

Arquitetura e Organização de Computadores II

Wolfenstein 3D



Grupo 1:

Arthur Trevizani Buback

Caio Eduardo Zanutelli Facini de Athayde

Gian Valério Zanatelli

Marcos Daniel Guasti Machado

Matheus Braga Cetrangolo

Capítulo 1 - Introdução

- Lançamento de Wolfenstein 3D
- Projeto liderado por John Carmack e John Romero
- Desenvolvido em uma época em que o PC era visto como uma "máquina impossível" para jogos de ação rápida



John Carmack



John Romero

Capítulo 1.1 - Introdução

Alguns obstáculos principais pareciam intransponíveis naquele período :

- O sistema de vídeo VGA não possuía double buffering, o que dificultava animações suaves sem artefatos visuais.
- Os processadores da época (CPUs) realizavam apenas operações de inteiros, enquanto cálculos 3D exigiam o processamento de frações (ponto flutuante).
- O dispositivo de áudio padrão era o PC Speaker, capaz de produzir apenas ondas quadradas simples ("beeps").
- A memória RAM era limitada a 1MB e o endereçamento não era plano, mas segmentado

Capítulo 1.2 – Revisão Bibliográfica

- O desempenho de um computador era determinado quase exclusivamente por sua CPU.
- Os computadores eram identificados pelo chip que possuíam (O Intel 80386 era o "386" e o Intel 80286 era o "286").
- Limitação de Ponto Flutuante: A maioria das máquinas domésticas não possuía uma FPU (Floating Point Unit).
- Aritmética de Inteiros: Todo o cálculo de renderização precisava ser adaptado para evitar o uso de decimais.
- A comunicação entre a CPU e a memória de vídeo (VRAM) era um gargalo constante.

Capítulo 1.3 – Metodologia

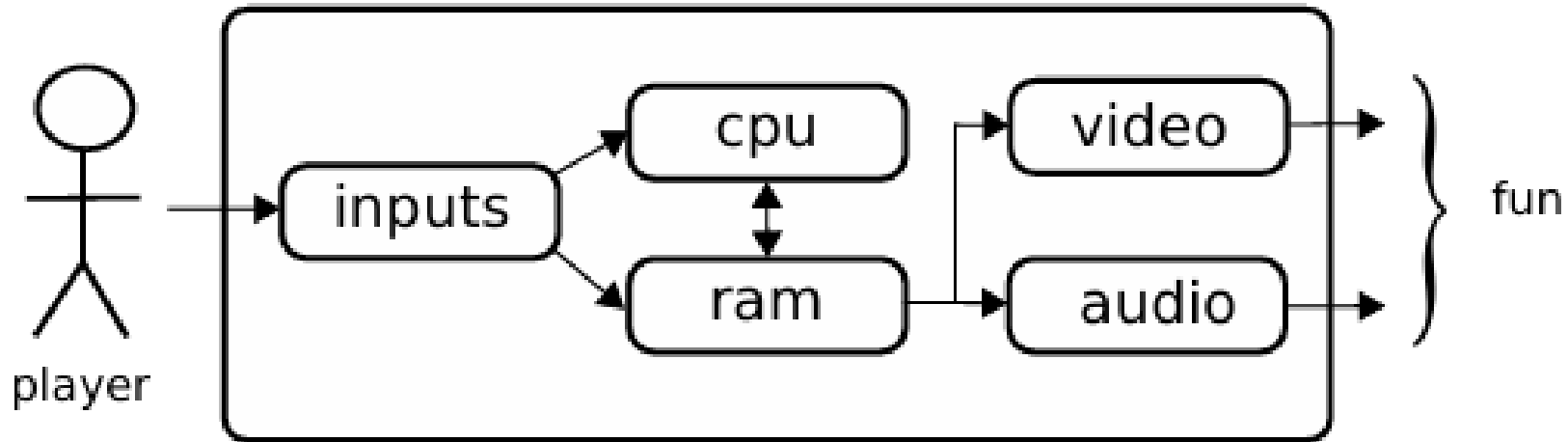
- Solução: implementação da técnica de Ray-casting
- Ray-casting utiliza uma grade 2D para projetar raios e determinar a distância das paredes, desenhando-as coluna por coluna
- John Carmack optou pelo ray-casting pelo custo computacional aceitável para o hardware disponível e entregava uma taxa de quadros (framerate) consistente.
- Otimização: O uso de "scalers" compilados e truques de escrita em múltiplas colunas no VGA permitiram que o jogo rodasse de forma fluida mesmo em máquinas modestas.

