous

Interrupt

NMI

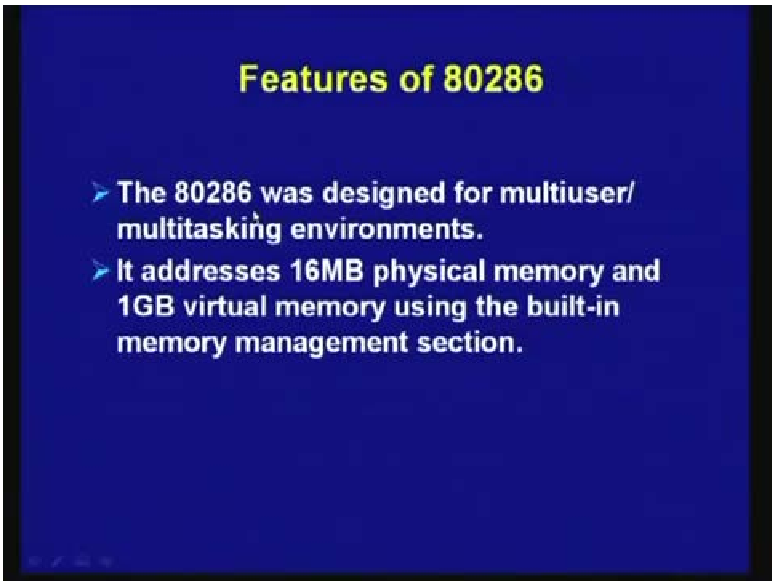
Maskable Interrupts

5

Memory Management Unit

We have already discussed about the need for memory management unit in the context of virtual memory, virtual memory allows you to have larger address space.

(Refer Slide Time: 18:25)

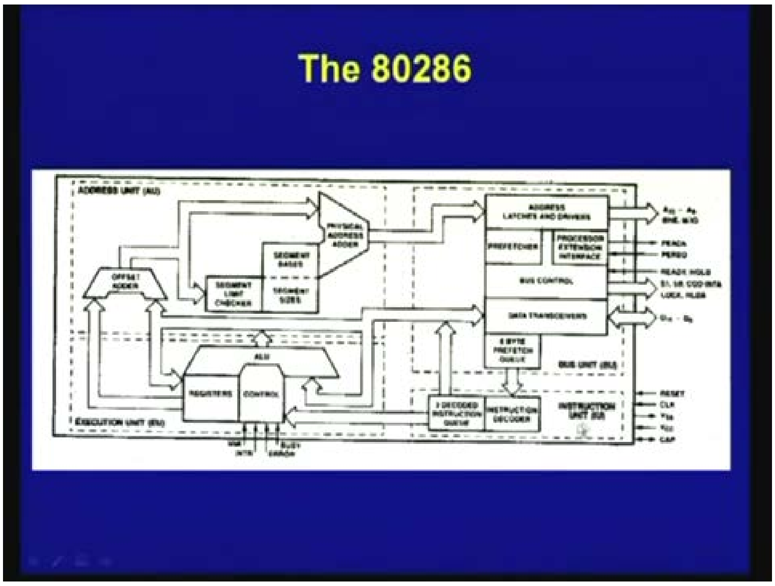


Features of 80286

- The 80286 was designed for multiuser/ multitasking environments.
- It addresses 16MB physical memory and 1GB virtual memory using the built-in memory management section.

And in fact, that is what was done in case of 80286, since the primary purpose was to develop computer the address space was extended from 16, MB physical memory, and also the virtual memory 1 GB virtual memory using the built in memory management section. So, both physical memory and virtual memory were enhanced with the help of this memory management unit.

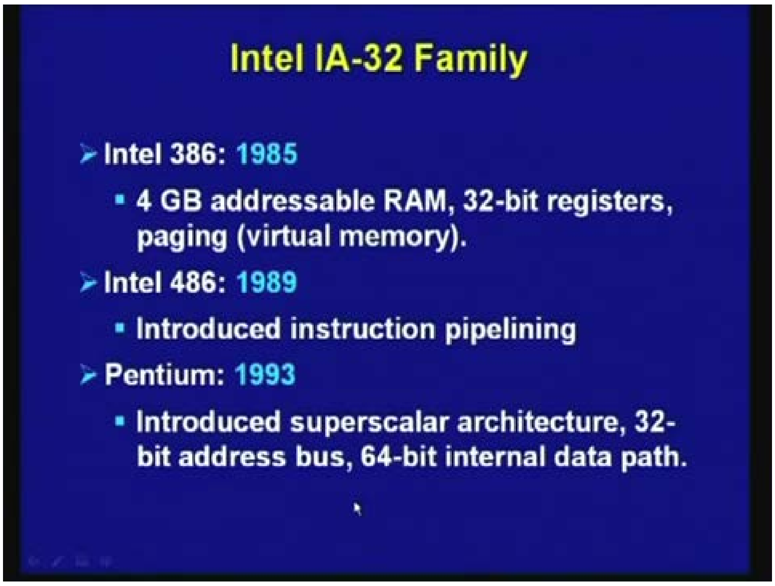
(Refer Slide Time: 18:53)



The 80286

So, this is the internal diagram, as you can see it has got the various registers the processor that for, this is how the address generation is done. And this part is bus interface unit, that which generates the different addresses and other things, and this is that instruction unit where the instruction decoding takes place and so on.

(Refer Slide Time: 19:28)



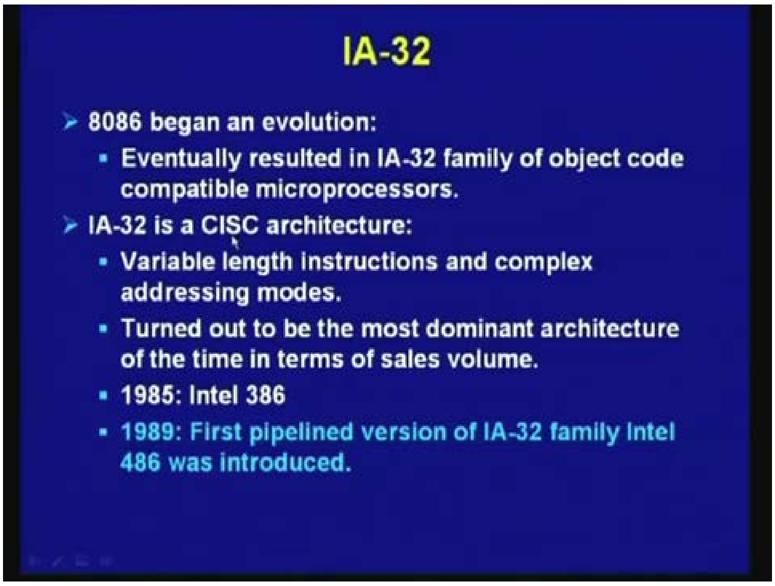
Intel IA-32 Family

- Intel 386: 1985
  - 4 GB addressable RAM, 32-bit registers, paging (virtual memory).
- Intel 486: 1989
  - Introduced instruction pipelining
- Pentium: 1993
  - Introduced superscalar architecture, 32-bit address bus, 64-bit internal data path.

Then coming to 32-bit family back in 1985 Intel 386 was proposed with 4 GB addressable RAM, all the registers were of 32-bit and it allows virtual memory using paging concept. So, the it is very suitable for building computers, then a in 1989, 80486 was introduced with instruction pipelining. So, far there was no pipelining up to 80386 a pipelining was done in a very limited way, there was no separate pipeline for instruction and data.

So, it was a single pipeline, but now you will see that separate pipelining I shall discuss in little more detail about it, then Pentium was introduced in 1993, which was having superscalar architecture. So, this superscalar capability was introduced for the first time in Pentium with 32-bit address bus and 64-bit internal data bus.

(Refer Slide Time: 20:43)

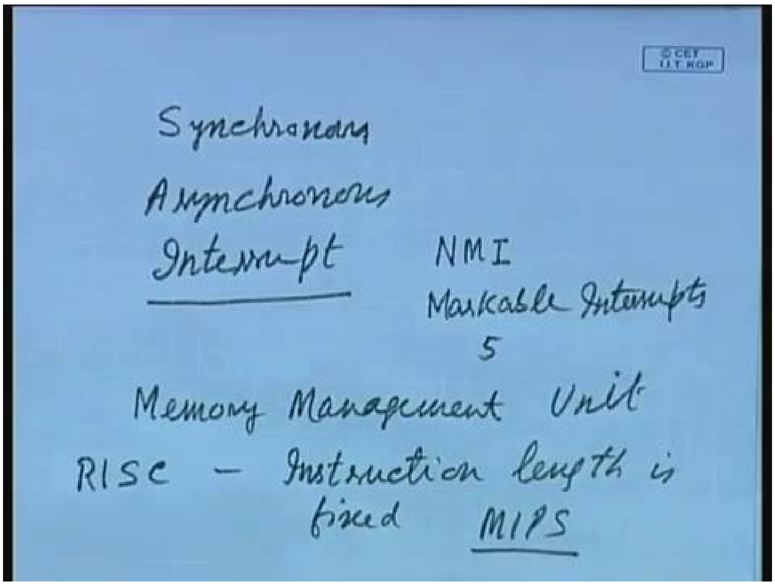


IA-32

- 8086 began an evolution:
  - Eventually resulted in IA-32 family of object code compatible microprocessors.
- IA-32 is a CISC architecture:
  - Variable length instructions and complex addressing modes.
  - Turned out to be the most dominant architecture of the time in terms of sales volume.
  - 1985: Intel 386
  - 1989: First pipelined version of IA-32 family Intel 486 was introduced.

So, as I mentioned 8086 evolved into IA-32 family with 32-bit processors, and basic features of these IA-32 bit processor are that CISC processors. As main difference between CISC processors and RISC processors are, the RISC processors reduce instructions at computer.

(Refer Slide Time: 21:19)



Synchronous

Asynchronous

Interrupt NMI

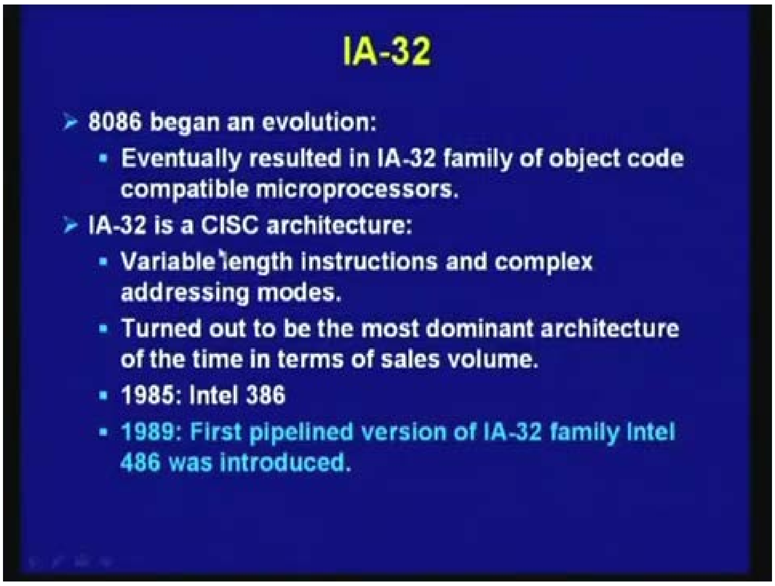
Maskable Interrupts 5

Memory Management Unit

RISC – Instruction length is fixed MIPS

RISC stands for Reduce Instructions Set Computer, and in RISC processor usually the instruction length is fixed for example, in mix instruction level is fixed.

