



Universidade de Brasília

Departamento de Ciência da Computação

Organização e Arquitetura de Computadores

Prof. Marcus Vinicius Lamar



Universidade de Brasília
Departamento de Ciência da Computação

Aula 0

Apresentação e Motivação



Perguntas que serão respondidas no curso:

- Como os programas escritos em linguagem de alto nível (C, Java, etc), são traduzidos em linguagem do processador e como o processador os executa?
- Qual é a interface entre software e hardware e como o software diz ao hardware o que fazer?
- O que determina o desempenho de um programa e como o programador pode melhorá-lo?
- Quais técnicas são usadas pelos projetistas de hardware para aumentar o desempenho e a eficiência energética?
- Quais são as razões e as consequências da troca do paradigma de processamento sequencial para o processamento paralelo?

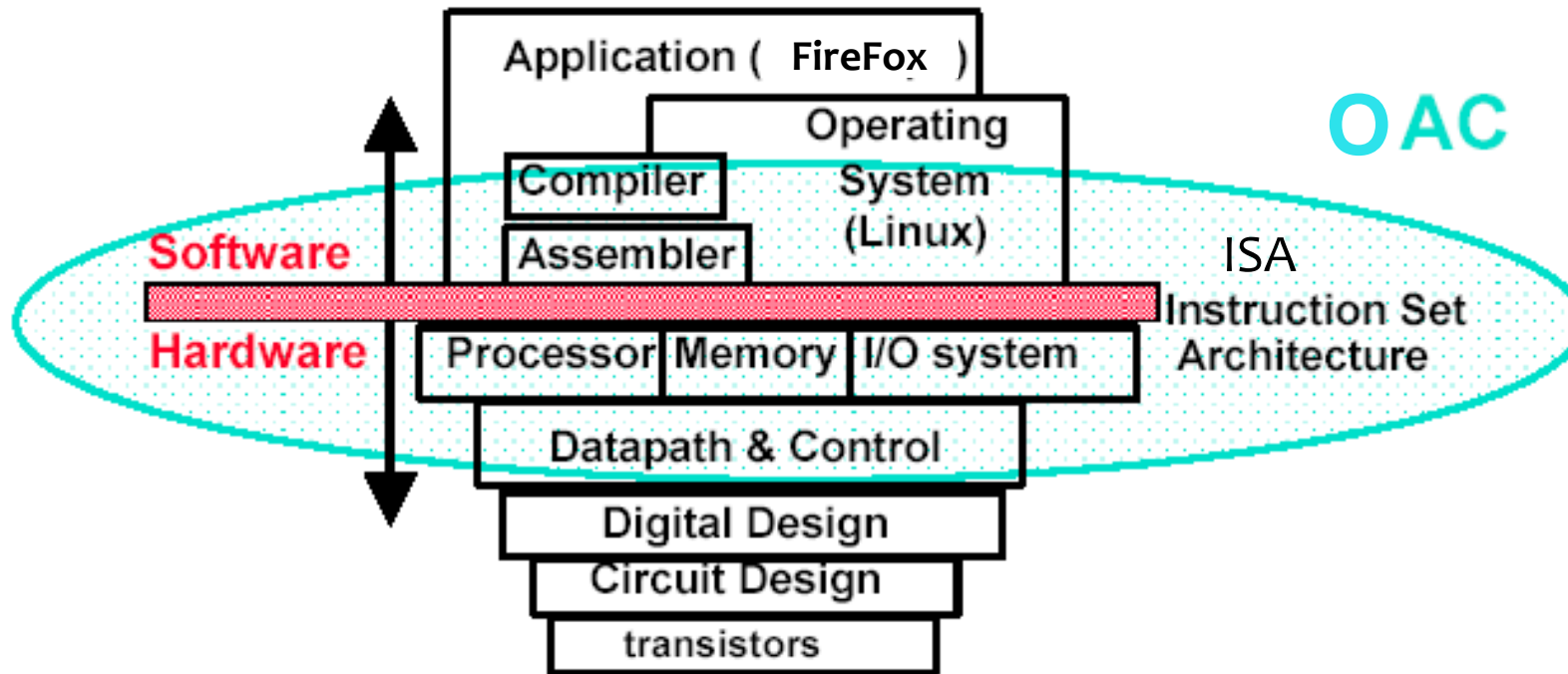


Por que aprender esse assunto?

- Você deseja entender como os modernos processadores funcionam 🤖
- Você deseja criar seu próprio processador 🧐
- Você deseja desenvolver softwares eficientes 😊
- Você precisa tomar uma decisão em relação a uma compra ou prestar consultoria 💰
- Está no currículo como disciplina obrigatória 😁



O que é: Organização e Arquitetura de Computadores?



Arquitetura do conjunto de instruções
+
Organização da máquina



Introdução

■ Computadores:

- **Diferentes tipos:** Servidores, desktops, notebooks, tablets, smartphones.
- **Diferentes usos:** escritório, design gráfico, bancos de dados, computação científica (simulação), redes sociais, CP2077, CS, games, games,
- **Diferentes fabricantes:** HP, IBM, Dell, ASUS, Sun, Apple, Samsung ,...
- **Diferentes processadores:** Intel, AMD, IBM, Motorola, HP, Sun, MIPS, ARM, Apple (IBM, Intel, ARM(Samsung, TSMC)), ...
- **Diferentes tecnologias subjacentes :** SSD, HD, Placa-mãe, RAM, GPU,

Assim: diferentes custos e diferentes desempenhos!

■ Melhor maneira de aprender:

- Concentrar em um exemplo específico, aprender como ele funciona e generalizar, exemplificando os conceitos em dispositivos modernos.

Ciência × Tecnologia



Principais Classes de Sistemas Computacionais

- Servidores
- Pessoais
- Embarcados

Principais Classes de Sistemas Computacionais

■ Servidores

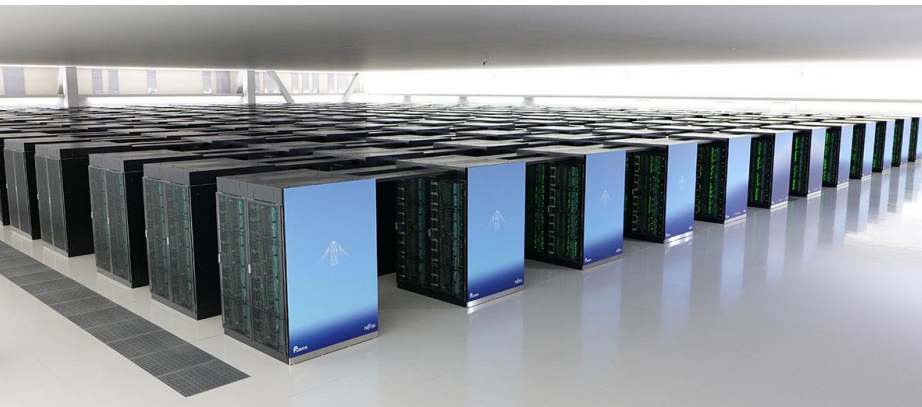
- Recursos compartilhados entre vários usuários
- Geralmente sistemas de software específicos
- Ex.: Desde simples servidores de arquivo, webserver, servidores em nuvem, supercomputadores
- Alta dependabilidade (confiabilidade, segurança, disponibilidade e manutenibilidade), geralmente alto custo.