Capítulo 1.4 – Conclusão

Wolfenstein 3D não foi apenas um sucesso técnico, mas também comercial, gerando dez vezes mais receita do que os títulos anteriores da série Commander Keen. O legado do jogo perdura não apenas pela inovação no gênero FPS, mas pela demonstração de que restrições severas de hardware podem ser superadas com algoritmos criativos e eficientes. A abertura do código-fonte em 1995 permitiu que gerações de programadores estudassem essa "arqueologia digital" e compreendessem como a "máquina impossível" foi finalmente domada.

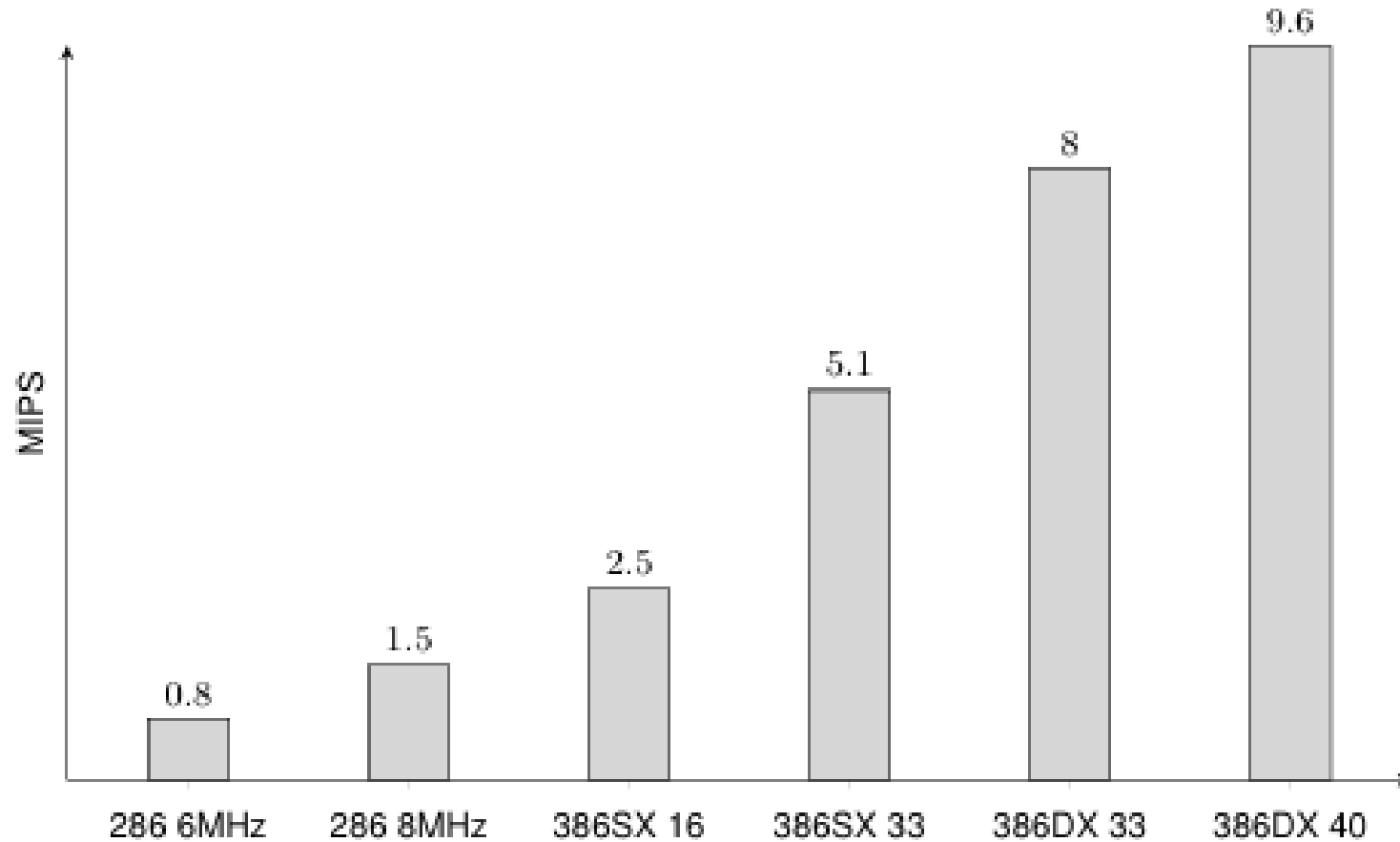
Capítulo 2 - Hardware

Pipeline do Hardware:



Etapa	Qualidade
RAM	Suportável
Vídeo	Impossível
Áudio	Muito ruim
Entradas	Ok
CPU	Impossível

