**80386 Microprocessor**

Table of Contents

[Limitations of 80286 2](#_Toc83227155)

[Salient Features 3](#_Toc83227156)

[Versions 4](#_Toc83227157)

[Internal Block Diagram 5](#_Toc83227158)

[Bus Control Unit 6](#_Toc83227159)

[Register Organization 7](#_Toc83227160)

[Segment Descriptor Registers 7](#_Toc83227161)

[System Address Registers 7](#_Toc83227162)

[Control Registers 7](#_Toc83227163)

[Debug and Test Registers 8](#_Toc83227164)

[Flag Register 8](#_Toc83227165)

[Modes of Operation 9](#_Toc83227166)

## Limitations of 80286

The 80286 microprocessor was limited in several ways:

* It had a **16-bit ALU**, which was insufficient for new computers coming to the market, which demanded **32-bit** processing.
* Its **segment size** was limited to **64KB**.
* Its **virtual memory** was limited to **1GB**.
* It was difficult to **switch between real and protected modes**. The microprocessor needed to be **turned off** to do this.

The 80386 addressed each of these limitations:

* It provided a **32-bit ALU**, which also created a brand new data size, the **double word**.
* **Segments** could now be as large as **4GB**.
* Programs were allowed to have **16000 segments**, for a total of of **virtual memory**.
* A new mode, the **Virtual 86 Mode**, allowed switching between real and protected modes without turning off the microprocessor.

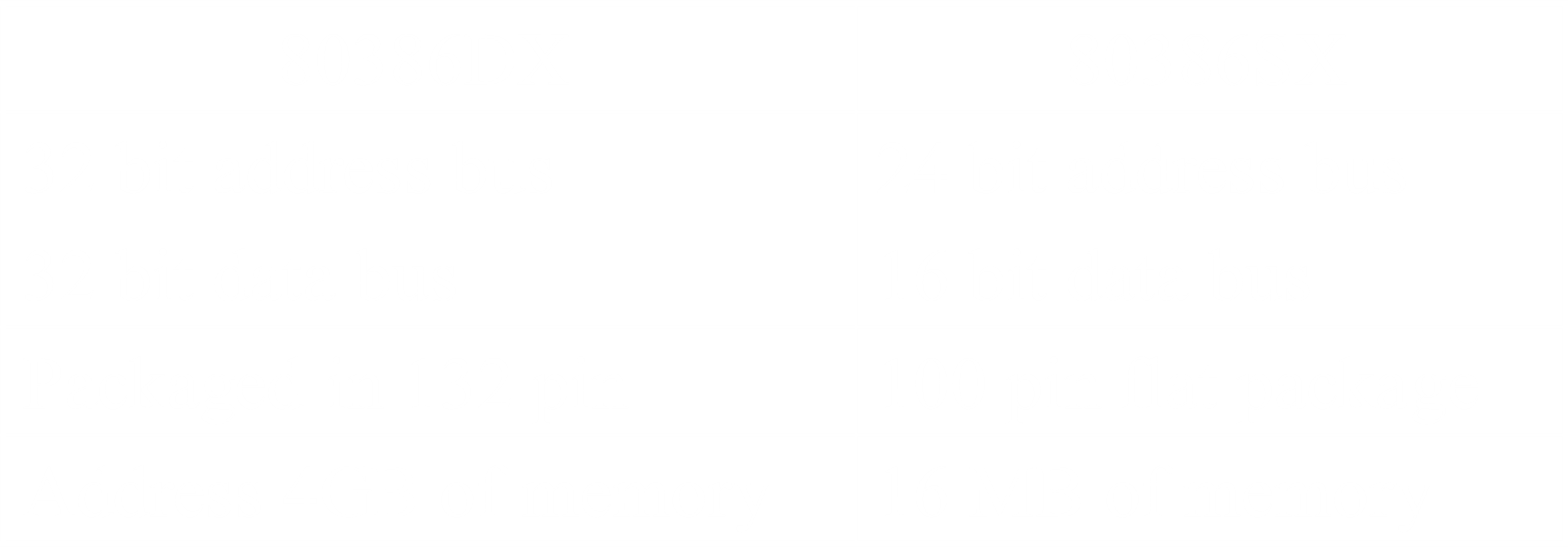
## Salient Features

* 275000 transistors
* 5 million instructions per second (MIPS)
* Clock speeds of 12 and 40MHz