■ Pessoais

■ Embarcados

<https://www.google.com/about/datacenters/inside/streetview/>



FaceBook

Fugaku (7.630.848 cores ARM)



Principais Classes de Sistemas Computacionais

■ Servidores

■ Pessoais

- Recursos utilizados geralmente por um único usuário
- Geralmente programas de terceiros
- Ex.: Desktops, notebooks, tablets, smartphones, etc.
- Compromisso entre custo e desempenho para o usuário

■ Embarcados





Principais Classes de Sistemas Computacionais

- Servidores
- Pessoais
- Embarcados



Casa inteligente

- Recursos projetados para fins específicos
- Software de difícil customização, geralmente integrado ao hardware.
- Ex.: Eletroeletrônicos (TV, DVD, Conversores, eletrodomésticos,...), Automóveis/Barcos/Aviões, Industriais, Brinquedos, Robôs, IoT.
- Geralmente baixo custo e baixa dependabilidade, embora alguns precisem de baixa taxas de falhas (sistemas redundantes).



Máquina de lavar



Aeronaves



Veículos autônomos



Era Pós-PC

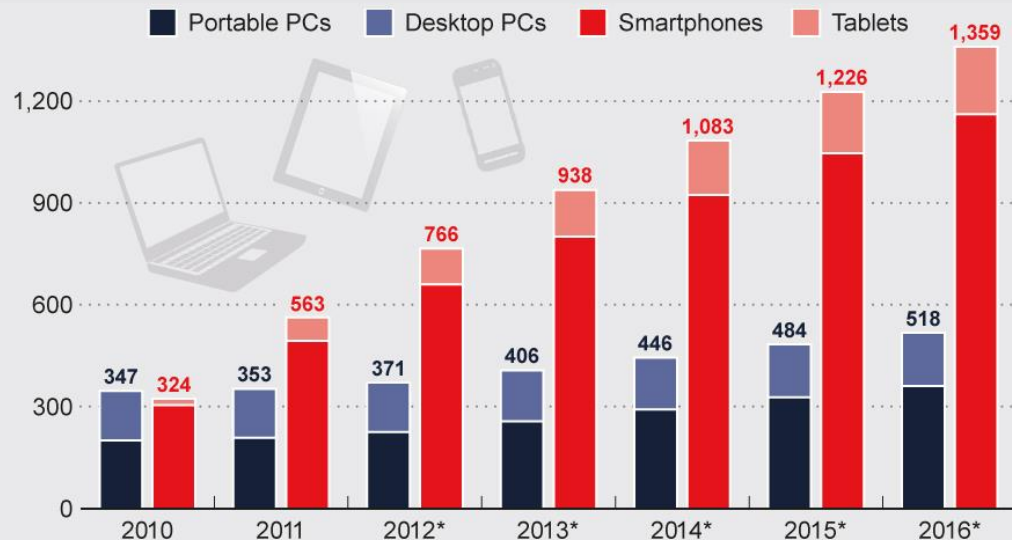
1940 - 1970: *Criação*. Grandes computadores (ENIAC)

1970 - 2000: *Popularização*. Computadores pessoais (PCs)

2000 - hoje: *Individualização*. Dispositivos portáteis pessoais (celular), embarcados (TV), internet das coisas (IoT), computação vestível, computação em nuvem

The Post-PC Era Has Arrived

Global smartphone, tablet and PC shipments (in millions)



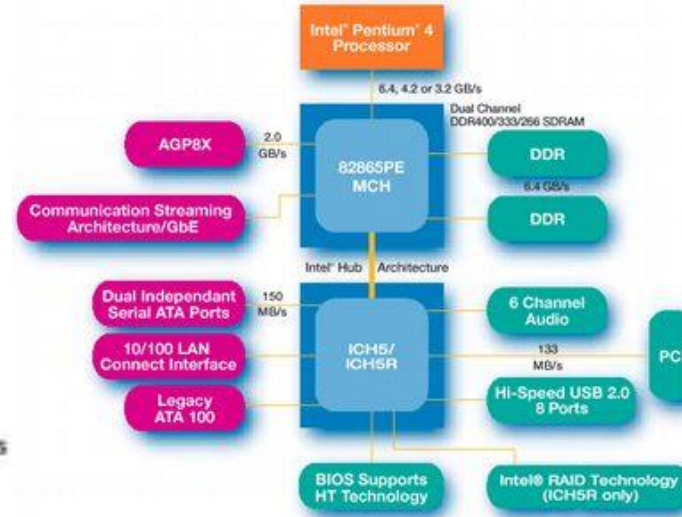
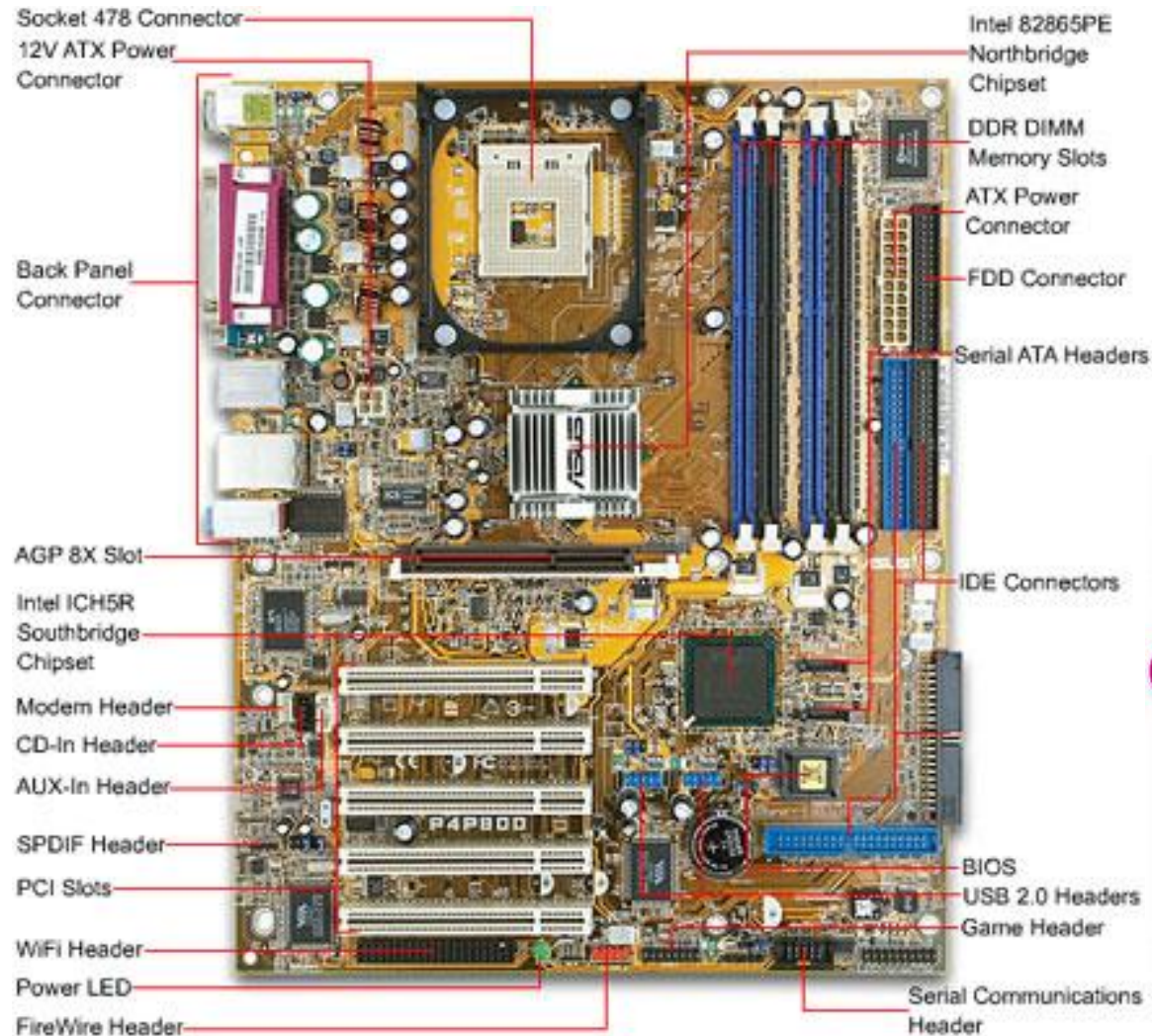