Capítulo 2.1 – CPU: Central Processing Unit



Comparação de CPUs com MIPS

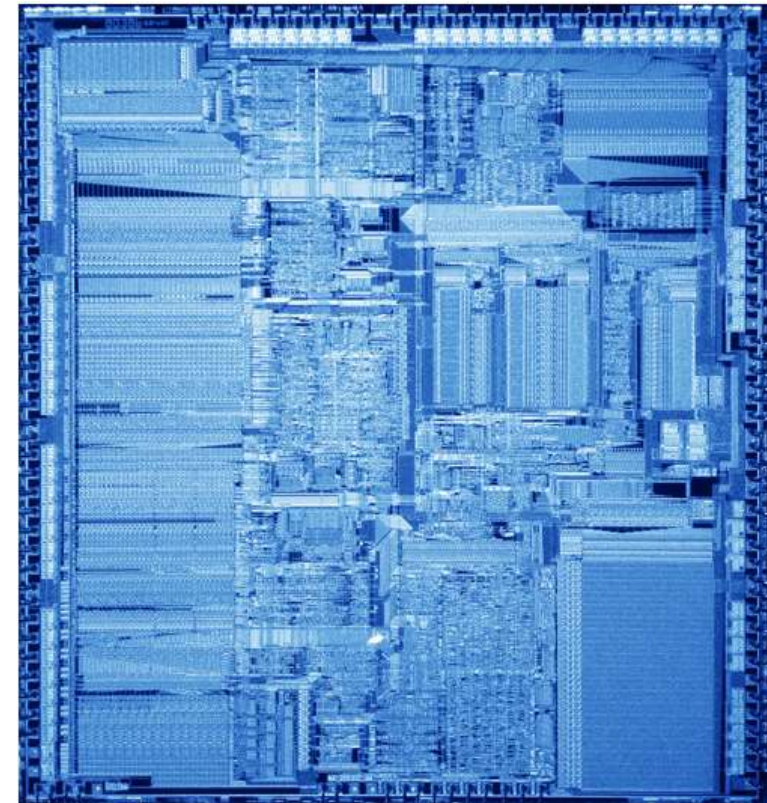
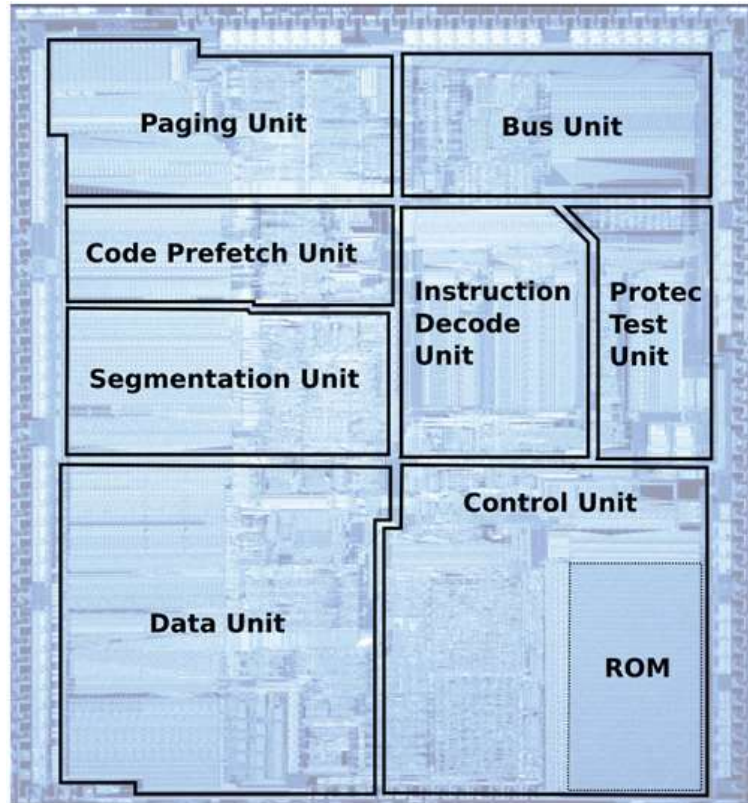
Capítulo - 2.1.2 - O Intel 80386

Origem:

- **Projeto Paralelo:** Começou como um projeto secundário em San Jose.
- **O Fracasso do i960:** Enquanto o time principal falhava com o chip i960 em Portland, o 386 assumiu o protagonismo.

Pilares do Sucesso:

- **Retrocompatibilidade:** Decisão estratégica de manter o conjunto de instruções do 286.
- **Modo 32 Bits:** Solução definitiva para os problemas de endereçamento de memória das gerações anteriores.
- **Escuta Ativa:** Designers ouviram o *feedback* dos programadores em vez de impor um novo set de instruções.