(Refer Slide Time: 21:41)

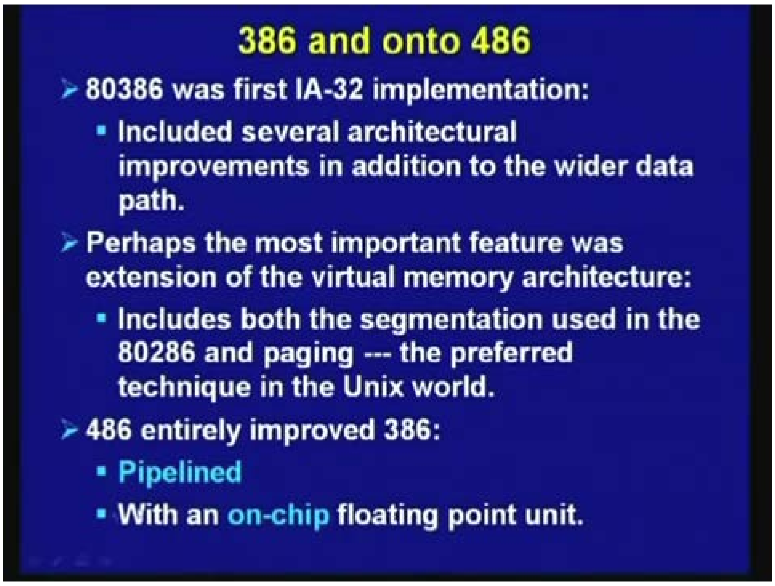


IA-32

- 8086 began an evolution:
  - Eventually resulted in IA-32 family of object code compatible microprocessors.
- IA-32 is a CISC architecture:
  - Variable length instructions and complex addressing modes.
  - Turned out to be the most dominant architecture of the time in terms of sales volume.
  - 1985: Intel 386
  - 1989: First pipelined version of IA-32 family Intel 486 was introduced.

But, in case of CISC processors, the complex instructions set computer architecture, there the instruction length is variable, and it gives you many complex addressing modes which are not provided in RISC processors. And this IA-32 processors turned out to with the most dominant architecture of the time, in terms of sales volume. That means, the computers PCs personal computers were built, not only personal computers, workstations and desktop systems were built, using these 32-bit processors and the sales volume was quite high. And as I have already mentioned, this Intel 386 was introduced in 1985 and 89, the 486 was introduced.

(Refer Slide Time: 22:39)

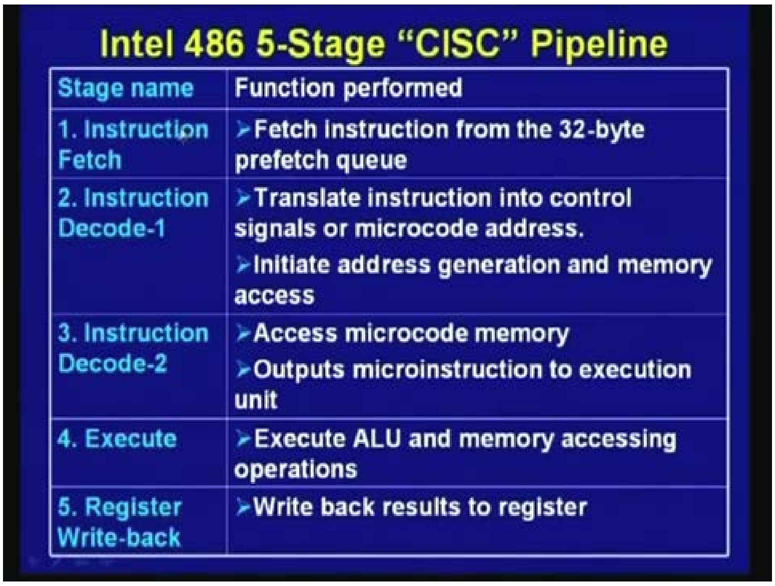


386 and onto 486

- 80386 was first IA-32 implementation:
  - Included several architectural improvements in addition to the wider data path.
- Perhaps the most important feature was extension of the virtual memory architecture:
  - Includes both the segmentation used in the 80286 and paging --- the preferred technique in the Unix world.
- 486 entirely improved 386:
  - Pipelined
  - With an on-chip floating point unit.

And as I have already mentioned in 80386, the most important feature was extension of virtual memory architecture, it includes both segmentation used in 80286 and paging. The preferred technique of Unix world, both segmentation raising are together used and in 80486 as I mentioned, it used pipelining with an on chip floating point unit, so earlier in case of 386 the pipeline that floating point unit was external.

(Refer Slide Time: 23:26)



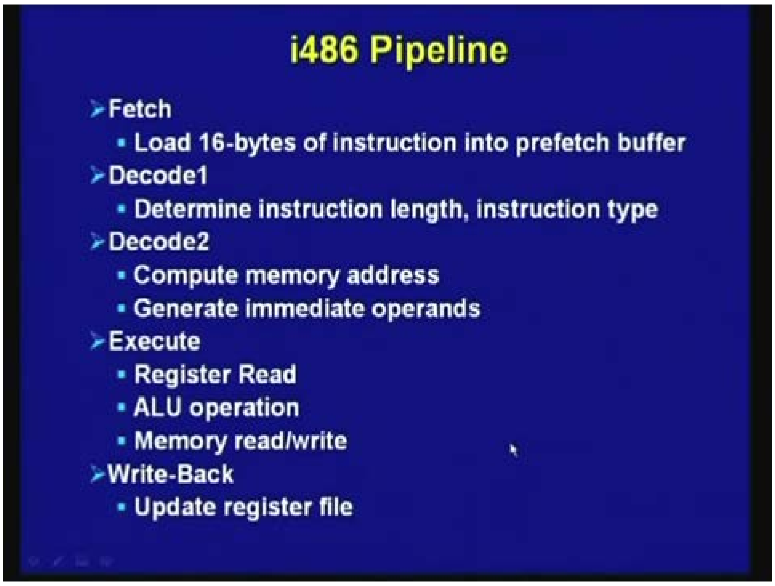
Intel 486 5-Stage "CISC" Pipeline

Stage name	Function performed
1. Instruction Fetch	➢Fetch instruction from the 32-byte prefetch queue
2. Instruction Decode-1	➢Translate instruction into control signals or microcode address. ➢Initiate address generation and memory access
3. Instruction Decode-2	➢Access microcode memory ➢Outputs microinstruction to execution unit
4. Execute	➢Execute ALU and memory accessing operations
5. Register Write-back	➢Write back results to register

But, in 486 it was put inside and so far as the pipelining is concerned as I mentioned that Intel 486, that used instruction pipelining, and that instruction pipelining for the first time in Intel processors. And it was having 5 stages instruction fetch, instruction decode 1, instruction decode 2, execute and register write back, and the various functions performing different stages are given here. Fetch instruction from the 32-bit prefetch queue, then instruction decode, translate instruction into control signals or microcode address.

And in this same stage it also performs the address generation, it initiate address generation and memory access, then instruction decode 2 access microcode memory, and it performs also output microinstruction to execution a unit. Later on as we shall see, instructions are converted into micro operations and this is what is being done that micro code memory, access memory and output microinstruction to execution unit. So, those micro operations are executed ultimately, and execute ALU and memory accessing operations, in the execute stage and in the register write back write back results to register take place.

(Refer Slide Time: 25:02)

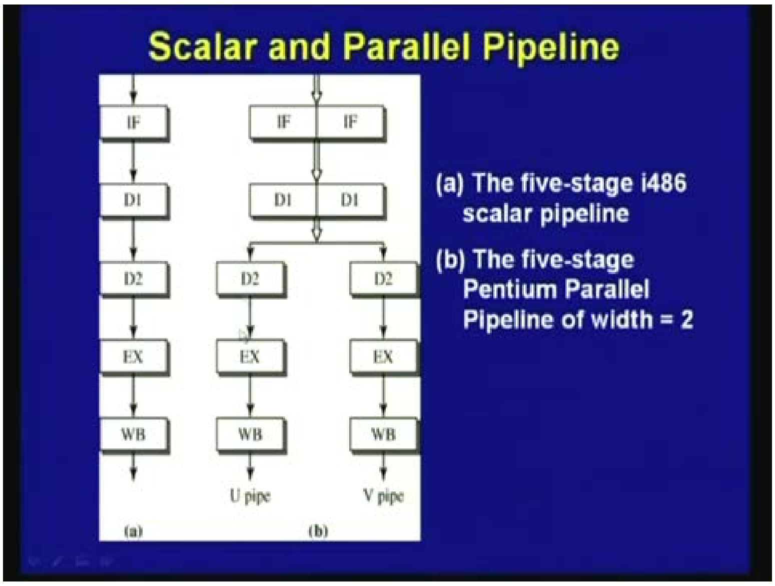


i486 Pipeline

- Fetch
  - Load 16-bytes of instruction into prefetch buffer
- Decode1
  - Determine instruction length, instruction type
- Decode2
  - Compute memory address
  - Generate immediate operands
- Execute
  - Register Read
  - ALU operation
  - Memory read/write
- Write-Back
  - Update register file

So, same thing fetch unit performs, in the fetch stage load 16 bytes of instruction into prefetch buffer, so in the prefetch buffer you can load 16 bytes of instructions. And then, the decode unit determines instruction length, instruction type, then in the decode 2 stage it performs the address calculation, it computes memory address. And generates immediate operands which is provided as part of the instruction, then in execute stage register read ALU operation and memory read write operations are performed, and as I mentioned in the write back stage of register files are updated.

(Refer Slide Time: 25:49)



Scalar and Parallel Pipeline

(a) The five-stage i486 scalar pipeline

(b) The five-stage Pentium Parallel Pipeline of width = 2

And you can see here, the 80486 stages that instruction fetch and decode 1, decode 2 execute and then, write back and later on we shall discuss about this Pentium, Pentium is an extension of this which we use superscalar. So, you can see there are two directions in which processing is performed a U pipe and V pipe, I shall come back to it little later.

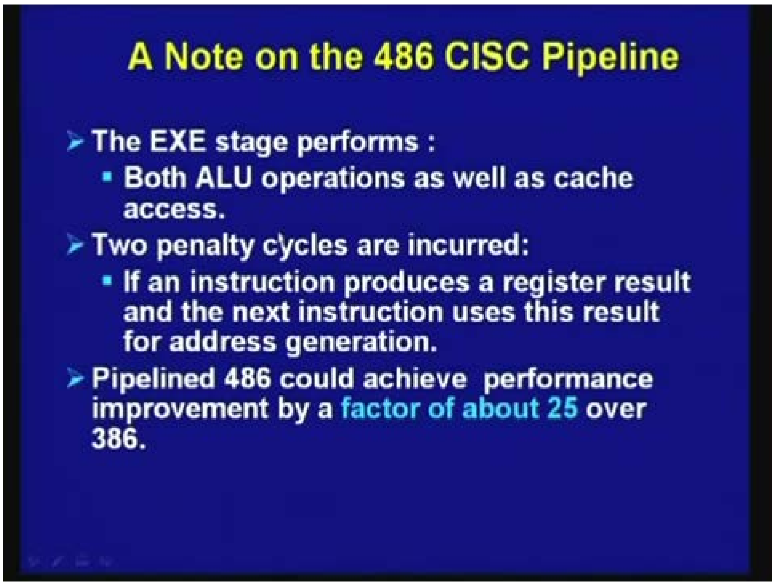
(Refer Slide Time: 26:21)

A Reflection on 486 Pipeline

- Two Decoding Stages:
  - Harder to decode CISC instructions.
  - Inevitable due to microcoded control.
  - Effective address calculation in D2.
- Multicycle Decoding Stages:
  - For more difficult decoding.
  - Stalls incoming instructions.