As oito grandes ideias (do Patterson) na Arquitetura e Organização de Computadores

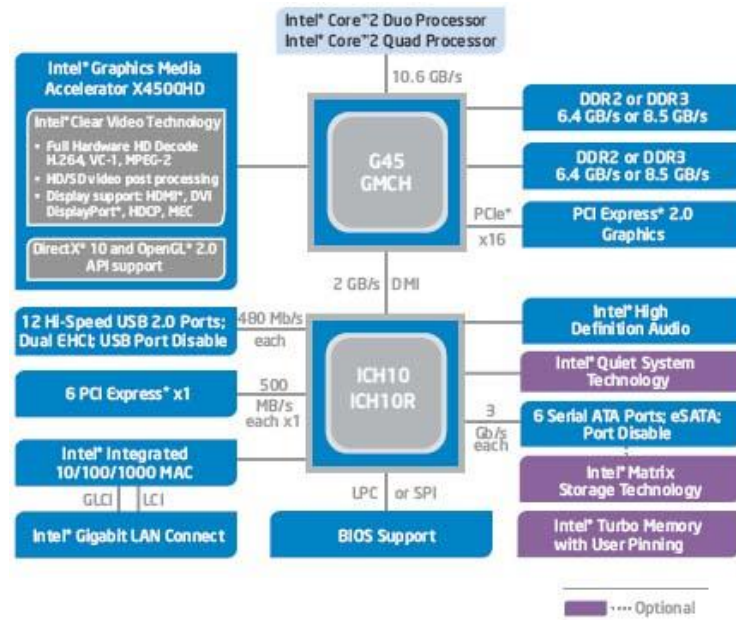
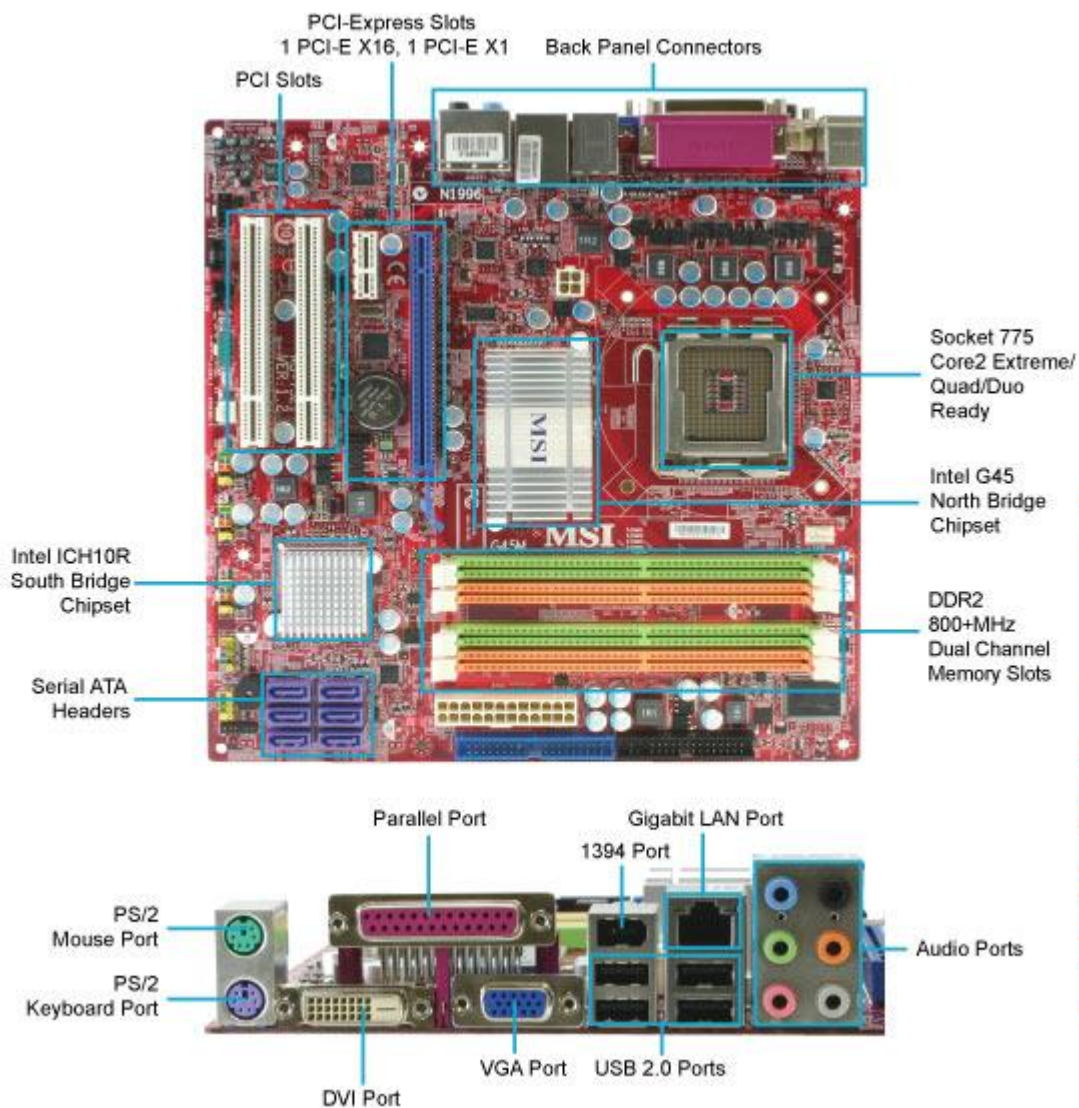
- Projetos considerando a Lei de Moore
- Uso da abstração para simplificar os projetos
- Tornar o caso comum rápido
- Aumentar o desempenho via paralelismo
- Aumentar o desempenho via pipeline
- Aumentar o desempenho via predição
- Hierarquia da memória
- Dependabilidade via redundância