Layout do chip Intel 80386-DX de 16 MHz

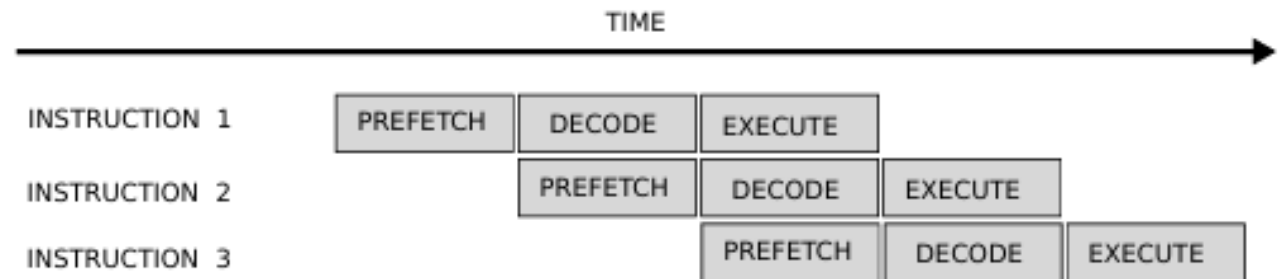
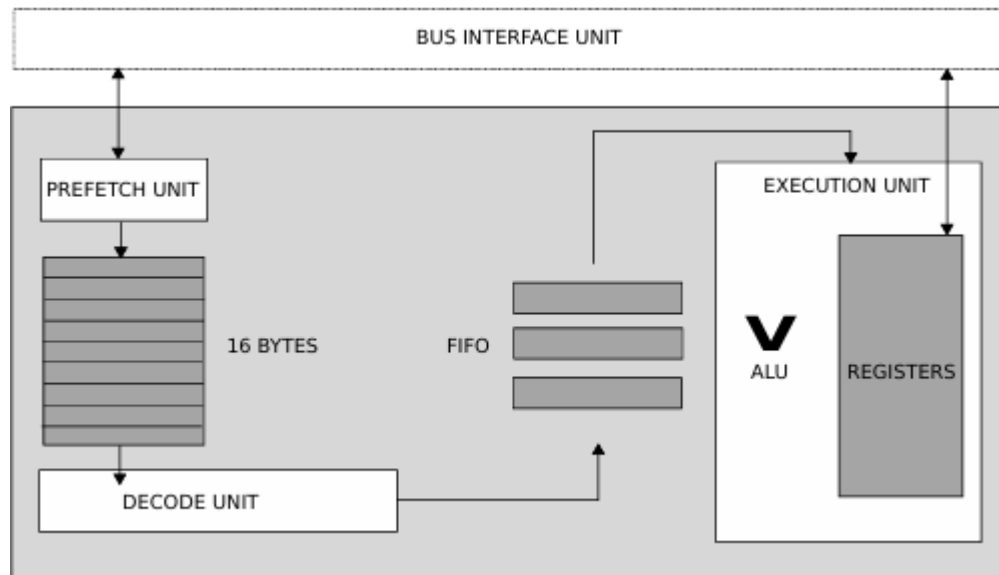
- Pipeline de 3 Estágios

O processamento é dividido em três etapas sequenciais:

- 1. Busca (Prefetch):** Busca instruções na memória e as coloca em uma fila de 16 bytes.
- 2. Decodificação (Decode):** Traduz a instrução e armazena o resultado em uma FIFO de 3 elementos.
- 3. Execução (Execute):** Onde a ULA e os Registradores processam a tarefa final.

- Gargalo do JMP

- **Busca Linear:** O Prefetch assume que o código sempre segue em linha reta.
- **Sem Predição:** O 386 não sabia prever para onde um desvio iria.
- **Pipeline Flush:** Uma instrução de salto (**JMP**) forçava a limpeza total do pipeline, jogando fora tudo o que já havia sido buscado e decodificado. Isso causava uma perda imediata de ciclos de clock.



Registradores uso geral de 32 bits: EAX, EBX, ECX, EDX.

•**Memória:** Barramento de endereços de 32 bits, permitindo até **4GB de RAM**.

•**Lentidão Nativa:** Mesmo com pipeline, o 386 levava no mínimo **2 ciclos por instrução**.

Causa: Falta de cache SRAM no chip e unidade de decodificação lenta.

Capítulo - 2.1.3 - Ponto Flutuante

- **Necessidade:** Cálculos 3D exigem frações para precisão.
- **O Padrão IEEE 754:** 1 bit de sinal, 8 bits de janela (expoente) e 23 bits de offset (mantissa).



- **O Grande Problema:** O 386 não possuía FPU (Unidade de Ponto Flutuante) integrada.
- **Emulação via Software:** Sem o coprocessador caro (i387), o compilador simulava o float via software..
Resultado: Processamento lento demais para um jogo em tempo real.
- **Inteiros:** Eram rápidos, mas não tinham precisão para profundidade 3D.
- **Floats:** Tinham a precisão necessária, mas eram impossivelmente lentos.

Capítulo - 2.1.4 - Conclusão

O hardware de 1991 não estava pronto para o 3D.

Wolfenstein 3D só existiu porque os programadores "hackearam" essa limitação usando a matemática de Ponto Fixo.