So, you may be wondering why two decoding stages were provided in 80486, see because it uses complex instructions computer, so decoding of instructions is much more complex compared to that of the decoding of risk instructions. So, it was necessary to have two decoding stages, because it is harder at to decode CISC instructions and it is inevitable due to microcoded control. Microcoded control that is being used in this processors and effective address calculation is done in the decoding stage 2. And multicycle decoding stages for more difficult decoding, and stalls incoming instructions, these are the two attributes of doing the decoding into 2 cycles.

(Refer Slide Time: 27:21)

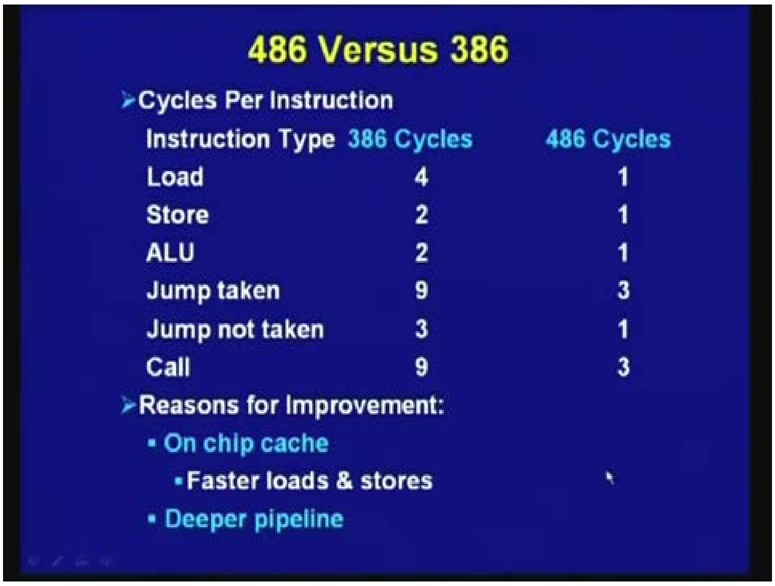


A Note on the 486 CISC Pipeline

- The EXE stage performs :
  - Both ALU operations as well as cache access.
- Two penalty cycles are incurred:
  - If an instruction produces a register result and the next instruction uses this result for address generation.
- Pipelined 486 could achieve performance improvement by a factor of about 25 over 386.

And execution stage performs both ALU operations, as well as cache memory access and whenever two penalty cycles are incurred, if an instruction produces a register result and the next instruction uses this result for address generations. So, whenever the you are doing pipelining a some kind of penalty is incurred, this is how it happens, so two penalty cycles are necessary in such situations. And pipeline 486 could achieve performance improvement by a factor of about 25 over 386, because of pipelining and the faster clock and various other features, the performance improvement was 25 times.

(Refer Slide Time: 28:19)



486 Versus 386

- Cycles Per Instruction

Instruction Type	386 Cycles	486 Cycles
Load	4	1
Store	2	1
ALU	2	1
Jump taken	9	3
Jump not taken	3	1
Call	9	3

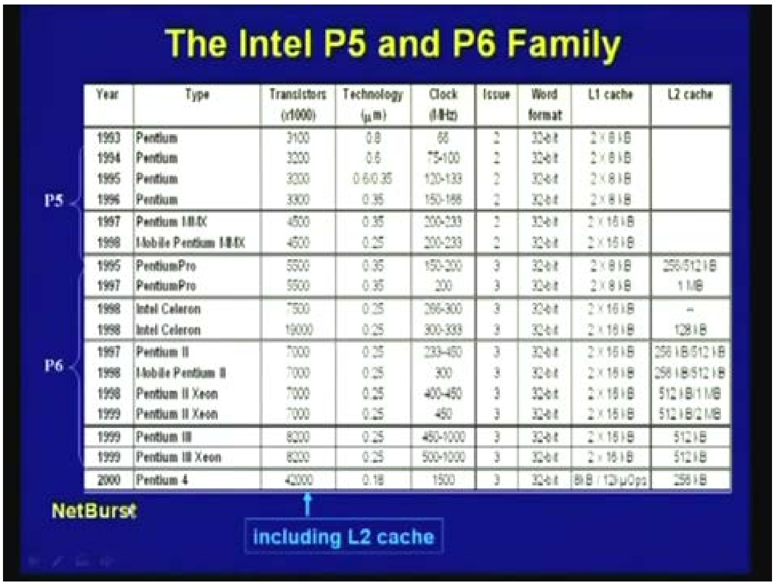
- Reasons for Improvement:
  - On chip cache
    - Faster loads & stores
  - Deeper pipeline

And you can see the number of clock cycles, that is required for the different processors 386 and 486, as you can see in case of 386 load instructions, load requests 4 cycles on the other hand in case of 486 only 1 cycle is sufficient. Similarly, store requires 4 cycles in case of 386 and 486 requires only 1 cycle, ALU operations requires 2 cycles in 386 and 486 requires only 1 cycle and jump instructions, when it is taken it requires 9 cycles.

Because, at this calculation and all this things are involved the 386 requires 9 cycles, and 486 can do it in 3 cycles and when jump is not taken, then at this calculation is not required and it is faster than when jump is taken, it requires 3 cycles and 486 it requires only 1 cycles. And subroutine calls requires 9 cycles in case of 386 and 486 requires 3 cycles, and main difference is coming, because of on chip cache that is used in 486, in 386 was not having any on chip cache memory.

But, in 486 small size cache memory were provided on chip and that has led to this improvement leading to faster loads and store. And also pipeline as we have already seen, five stage pipeline was used in 486, there is another reason for improvement in 486.

(Refer Slide Time: 30:04)



The Intel P5 and P6 Family

Year	Type	Transistors (x1000)	Technology (μm)	Clock (MHz)	Issue	Word format	L1 cache	L2 cache
1993	Pentium	3100	0.8	66	2	32-bit	2 × 8 kB	
1994	Pentium	3200	0.6	75-100	2	32-bit	2 × 8 kB	
1995	Pentium	3200	0.6/0.35	120-133	2	32-bit	2 × 8 kB	
1996	Pentium	3300	0.35	150-166	2	32-bit	2 × 8 kB	
1997	Pentium MMX	4500	0.35	200-233	2	32-bit	2 × 16 kB	
1998	Mobile Pentium MMX	4500	0.25	200-233	2	32-bit	2 × 16 kB	
1995	PentiumPro	5500	0.35	150-200	3	32-bit	2 × 8 kB	256/512 kB
1997	PentiumPro	5500	0.35	200	3	32-bit	2 × 8 kB	1 MB
1998	Intel Celeron	7500	0.25	266-300	3	32-bit	2 × 16 kB	–
1998	Intel Celeron	19000	0.25	300-333	3	32-bit	2 × 16 kB	128 kB
1997	Pentium II	7000	0.25	233-450	3	32-bit	2 × 16 kB	256 kB/512 kB
1998	Mobile Pentium II	7000	0.25	300	3	32-bit	2 × 16 kB	256 kB/512 kB
1998	Pentium II Xeon	7000	0.25	400-450	3	32-bit	2 × 16 kB	512 kB/1 MB
1999	Pentium II Xeon	7000	0.25	450	3	32-bit	2 × 16 kB	512 kB/2 MB
1999	Pentium III	8200	0.25	450-1000	3	32-bit	2 × 16 kB	512 kB
1999	Pentium III Xeon	8200	0.25	500-1000	3	32-bit	2 × 16 kB	512 kB
2000	Pentium 4	42000	0.18	1500	3	32-bit	8kB / 12kμOps	256 kB

P5

P6

NetBurst

including L2 cache

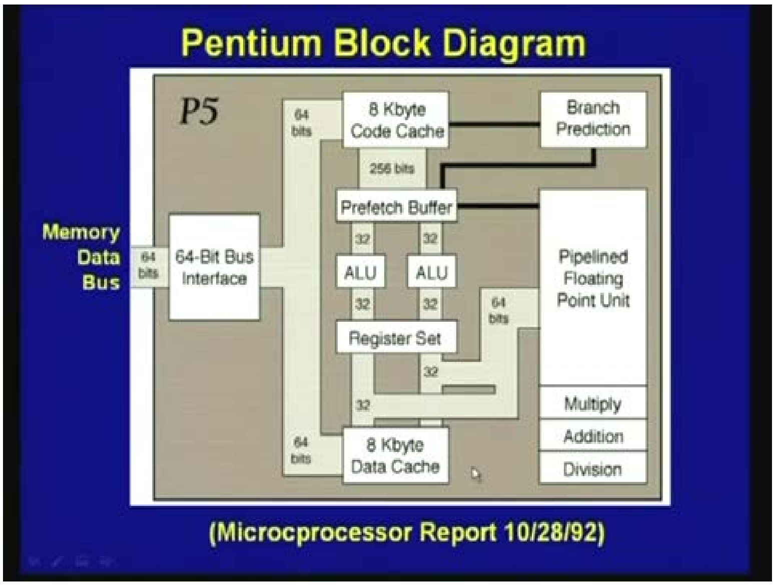
So, these are the different Pentium family, and how does it evolve is shown here and you can see the number of transistors has increased from 3100 1000 that means, 3100 100 transistors to 42000 1000 transistors that means, 42 million transistors that is used in Pentium 4. So, the number of transistors has increased and that has been possible with the advancement of VLSI technology, and technology has also improved as you can see from 0.8 microns gradually, then the size of the devices has synced have becomes smaller and smaller from 0.8 microns to point 0.1 microns.

And because of that, it was possible to incorporate larger number of transistors in later processors, so Pentium from 1933 to 1936 Pentium series of processors. And you can see clock frequency was in the range of 66 Megahertz to 166 Megahertz and as I have already mentioned it use superscalar, so to two were performed, two instructions were issued in Pentium and word size was 32-bit and L 1 cache was restricted to 2 8 kilo byte. So, one for 2 8 kilo byte cache memory was on chip and of course, there was no L 2 cache in Pentium possessors.

Then Pentium MMX and mobile Pentium MMX, which were introduced in 1997 and 1998, the clock frequency were enhanced and also the L 1 cache memory size was also enhanced. Then in Pentium Pro which was introduced in 1995 the clock frequency was remained more or less same, you can see as Pentium MMX. But, L 2 cache was introduced in Pentium processors and in Intel Celeron processors, you can see the transistors have been increased clock frequency, has been enhanced. And of course, the cache memory has been made from 8 kilo bytes to 16 kilo byte, and second level cache was, in case of Intel Celeron in first version there was no L 2, cache in second version there was L 2 cache, then Pentium II, mobile Pentium II and so on.

You can see the enhancement that has taken place, in terms of increased cache memory, increased L 2 cache size increase a clock frequency and then, we shall come to the net burst we shall discuss about that little later. So, this is how the Pentium 5 and P5 and P6 have evolved over the years and from nineteen 1993 to 2000.