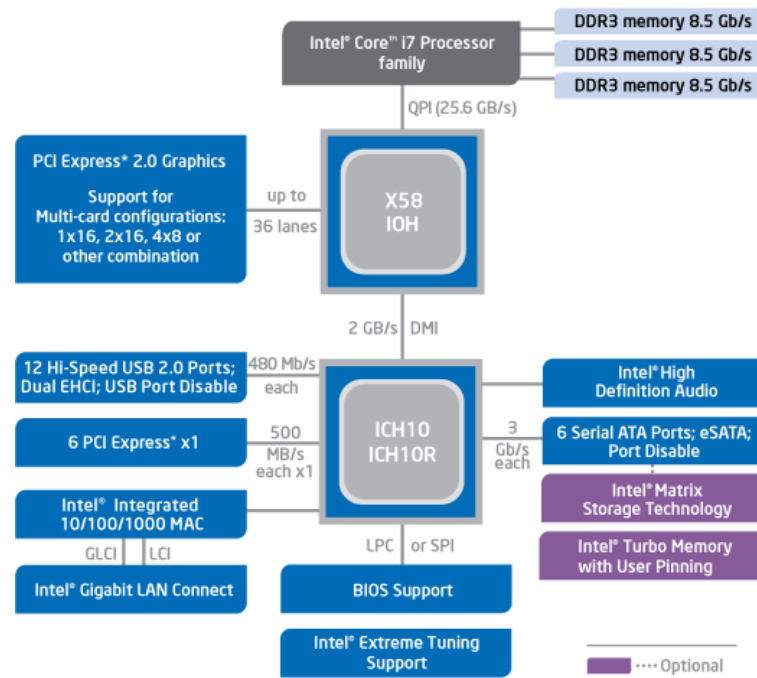
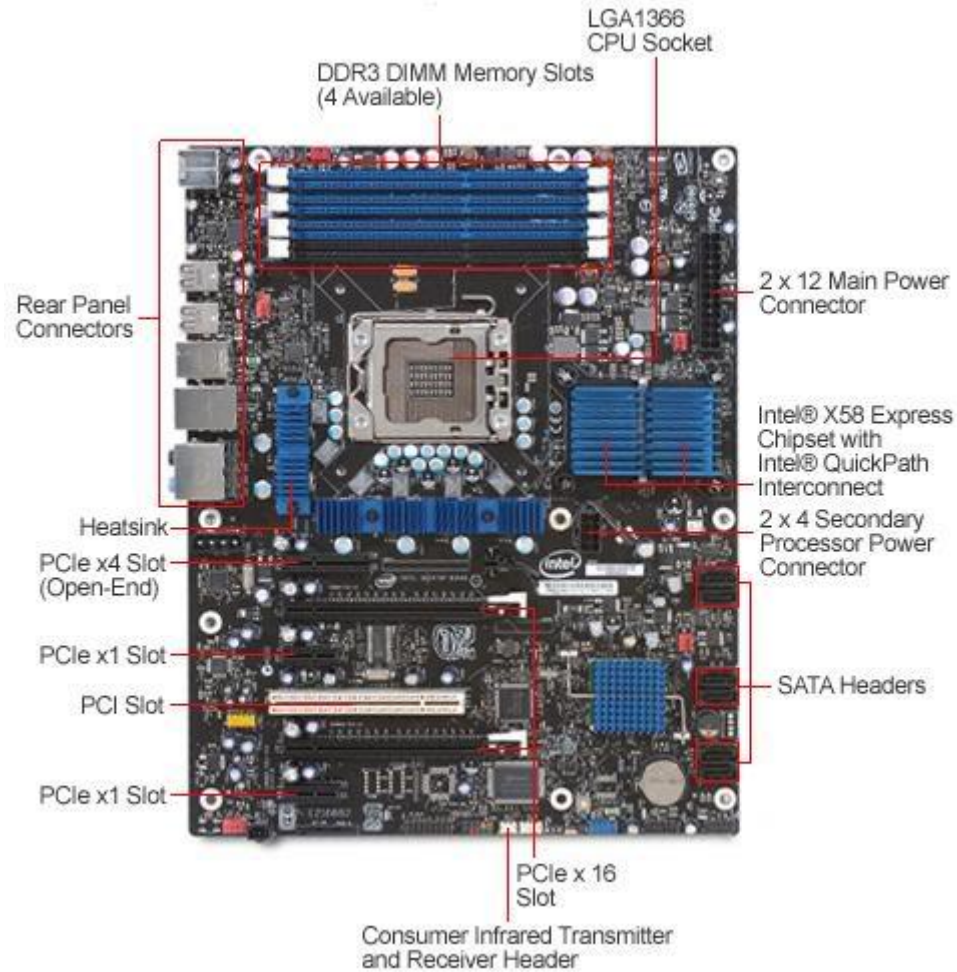
Placa mãe para Pentium IV