## Versions

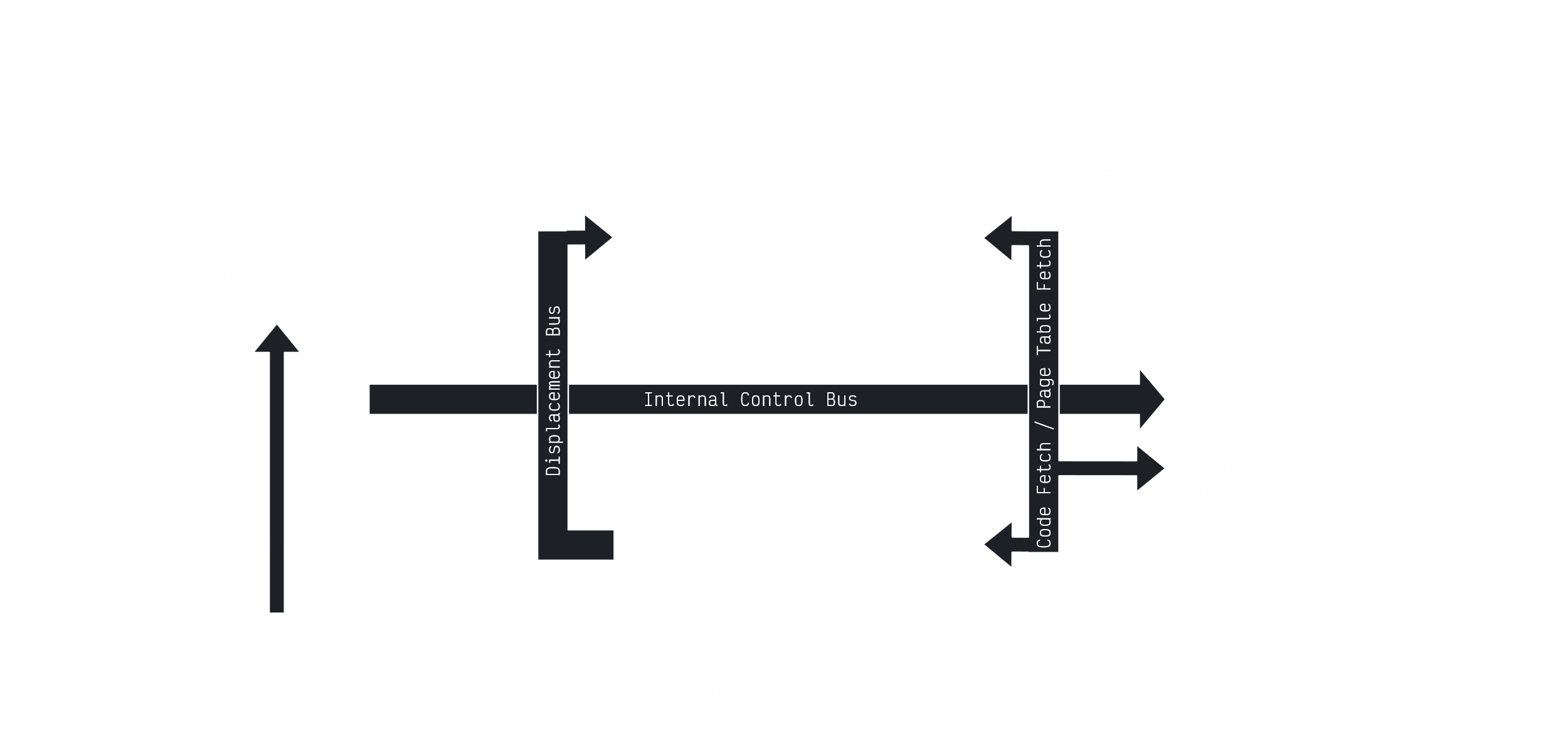
Two versions of the 80386 were commonly available, the original 80386, later renamed to **80386 DX**, and the **80386 SX**, which was a **low-cost alternative** for applications that did not require 32-bit processing. The chips were available in clock speeds of 16MHz, 20MHz, 25MHz and 33MHz.