(Refer Slide Time: 33:34)



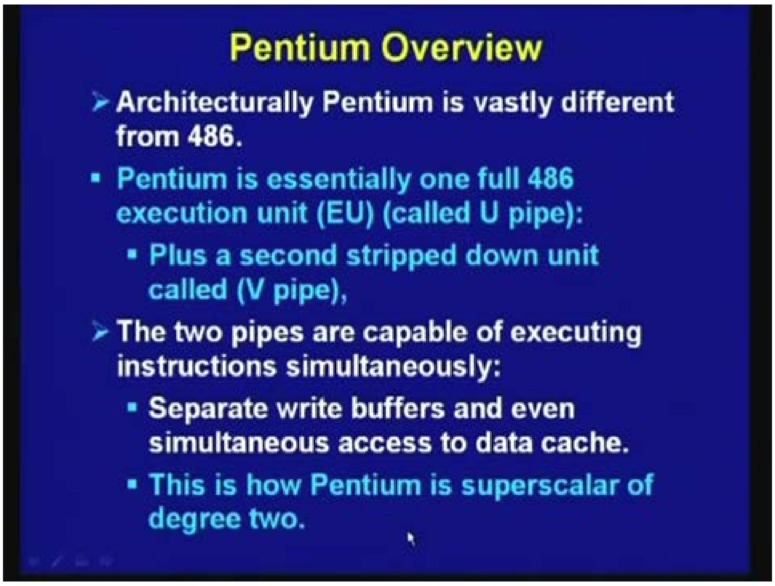
Pentium Block Diagram

(Microcprocessor Report 10/28/92)

This is the internal architecture of Pentium P5, you can see here the floating point unit is built, in this is that bus interface unit which interfaces with the memory and IO devices. And the internal bus is 64-bits, external bus is also 64-bits, in case of Pentium and the cache memory is shown here, and that instruction cache 8 kilobyte of instruction cache and 8 kilo byte of data cache, and we have there are two ALU. So, it uses superscalar process architecture to logic units, these are the register sets and prefetch buffer, the instructions are prefetched and stored there.

And it has got separate branch prediction unit built in and then, as I mentioned multiplication division, these are available in the hardware which were not available in the earlier processors. So, in addition to floating point a unit, processing unit multiply addition and division, these are all available in built in hardware as a consequence it gave you very faster instruction processing.

(Refer Slide Time: 35:04)

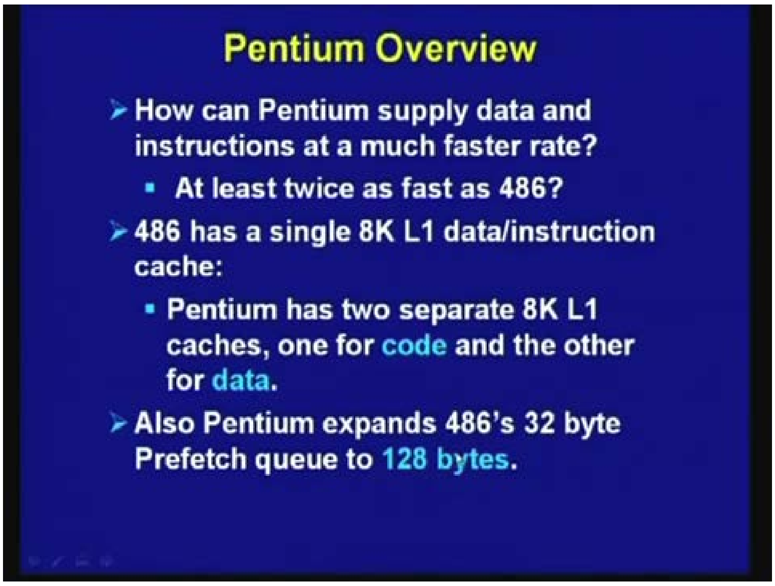


Pentium Overview

- Architecturally Pentium is vastly different from 486.
  - Pentium is essentially one full 486 execution unit (EU) (called U pipe):
    - Plus a second stripped down unit called (V pipe),
- The two pipes are capable of executing instructions simultaneously:
  - Separate write buffers and even simultaneous access to data cache.
  - This is how Pentium is superscalar of degree two.

So, this is some overview of the Pentium processor architecture, Pentium is vastly different from 486 as we have already seen, Pentium is essentially one full 486 execution unit called U pipe and V pipe. As I said there are two processing units U pipe and V pipe and two pipes are capable of executing instruction simultaneously, there are separate write buffers and even simultaneous access to data cache. So, which is represented here data cache data cache, so simultaneously they can access, you can see 32-bit 8 byte through, the register can set can also be accessed simultaneously.

(Refer Slide Time: 36:04)

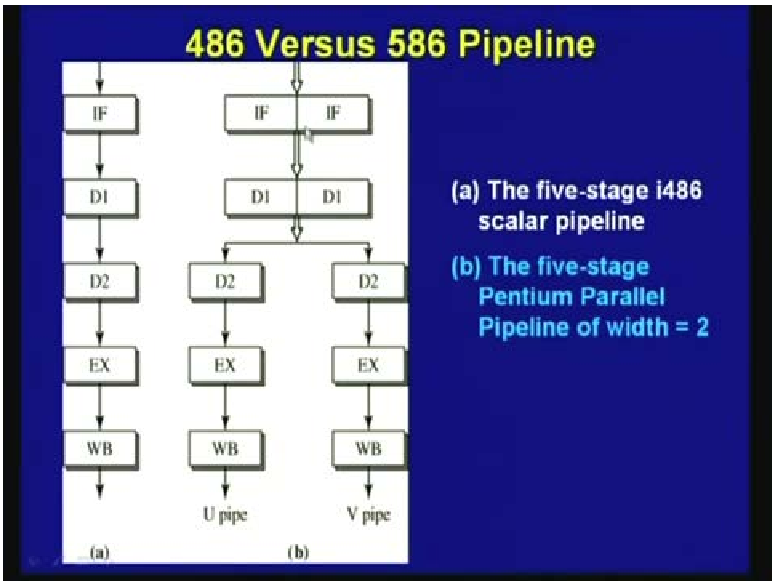


Pentium Overview

- How can Pentium supply data and instructions at a much faster rate?
  - At least twice as fast as 486?
- 486 has a single 8K L1 data/instruction cache:
  - Pentium has two separate 8K L1 caches, one for code and the other for data.
- Also Pentium expands 486's 32 byte Prefetch queue to 128 bytes.

This how Pentium is superscalar of degree 2 that means, there are two processing units, so as we have seen in the diagram it is at least twice as fast as 486, because of the internal architecture. And I have already mentioned about this Pentium expands, 32 byte prefetch queue to 128 bytes, we have seen that prefetch is present here, which is 128 bytes.

(Refer Slide Time: 36:36)



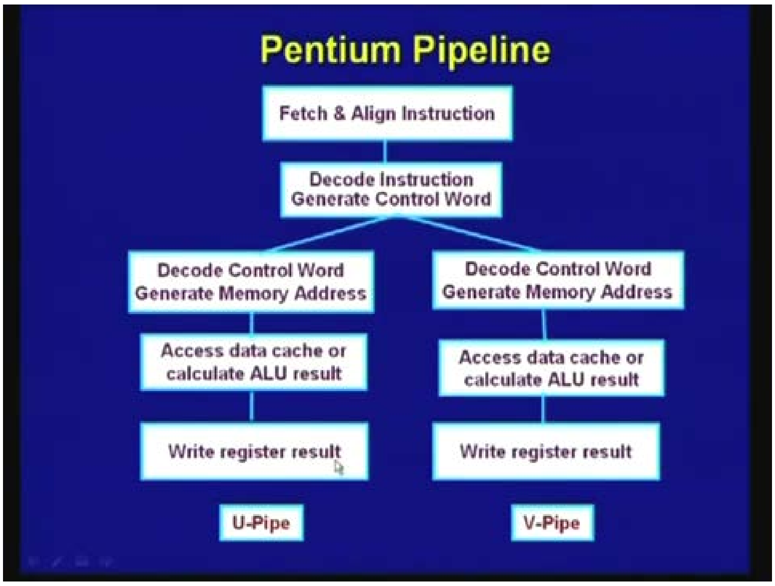
486 Versus 586 Pipeline

(a) The five-stage i486 scalar pipeline

(b) The five-stage Pentium Parallel Pipeline of width = 2

So, this shows that Pentium superscalar processor, it has got two pipes U pipe and V pipe separate execution in it is, separate write buffer, separate decode unit and so on. So, instruction page instruction decode, two stages of instruction decode, execution write back these are done for the two different execution units.

(Refer Slide Time: 37:04)



Pentium Pipeline

Fetch & Align Instruction

Decode Instruction
Generate Control Word

Decode Control Word
Generate Memory Address

Access data cache or
calculate ALU result

Write register result

U-Pipe

Decode Control Word
Generate Memory Address

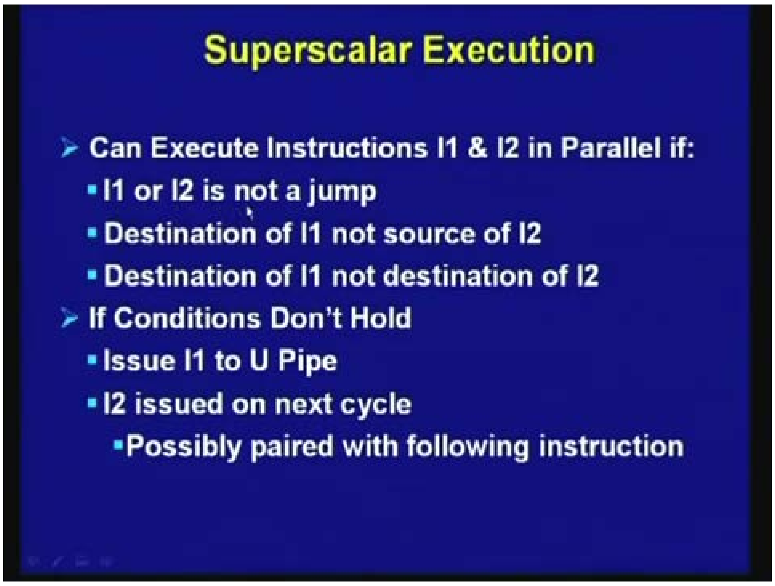
Access data cache or
calculate ALU result

Write register result

V-Pipe

And this is logically how it really works this one, fetch and align instructions, decode instruction generate control word, then it has got two separate pipes, decode control word, generate memory access. So, same thing is repeated in U pipe and V pipe access data cache or calculate ALU result and write a register result, that is your write back stage.

(Refer Slide Time: 37:34)



Superscalar Execution

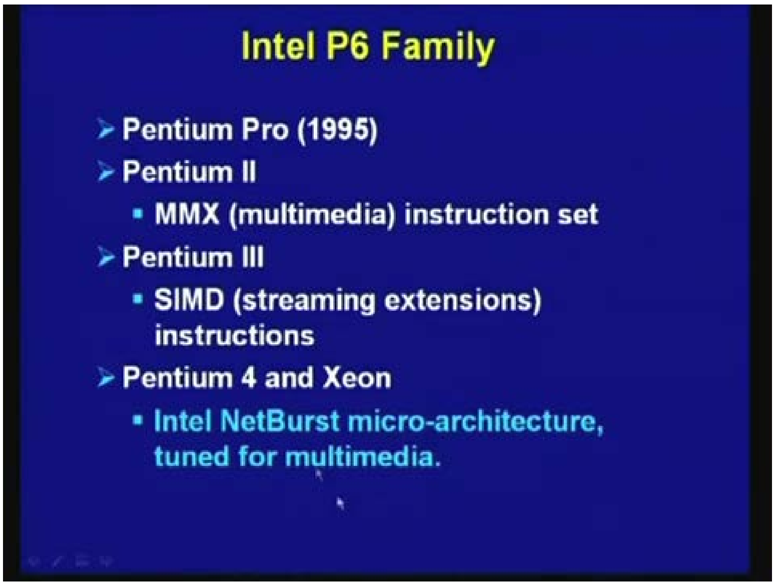
- Can Execute Instructions I1 & I2 in Parallel if:
  - I1 or I2 is not a jump
  - Destination of I1 not source of I2
  - Destination of I1 not destination of I2
- If Conditions Don't Hold
  - Issue I1 to U Pipe
  - I2 issued on next cycle
    - Possibly paired with following instruction

The superscalar execution can execute instruction I that 1 and instruction 2 in parallel, if instruction 1 and instruction 2 is not a jump, so you see there is there is some restriction, it cannot always perform execution of two instructions together, when it cannot perform is highlighted here. So, can execute instructions, instruction 1 and instruction 2 in parallel that means, after fetching those instructions, they are stored in a buffer. Then they are checked which pairs can be executed in U pipe and V pipe two separate pipes.