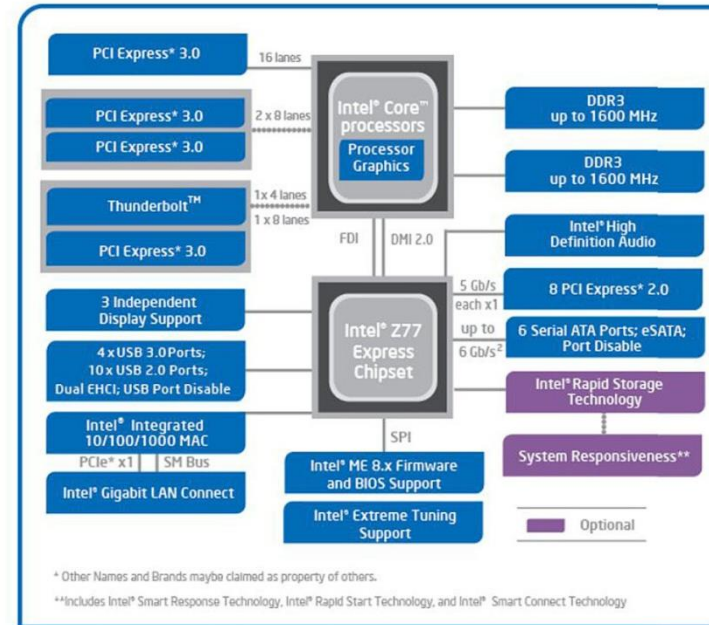
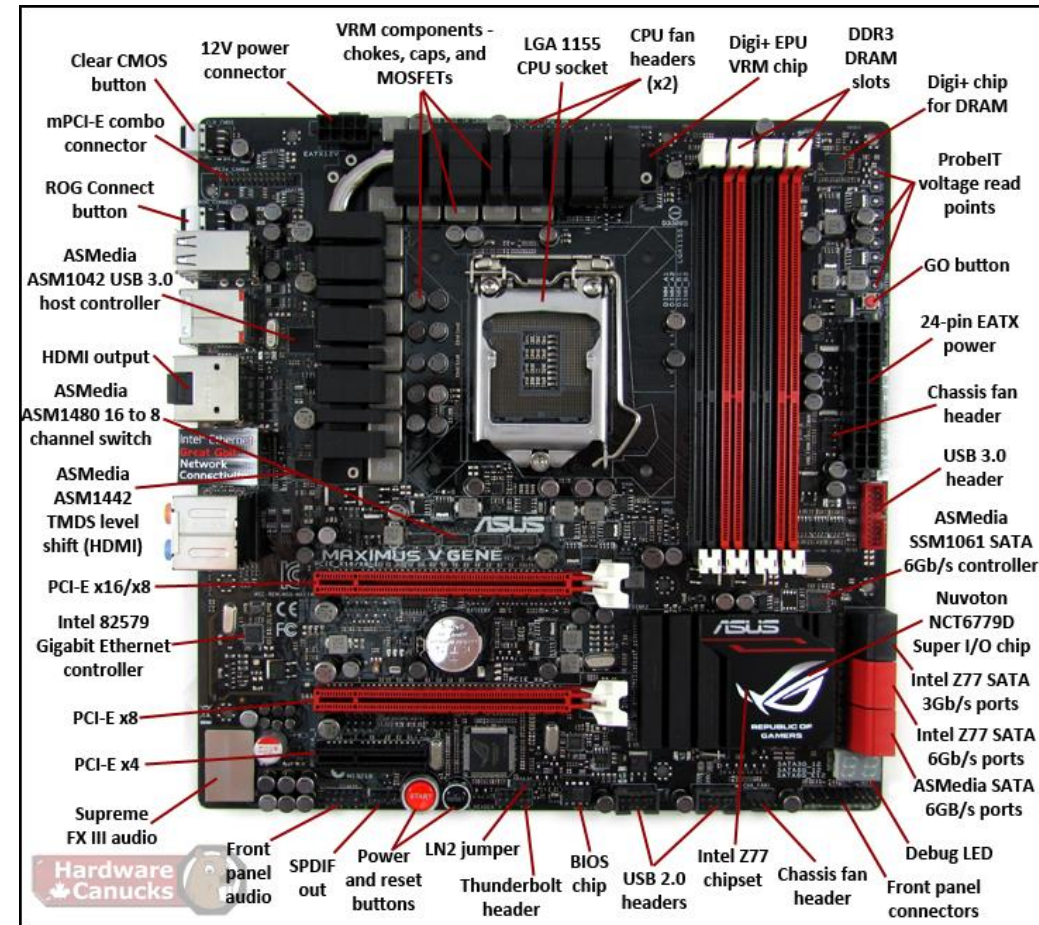
Placa mãe para Core2