It was found that many applications required less than 16MB of memory, which made the 80386 SX popular. It did not have a **math coprocessor**, but still featured the **32-bit architecture** and **built-in multitasking**.



Both versions had the **same internal architecture**. The **lower cost package** and ease of **interfacing** to 8-bit and 16-bit memory and peripherals made the SX version suitable for **low-cost systems**.

## Internal Block Diagram



The above diagram is divided into multiple sections.

The **Segmentation** unit and **Paging** unit deal with physical address derivation, which will be discussed separately later on.

The **Instruction Prefetch** unit prefetches instructions and can store up to **16-bytes** of instructions.

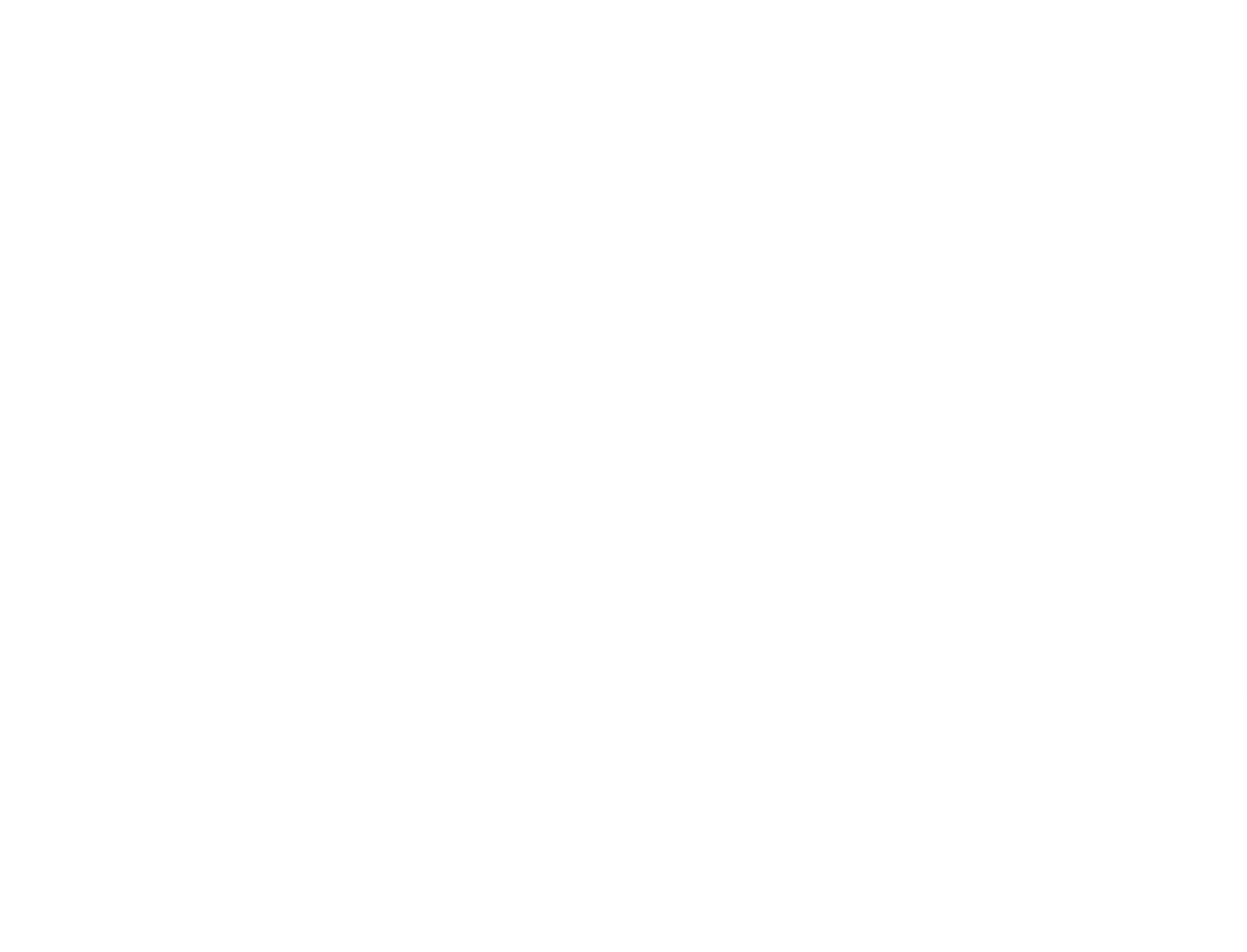
The **Instruction Pre-decode** section includes the Instruction Decoder and the ability to store 3 decoded instructions.