So, they are executed in parallel, they can be executed in parallel if instruction 1 and instruction 2 are instruction 2 is not jump destination of instruction one is not the source of instruction two destination of instruction 1 is not destination of instruction 2. So, this is checked and if this is not true, then only I 1 and I 2 can execute it in parallel, otherwise two instructions cannot be issued. So, as we know whenever we use multiple issue processors, it may be not be necessary to all the execution units, may not be always busy.

So, if conditions hold issue I 1 to U pipe and I 2 issued on next cycles, so in such cases you have to do it serially and that means, that instruction 2 can be paired with the following instruction. So, this is how superscalar execution takes place in Pentium processors.

(Refer Slide Time: 39:34)

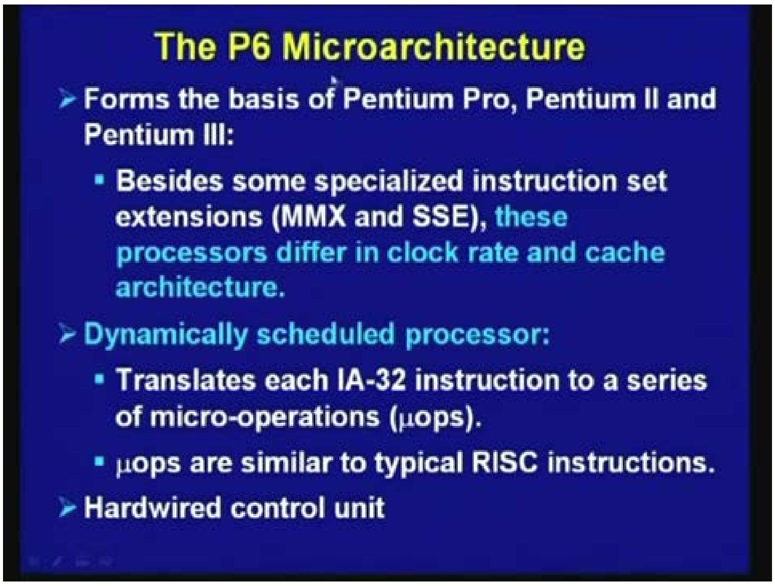


Intel P6 Family

- Pentium Pro (1995)
- Pentium II
  - MMX (multimedia) instruction set
- Pentium III
  - SIMD (streaming extensions) instructions
- Pentium 4 and Xeon
  - Intel NetBurst micro-architecture, tuned for multimedia.

Then, as I mentioned in 1995 Pentium Pro was introduced and Pentium II was introduced that is your multimedia instruction set. So, that multimedia processing can be done at a faster rate, than the that SSE processor that I mentioned, which allows you streaming, which is essentially SIMD instructions allows you streaming extensions. So, this streaming extensions provided included in Pentium III and then Pentium 4 and Xeon they use Intel NetBurst micro architecture and that is also tuned for multimedia, I shall briefly highlight their internal architecture.

(Refer Slide Time: 40:20)

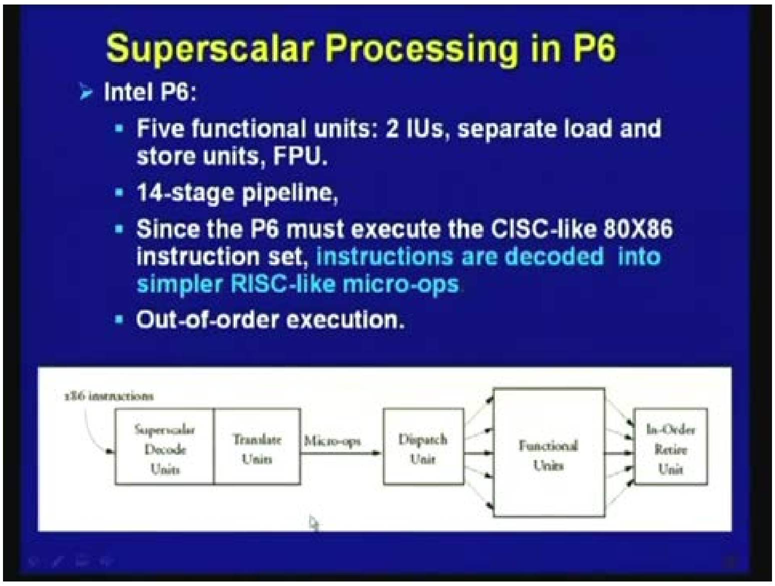


The P6 Microarchitecture

- Forms the basis of Pentium Pro, Pentium II and Pentium III:
  - Besides some specialized instruction set extensions (MMX and SSE), these processors differ in clock rate and cache architecture.
- Dynamically scheduled processor:
  - Translates each IA-32 instruction to a series of micro-operations (μops).
  - μops are similar to typical RISC instructions.
- Hardwired control unit

This is the Pentium 6, P6 micro architecture it forms the basis of Pentium Pro, Pentium II and Pentium III, besides some specialized instruction set extensions like MMX and SSE, these processors differ in clock rate and cache architecture. And these are dynamically schedule processors translates each IA-32 instructions to a series of micro operations we have already mentioned about that how instructions are converted into micro operations, which are executed. And micro operations are similar to typical RISC instructions, and that allows you hardware control unit.

(Refer Slide Time: 41:05)



Superscalar Processing in P6

- Intel P6:
  - Five functional units: 2 IUs, separate load and store units, FPU.
  - 14-stage pipeline,
  - Since the P6 must execute the CISC-like 80X86 instruction set, instructions are decoded into simpler RISC-like micro-ops
  - Out-of-order execution.

x86 instructions

Superscalar Decode Units

Translate Units

Micro-ops

Dispatch Unit

Functional Units

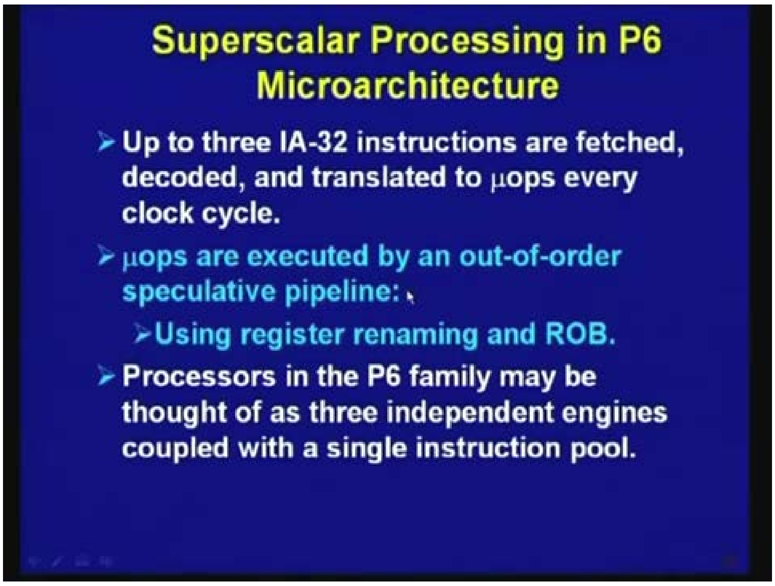
In-Order Retire Unit

And this is how the conversion of instructions into micro operations takes place shown here, so these are the X 86 instructions, which are fed to superscalar decode unit. Then there is a translate units, these two together converts the C 86 instructions into RISC like micro operations, this RISC like micro operations are sent to the despatch unit, then the despatch unit identifies to which function unit which operation will go. So, you can have different functional units, it can be some fixed point units, floating point unit, fetch unit, so depending on what type of micro operation it is there those are fed into the functional units.

And then, functional units can perform the execution and it may lead to some kind of out of order execution, so since the out of order execution has to be converted into in order where the instructions were generated in the program. In the same order results should be produced output should be generated, that is achieved with the help of this in order retire unit, with the help of this in order retire unit where writing into the register takes place in the appropriate sequence.

So, that the way it should take place I mean specified by the program, in the same way it is done here and an Intel Pentium P6, then that Pentium 6 has five functional units, 2 instructions units, separate load and store units and floating point unit. And it has used 14 stage pipeline, so in context to 5 stage pipeline that is used in Pentium 4, Pentium 6 uses ah much differ pipeline that is 14 stage pipeline. And since, the Pentium P6 must execute the CISC like, X 86 instruction instructions set are decoded into simpler RISC like micro operation as I have already mentioned. And out of order execution take place which have already mentioned with the help of this diagram.

(Refer Slide Time: 43:33)



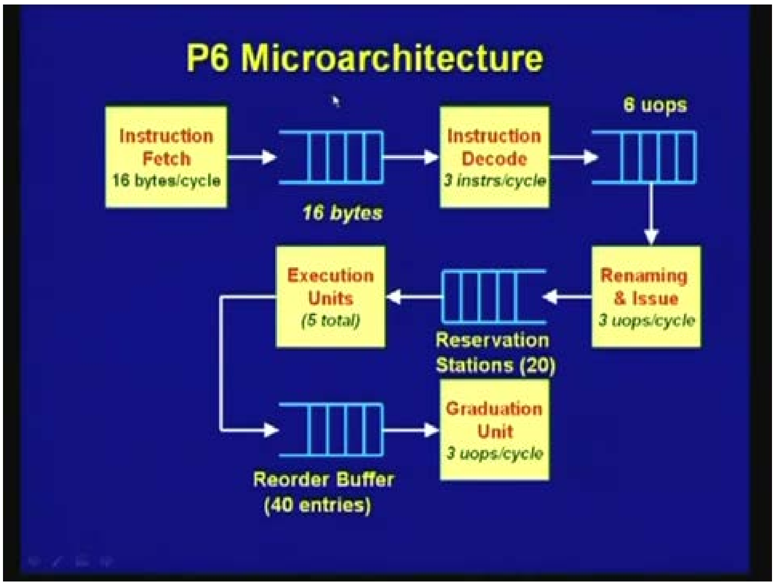
Superscalar Processing in P6 Microarchitecture

- Up to three IA-32 instructions are fetched, decoded, and translated to μops every clock cycle.
- μops are executed by an out-of-order speculative pipeline:
  - Using register renaming and ROB.
- Processors in the P6 family may be thought of as three independent engines coupled with a single instruction pool.

So, we in Pentium 6 up to three IA-32 instructions are fetched decoded and translated into micro operations in every clock cycle, and micro operations are executed by the out of order speculative pipeline. So, here you are using some kind of a speculative execution, and for executing micro operations and you have already discussed in detail the register renaming, then reorder buffer that is being used, whenever you do out of order execution. And which have been incorporated in this Pentium 6 processors to facilitate this out of order execution of the micro operations.