Placa mãe para Core i7

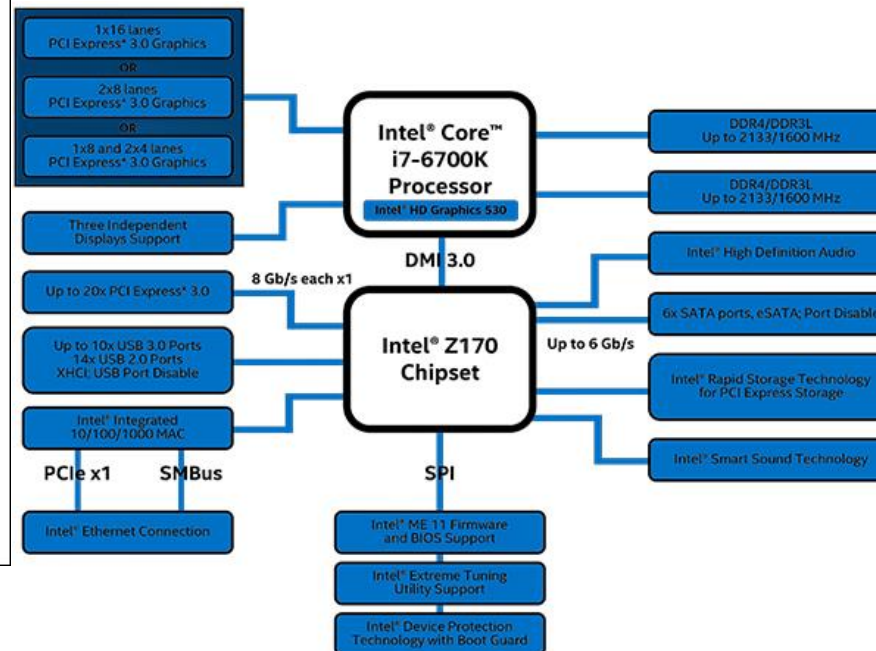
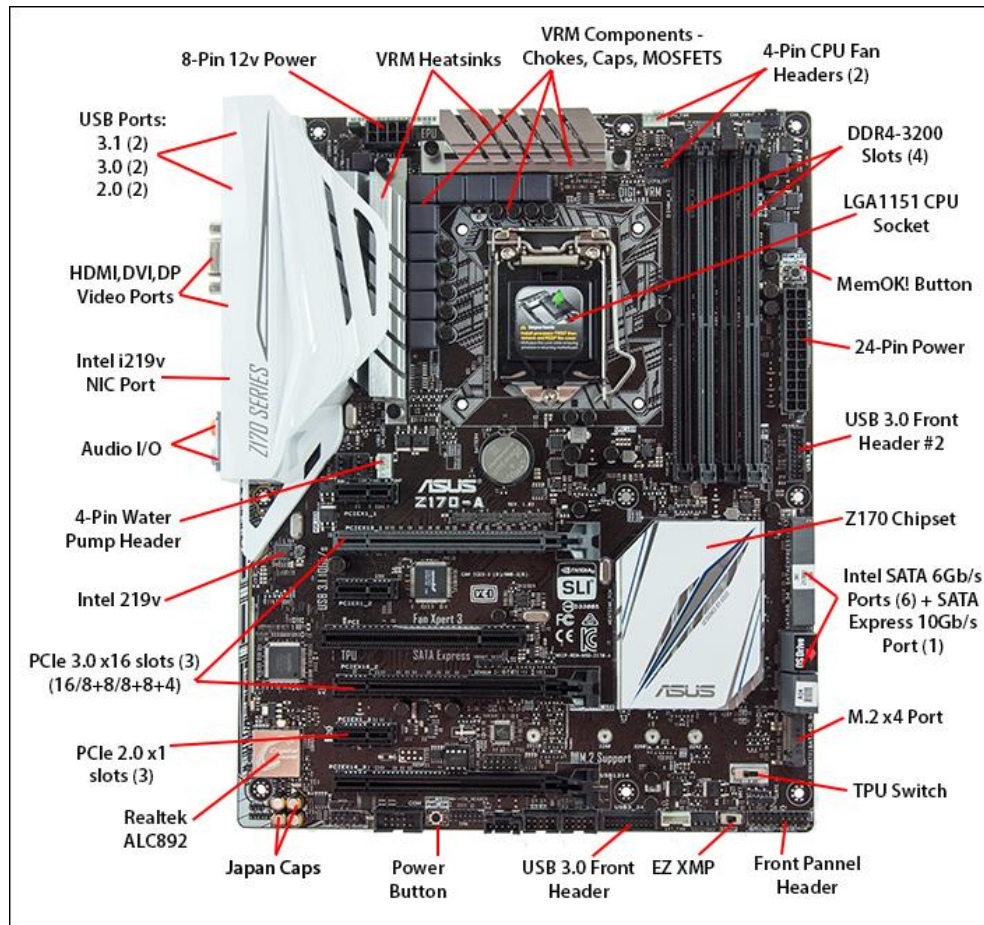


Placa mãe para Core i7 de 3ª Geração

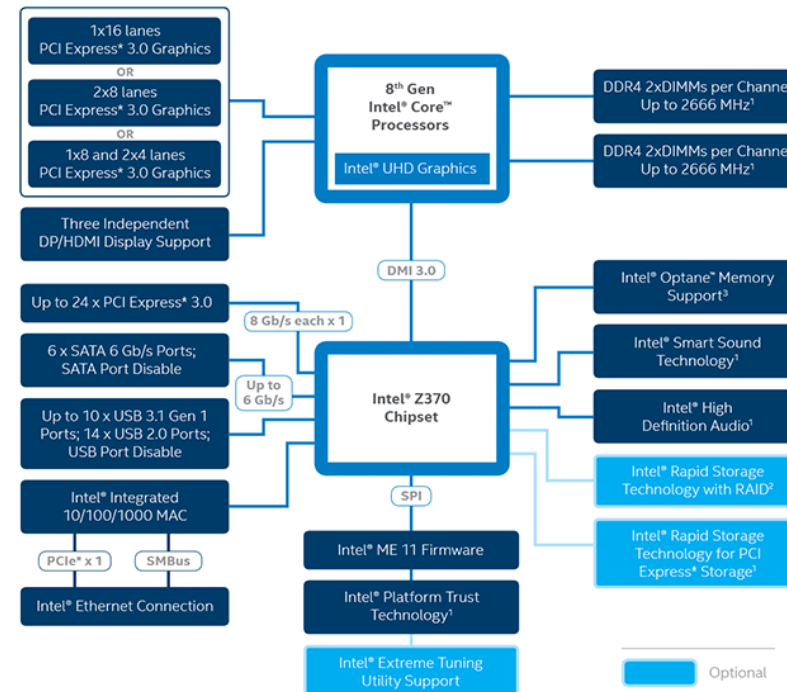
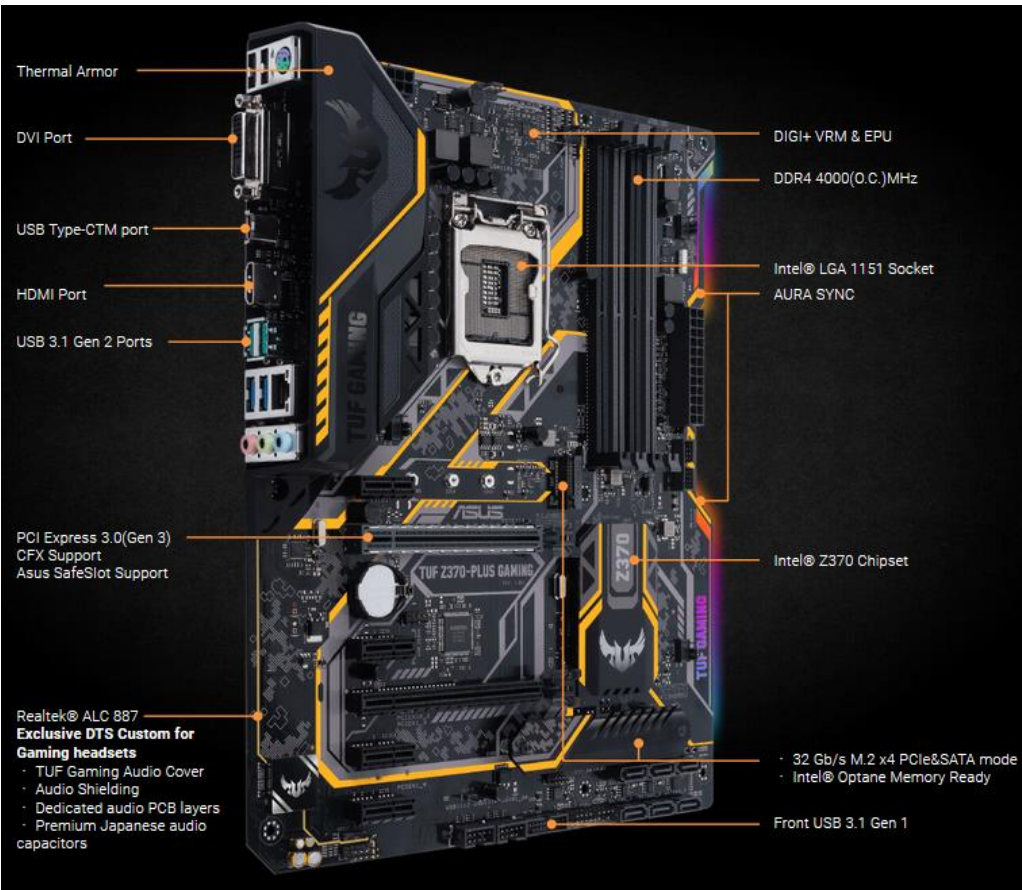