From there, decoded instructions go to the **Control Unit** and the **ALU**. The ALU includes **register files**.

The 80386 is **non-multiplexed**, since it has separate address and data buses, each of 32-bits.

### Bus Control Unit

The data bus and address bus are simple enough, but the address bus has some new features, specifically dealing with the **memory banks**. There are now **four** memory banks, BE0 to BE3.



Each bank is **1GB**, which means there is a total of 4GB of memory here. For **double words**, four consecutive memory locations are occupied.

From the address bus, the **A0 and A1** pins are used to identify one of the **banks**. The other pins, from **A2 to A31**, identify an address within the bank. Thus, A0 and A1 do not have address lines, since they are decoded internally.

## Register Organization

### Segment Descriptor Registers

In 80386, we have **six segment registers**, which are treated as **descriptor registers**. What those do is a part of the discussion on physical memory addressing. Each of these registers are of **73 bits**, which has a 32 bit base address, a 32 bit base limit and 9 bits of attributes. The base can use multiples of memory locations, and the number of memory locations is described by the base limit.

### System Address Registers

There are four new special registers called **System Address Registers** in the 80386, which are used to refer to each of the **descriptor tables**:

1. Global Descriptor Table (GDT)
2. Interrupt Descriptor Table (IDT)
3. Local Descriptor Table (LDT)
4. Task State Segment Descriptor Table (TSS)

### Control Registers

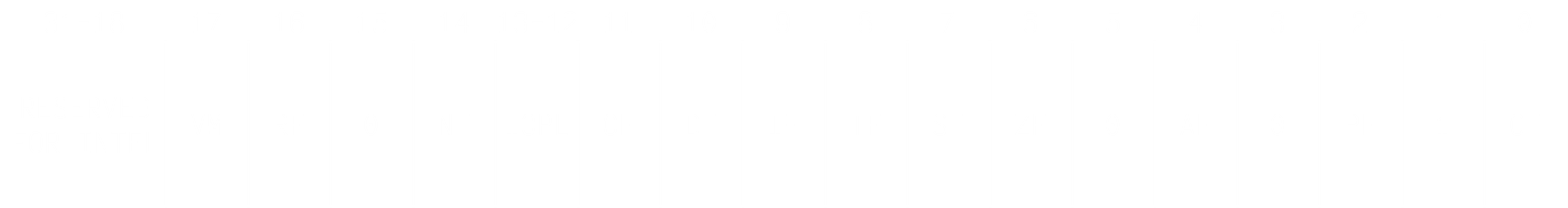
There are also three new **32 bit Control Registers**, CR0 through CR2. They hold the **Global Machine Status** independent of the executed task. **Load and store instructions** are available to access different registers through these control registers.

### Debug and Test Registers

These are also new registers, but their work is a little complicated, so they are being skipped.

### Flag Register

The **Flag register** is a 32 bit register. D18 to D31 are reserved, D15, D5 and D3 are set to 0 and D1 is set to 1. Note that we do not have **don’t care** values.



The only new flags are the **Virtual Memory Flag** and the **Resume Flag**. The Virtual Memory flag is set if the microprocessor is operating in Virtual Mode. The Resume Flag being set causes the microprocessor to ignore all debug faults. If it is not set, it will solve the debug faults first before continuing normal execution. By default, the flag is set to 1.

## Modes of Operation

The Real Mode and the Protected Mode of the 80386 are the same as that in 80286. The new mode is the **Virtual 8086 Mode**. This mode allows the execution of **real mode applications** while the processor is running in **protected mode**. For example, **interrupts** require the microprocessor to know the address of the stack, which requires the real mode.