So, processors in Pentium 6 family, may be thought of as three independent engines coupled with a single instruction pool. So, instructions are fetched, instructions are kept in that instruction buffer, from which the instructions are going into the execution units three independent execution units, and the execution take place parallelly.

(Refer Slide Time: 44:53)



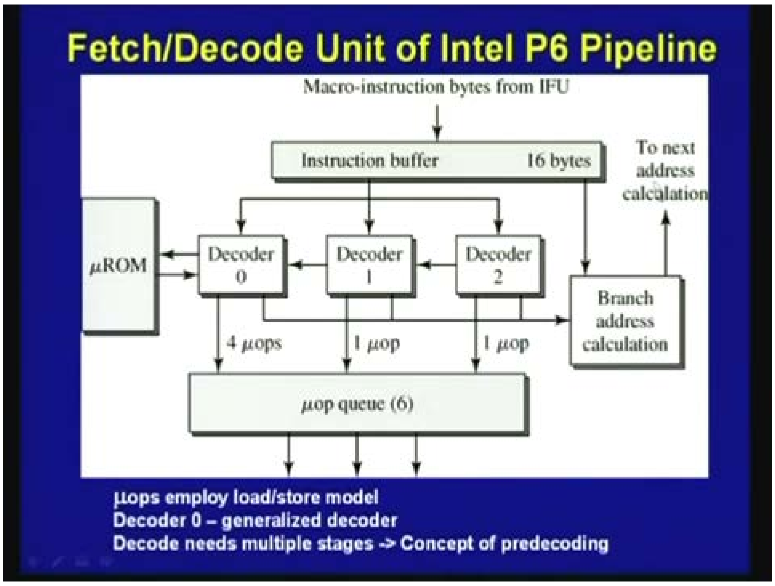
P6 Microarchitecture

So, this diagram shows functionally how exactly this happens, so this instruction fetch unit as I mentioned it can it fetches 16 bytes in every cycles. So, 16 bytes of instructions are fetched and they are stored in a instruction queue. So, then instruction queue from here to three instruction per cycle goes to instruction decode unit, so the three instructions are decoded, which are converted into micro operations, as it is mentioned here, they translated into micro operations. Those micro operations, so it is converted into micro operations, 6 micro operations three instructions can lead to 6 micro operations.

And there sent to the renaming and issue unit, 3 micro operations per cycles from this buffer it goes to the renaming and issue unit and so, 3 micro operations per cycle that is rate at which it goes, then it goes to the reservation stations. And in the reservation station there are 20 entries, 20 buffers are there, reservation stations are nothing but, buffers and 20 buffers are provided, from this reservation stations it goes to the functional execution units.

As I have already mentioned there are five total functional units, including one floating point, as we have already seen separate load storing unit and floating point unit. So, the outputs are generated they go to the reorder buffers, because the execution the out of order execution can take place and results will be produced in different order, so they will go to the reorder buffer, which has got 40 entries. So, and then which will go to the graduation unit where 3 micro operations per cycle, at the rate of 3 micro operations per cycle it performs the graduation unit, so this is how the Pentium 6 micro architecture performs execution of instructions.

(Refer Slide Time: 47:23)



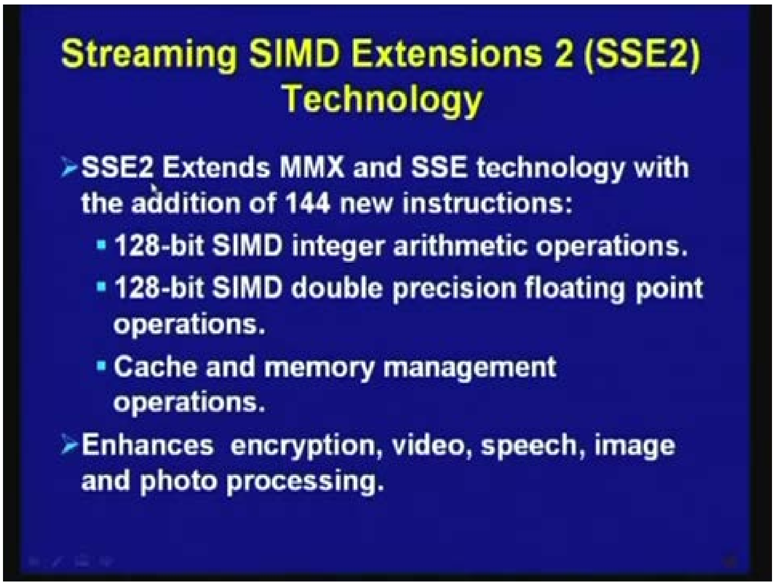
Fetch/Decode Unit of Intel P6 Pipeline

μops employ load/store model
Decoder 0 – generalized decoder
Decode needs multiple stages -> Concept of predecoding

So, this is the schematic diagram of the fetch and decoding unit of the Intel Pentium 6 pipeline, so this is the instruction buffer after fetching instructions, which are stored in instruction buffer as you have seen there are 16 bytes can be stored. And which will go to the different functional decoder units, you can see 1, 2 then decoder 2, so 6 micro operations are generated. And this is that micro control memory is stored micro program control memory stored in ROM, the control unit can be of two types hardware control unit and micro program control unit.

So, whenever you use micro control unit that micro operations are stored in a separate memory known as micro program memory, and this is where that it is being stored and that micro those operations are fetched from here. And then, decoding take place, and this micro operation queue stores these 6 micro operations, and this is branch address calculation take place for generating the next instructions, so this is the fetch and decode unit of the Pentium 6 pipeline

(Refer Slide Time: 49:04)



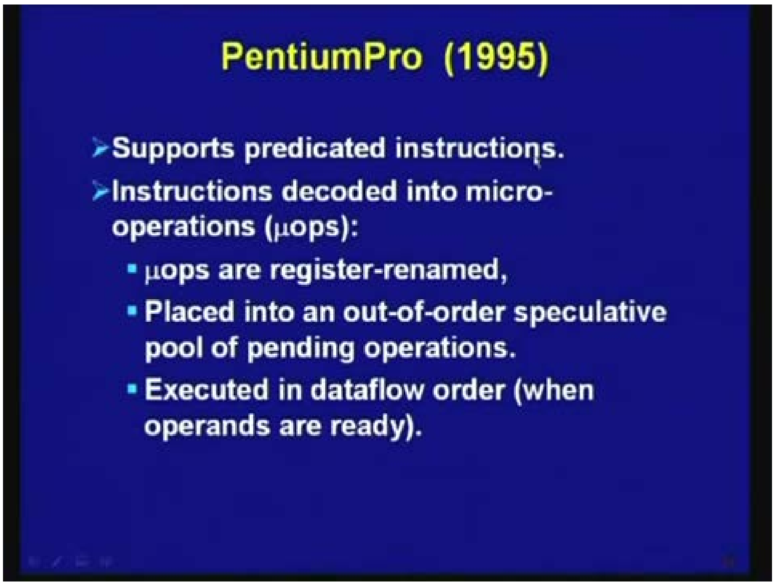
Streaming SIMD Extensions 2 (SSE2) Technology

- SSE2 Extends MMX and SSE technology with the addition of 144 new instructions:
  - 128-bit SIMD integer arithmetic operations.
  - 128-bit SIMD double precision floating point operations.
  - Cache and memory management operations.
- Enhances encryption, video, speech, image and photo processing.

And then, that streaming SIMD extensions which is known as SSE 2 technology, so SSE 2 extends that MMX and SSE technology with the addition of 144 new instructions. So, many new instructions have been added, like 128-bits SMID integer arithmetic operations, 128-bit SMID double precision floating point operations, cache and memory management operations. And with the help of this additional instructions it allows enhanced encryption video, speech, image and photo processing.

That means, various types of applications can be performed with much higher efficiency with the help of this enhanced instruction set.

(Refer Slide Time: 50:07)



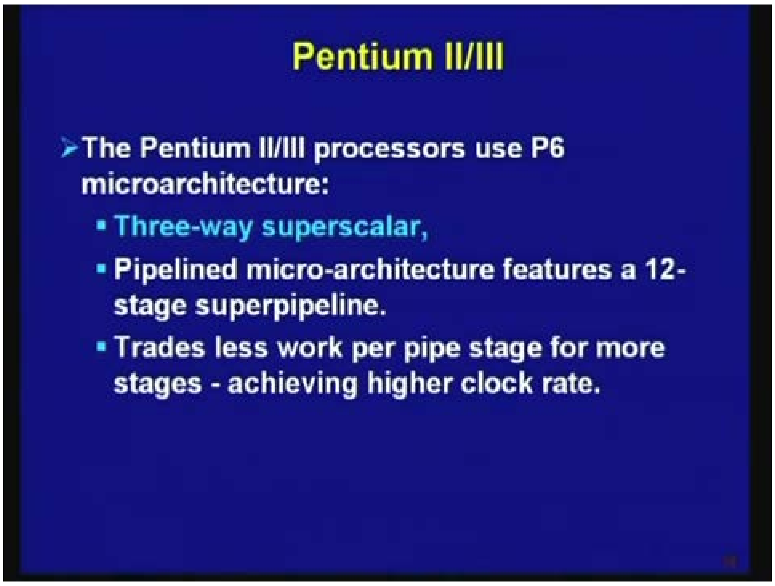
PentiumPro (1995)

- Supports predicated instructions.
- Instructions decoded into micro-operations (μops):
  - μops are register-renamed,
  - Placed into an out-of-order speculative pool of pending operations.
  - Executed in dataflow order (when operands are ready).

Then the Pentium Pro was introduced in the year 1995, it supports predicated instructions, instructions that are decoded into micro operation just like other processors and micro operations are register renamed and placed into out of order speculative pool for pending operations. And then another new feature that is been added here it is execution is done in dataflow order; that means, as soon as the operands in data flow machines were invented at some point of time.

The basic idea was as soon as the operands are available performs that execution of that particular operation. So, here also some out similar concept has been used the data flow order. That means when operands are available, operands are ready you perform that particular operation and based on this Pentium Pro execution of instructions take place.

(Refer Slide Time: 51:14)

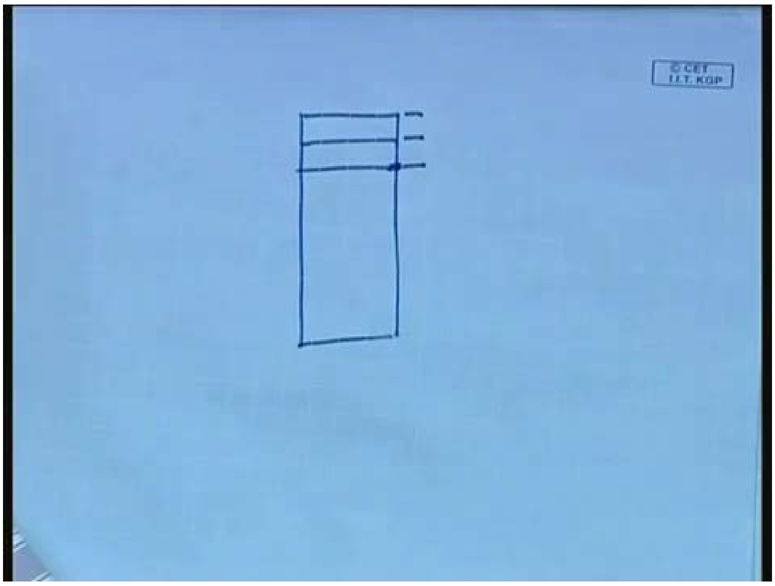


Pentium II/III

- The Pentium II/III processors use P6 microarchitecture:
  - Three-way superscalar,
  - Pipelined micro-architecture features a 12-stage superpipeline.
  - Trades less work per pipe stage for more stages - achieving higher clock rate.