Intel® Z77 Express Chipset Platform Block Diagram

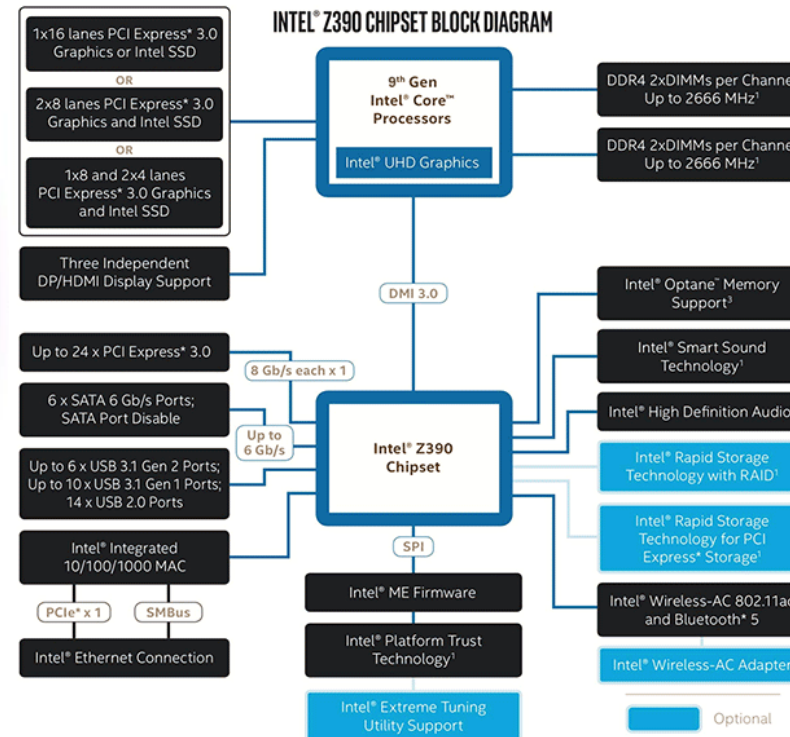
Placa mãe para Core i7 6ª geração



Placa mãe para Core i9 8ª geração



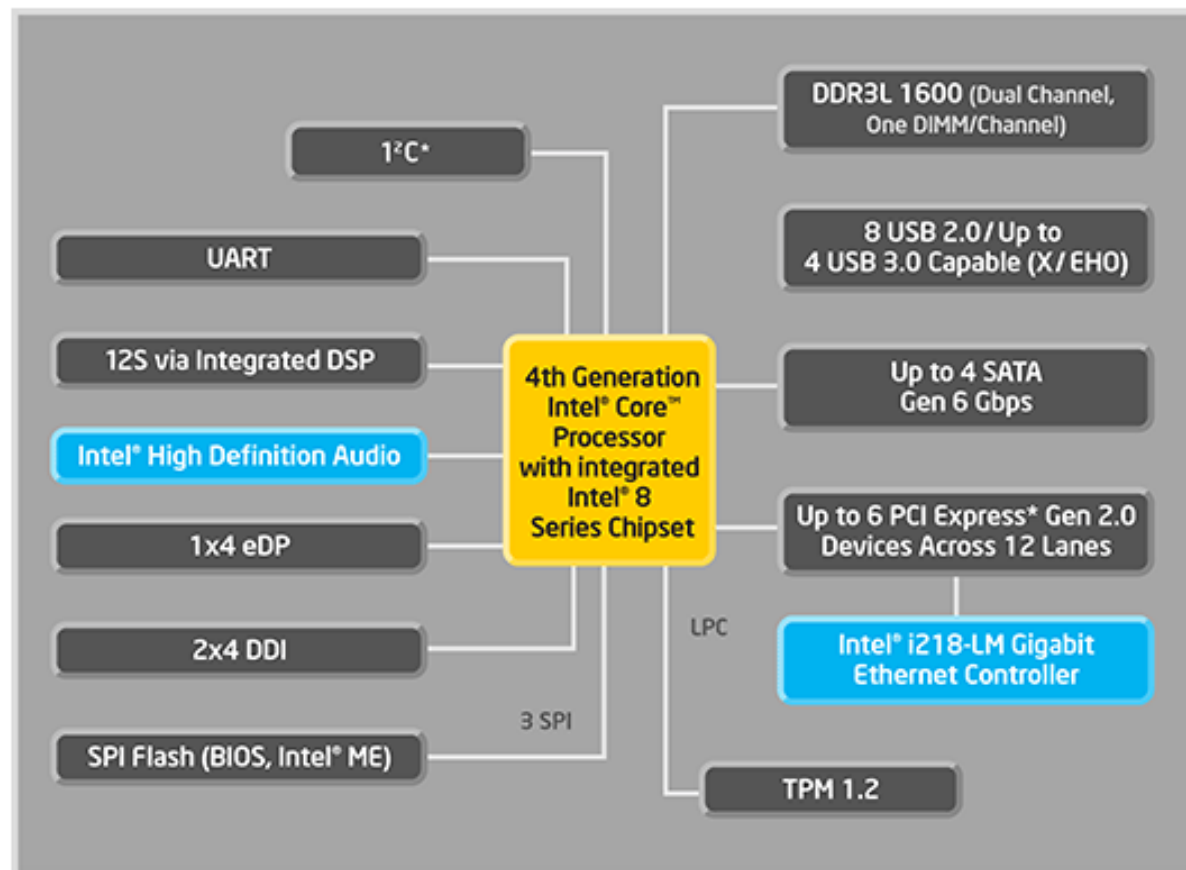
Placa mãe para Core i9 9ª geração





SoC – System on Chip

Chipset integrado

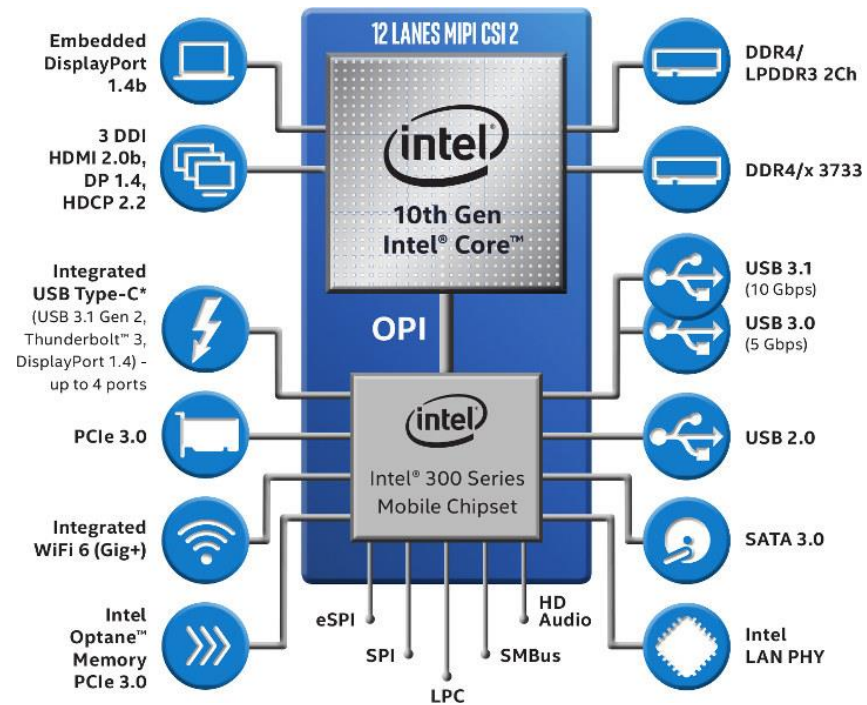


Usado em portáteis e embarcados



Placa mãe para Core i9 10ª geração

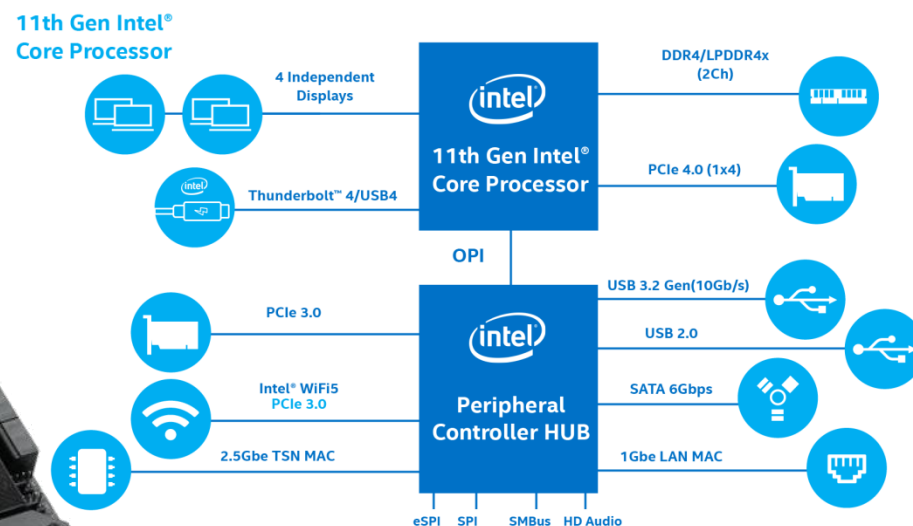
Comet Lake
Ice Lake



Placa mãe para Core i7 11ª geração

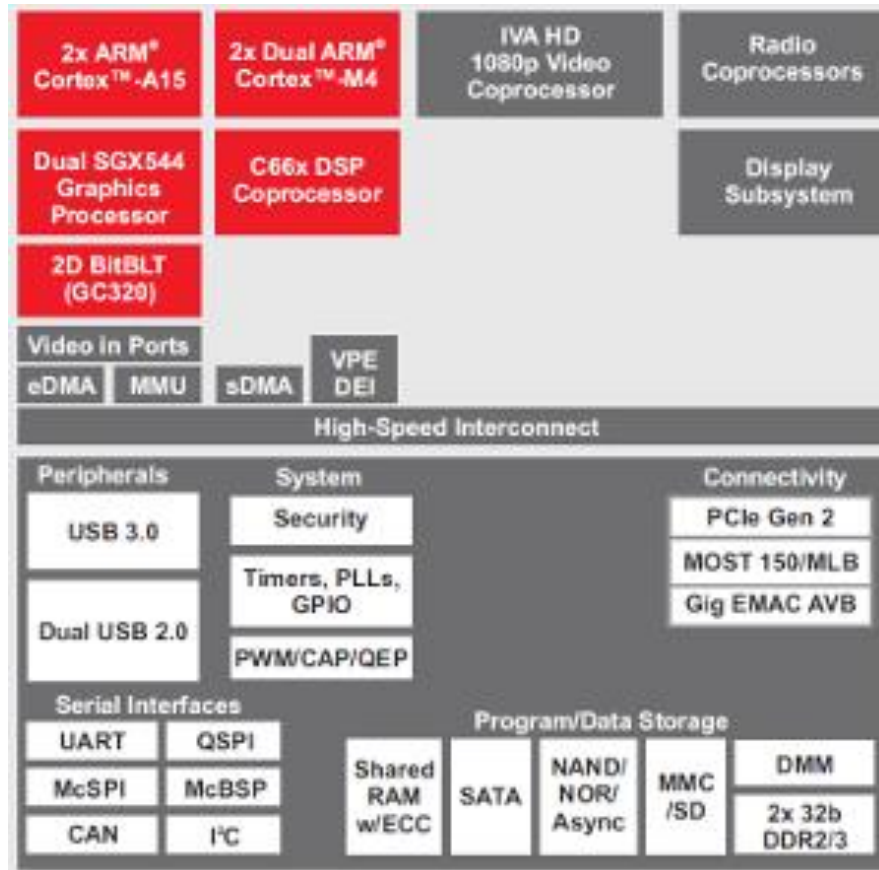
Z490 – mesma do 10ª geração

Tiger Lake





Processamento Heterogêneo: CPU+ μ CPU+GPU+DSP+ASIC+FPGA+Interfaces



Ex.: Smartphones \Rightarrow Arm big.LITTLE