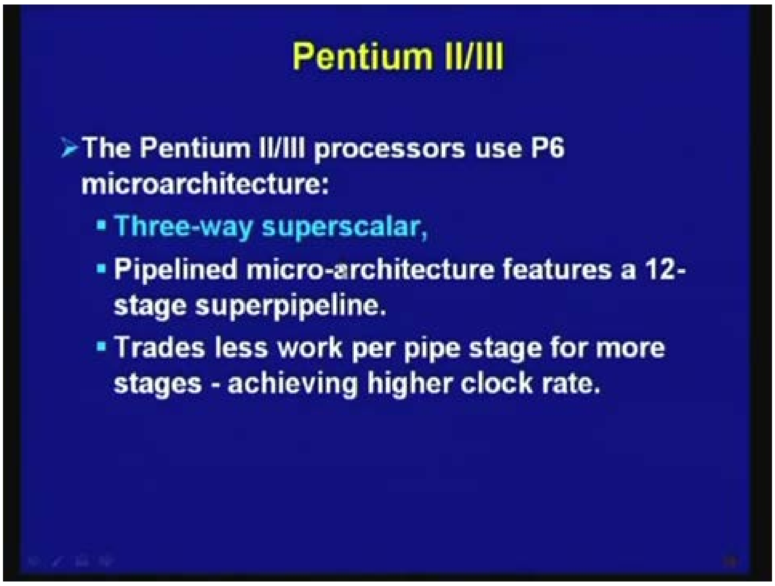
Then Pentium II and III processors use Pentium 6 micro architecture, which are three way superscalar as I have already mentioned. Then pipelined micro architecture features of 12 stage super pipeline and in case of super pipeline as you as you know in a single cycle.

(Refer Slide Time: 51:40)



Say, this is the conventional pipeline, so in a single cycle two separate operations can be performed; that means, in may be in half of the cycles one operations, another half of the cycle another operation and which is known as superscalar.

(Refer Slide Time: 51:58)

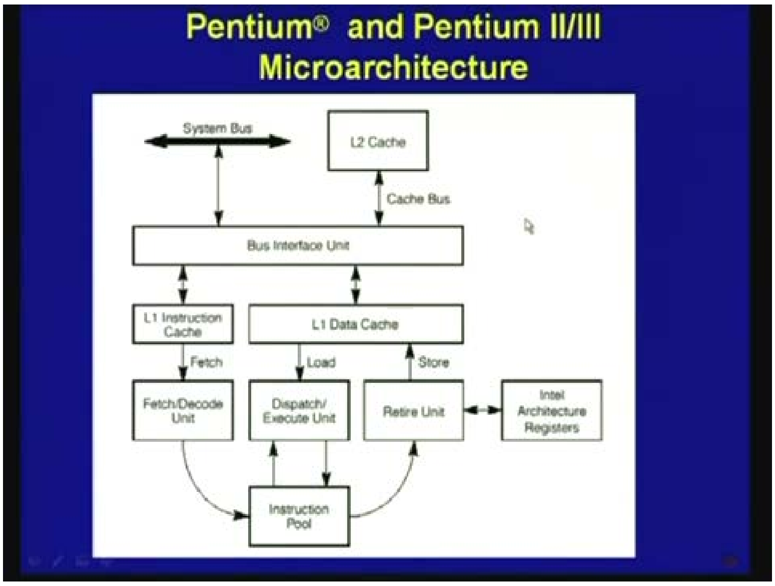


Pentium II/III

- The Pentium II/III processors use P6 microarchitecture:
  - Three-way superscalar,
  - Pipelined micro-architecture features a 12-stage superpipeline.
  - Trades less work per pipe stage for more stages - achieving higher clock rate.

So, this it is uses superscalar three way superscalar execution to improve the performance. So, pipeline micro architecture features of 12 stage super pipeline and it trades less work per pipe stage for more stages. So; that means, it has got more number of staging achieving higher clock rate, as we know the number of stages decides the clock frequency more the number of stages and higher will be the clock frequency. So, that was the another feature that has been used here. So, more numbers of stages have been used, so that the it can use higher clock frequency.

(Refer Slide Time: 52:39)

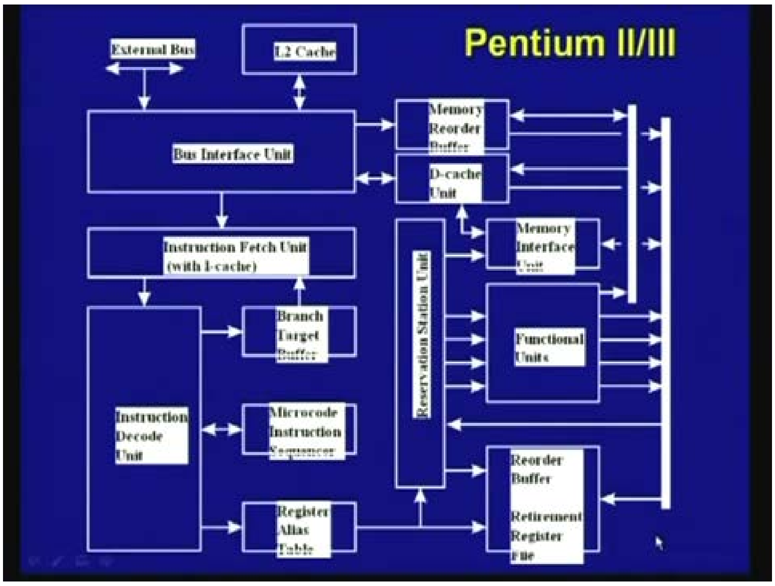


Pentium® and Pentium II/III Microarchitecture

So, this is the Pentium II and III micro architecture is a bus interface unit system bus from which instructions are fetched and this is the L 2 cache and here is your L 1 instructions cache and L 1 data cache. So, the instructions and data which are fetched from the memory using the bus interface unit as stored in the L 1 instruction and L 1 data cache and also it uses L 2 cache. And then it has got fetch and decode unit despatch and execute unit and retire unit.

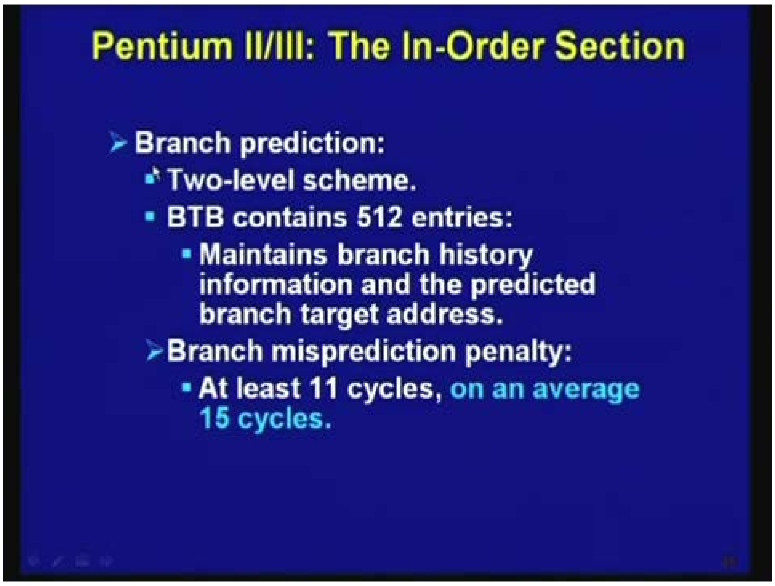
So, after fetching it goes to the instruction goes to the instruction fetch and decode unit for execution and whenever it has to be loaded it goes to the despatch and execute unit. And this is where the instruction pool is stored after and then it is it goes to the retire unit for storing the result in the registers. And here you have got your registers where the retire unit the registers values and also, so this is this keeps you Pentium II and III micro architecture.

(Refer Slide Time: 54:18)


Pentium II/III

So, same thing where different units are shown bus interface unit instruction fetch instruction decode unit, branch target buffer, sequencer instruction, memory reorder buffer data cache, instruction cache is here L 2 cache. And this is the memory interface unit and various functional units are shown here and this is reorder buffer and retirement register unit. So, this is the micro architecture of Pentium II and III.

(Refer Slide Time: 54:57)

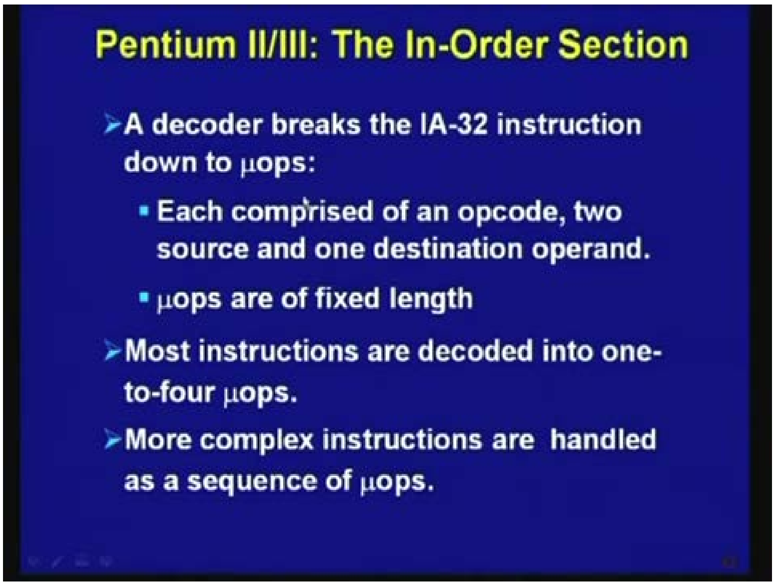


Pentium II/III: The In-Order Section

- Branch prediction:
  - Two-level scheme.
  - BTB contains 512 entries:
    - Maintains branch history information and the predicted branch target address.
  - Branch misprediction penalty:
    - At least 11 cycles, on an average 15 cycles.

So, it has got in order section and out of order section, so branch prediction unit it uses two level scheme for branch prediction using branch prediction buffer containing 512 entries. So, it maintains branch history information and the predicted branch target address. So, 512 entries history information as well as the branch target address is stored, which allows you branch prediction in a very effective way. And whenever prediction is not correct in its quite heavy at least eleven cycle mis prediction penalty is there and that is the minimum and on an average 15 cycles of mis prediction penalty is there.

(Refer Slide Time: 55:52)

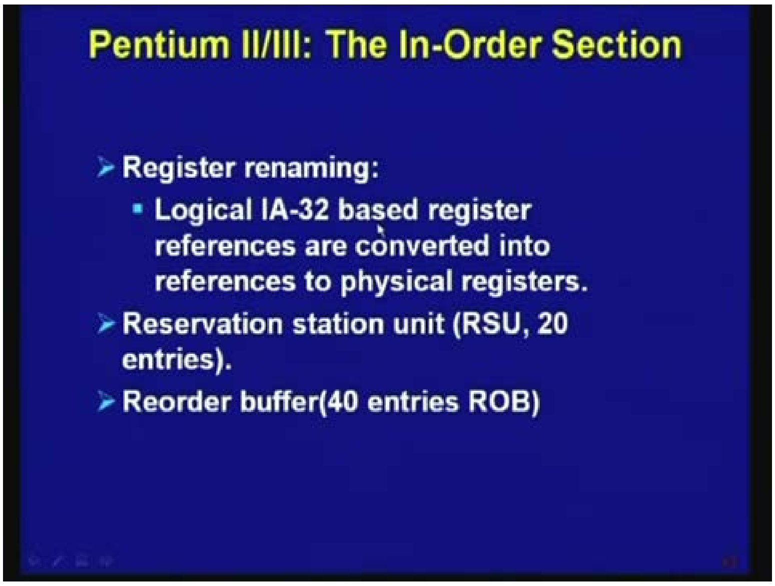


Pentium II/III: The In-Order Section

- A decoder breaks the IA-32 instruction down to μops:
  - Each comprised of an opcode, two source and one destination operand.
  - μops are of fixed length
- Most instructions are decoded into one-to-four μops.
- More complex instructions are handled as a sequence of μops.

And decoder breaks IA-32 instruction down to micro operations which I have already mentioned each comprised of an opcode two source and one destination operand and micro operations are of fixed length. Most instructions are decoded into one to four micro operations, so more complex instructions are handled as a sequence of micro operations.

(Refer Slide Time: 56:17)

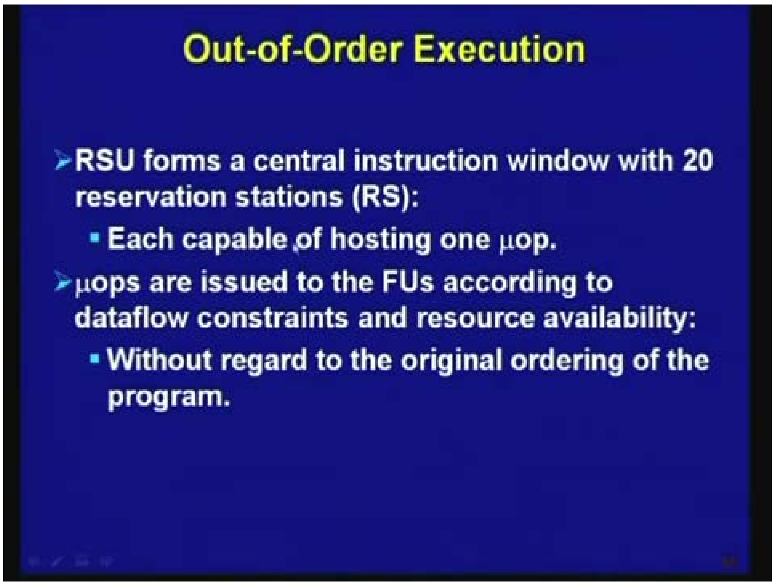


Pentium II/III: The In-Order Section

- Register renaming:
  - Logical IA-32 based register references are converted into references to physical registers.
- Reservation station unit (RSU, 20 entries).
- Reorder buffer(40 entries ROB)

So, this is the in order sections that register renaming is done logical IA-32 based register references are converted into references to physical registers with using register renaming; then it just got register reservation station unit with 20 entries as I have already mentioned and reorder buffer 40 entries.

(Refer Slide Time: 56:43)

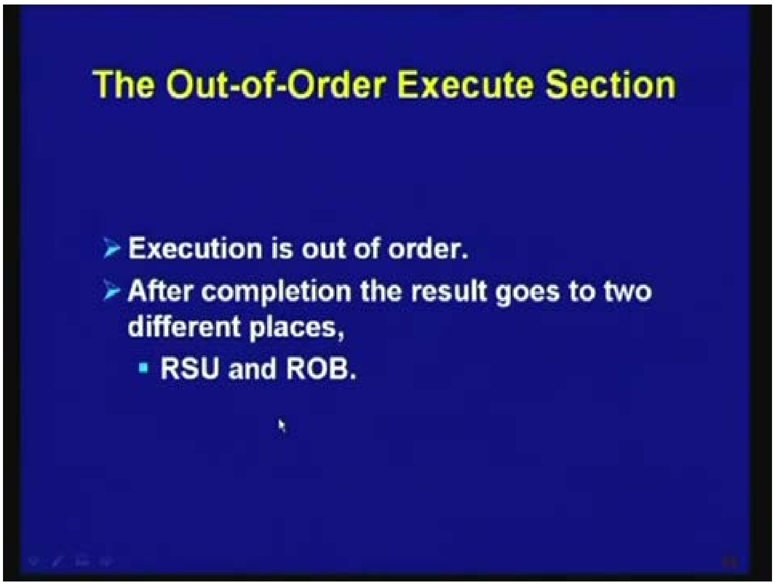


Out-of-Order Execution

- RSU forms a central instruction window with 20 reservation stations (RS):
  - Each capable of hosting one μop.
- μops are issued to the FUs according to dataflow constraints and resource availability:
  - Without regard to the original ordering of the program.

And then out of order execution is performed with the help of that RSU forms a central instruction window with 20 reservation stations, each capable of hosting 1 micro operation. And micro operations are issued to the functional units according to the dataflow constraints and resource availability. So, without regard to the original ordering of the program, so using the dataflow concept it performs the out of order execution.

(Refer Slide Time: 57:11)

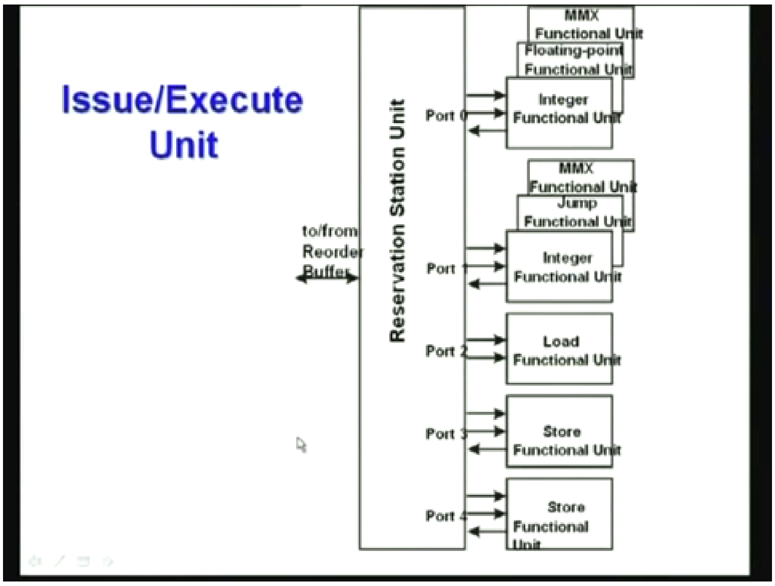


The Out-of-Order Execute Section

- Execution is out of order.
- After completion the result goes to two different places,
  - RSU and ROB.

So, out of order execution with the help of RSU and ROB.

(Refer Slide Time: 57:19)



Issue/Execute Unit

to/from Reorder Buffer

Reservation Station Unit

Port 0

MMX Functional Unit

Floating-point Functional Unit

Integer Functional Unit

Port 1

MMX Functional Unit

Jump Functional Unit

Integer Functional Unit

Port 2

Load Functional Unit

Port 3

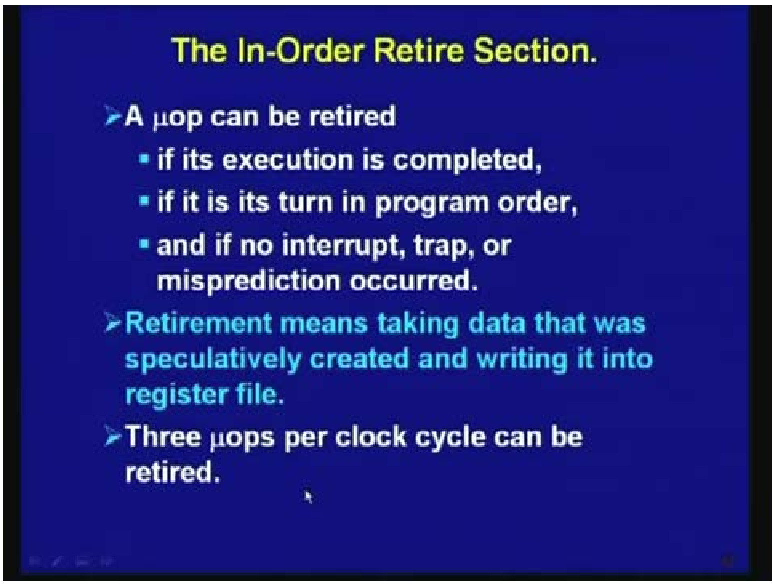
Store Functional Unit

Port 4

Store Functional Unit

And this is that issue and execute unit it makes functional unit integer unit. So, there are three such units MMX functional unit, jump functional unit and integer functional unit. So, Port 1, Port 0, Port 1 and Port 2 Port 2 and then load units. So, two integer unit and one load store unit, load unit, store unit 2, 3, 4 ports is provided, so these are the reservation station unit.

(Refer Slide Time: 57:59)

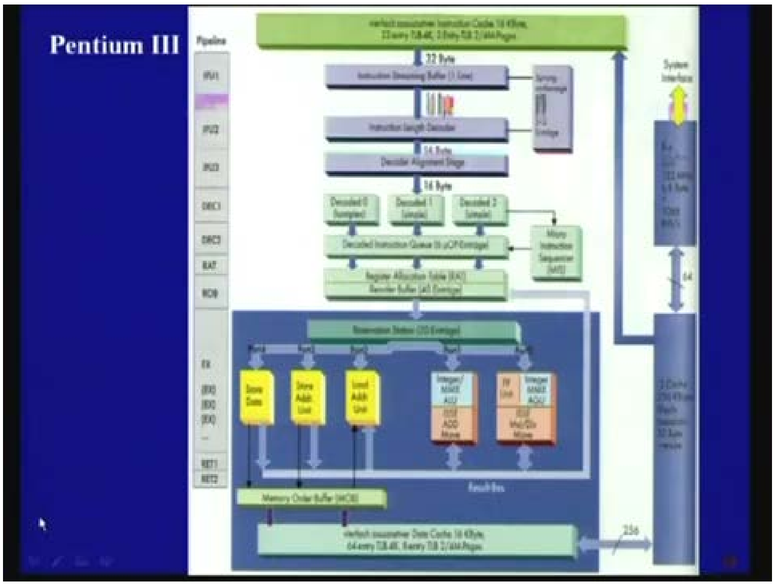


The In-Order Retire Section.

- A μop can be retired
  - if its execution is completed,
  - if it is its turn in program order,
  - and if no interrupt, trap, or misprediction occurred.
- Retirement means taking data that was speculatively created and writing it into register file.
- Three μops per clock cycle can be retired.

And this is in order retire section and micro operations are retired, so basic function is to store the data into the registers.

(Refer Slide Time: 58:15)



Pentium III

Pipeline
IFU1
IFU2
IFU3
DEC1
DEC2
RAT
ROB
EX
RET1
RET2
32 Byte
Instruction Streaming Buffer (1 Line)
Instruction Length Decoder
16 Byte
Decoder Alignment Stage
16 Byte
Decoder 0 (complex)
Decoder 1 (simple)
Decoder 2 (simple)
Decoded Instruction Queue (6 μOP Entries)
Micro Instruction Sequencer (MIS)
Register Allocation Table (RAT)
Reorder Buffer (40 Entries)
Reservation Station (20 Entries)
Store Data
Store Addr Unit
Load Addr Unit
Integer/ MMX ALU
FP Unit
Integer/ MMX ALU
Result Bus
Memory Order Buffer (MOB)
System Interface
64
256

This is internal diagram let us not go into the detail. So, I have covered the micro architecture of Pentium II and Pentium III and starting with your 8086.

Thank you.