

Instalação e Configuração de Servidores

Prof. Diego Cirilo

Aula 03: O processo de boot

O termo

- *To pull oneself up by one's bootstraps*



O processo de *boot*

- É o processo que compreende o momento da energização do processador ao momento em que o computador se torna utilizável pelo usuário;
- É a auto-inicialização dos dispositivos de *hardware* e componentes de *software*;

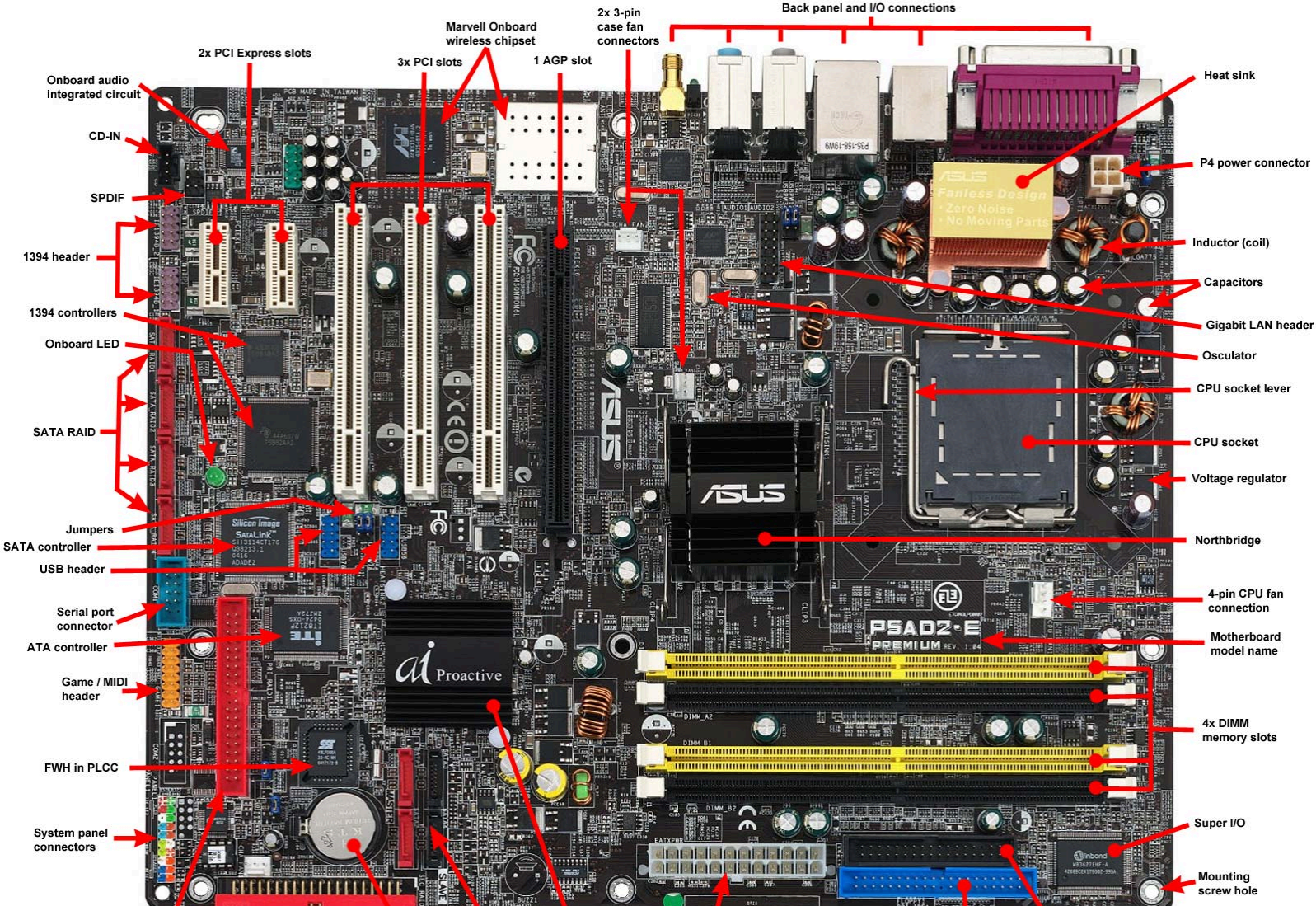
BIOS

- *Basic Input/Output System*;
- *Software* armazenado em uma memória ROM (atualmente EEPROM/*Flash*);
- Responsável pelos testes de inicialização, controle de *hardware* em baixo nível e pela chamada do sistema operacional;
- Um tipo de *firmware*;
- Hoje é *legacy* devido ao UEFI.

UEFI

- *Unified Extensible Firmware Interface*;
- Um *firmware*, assim como o BIOS;
- Mais funcionalidades como GUI, partições GPT, rede, traduções, etc;
- Disponibiliza um *shell* básico;
- Infelizmente mais controle para bloqueio de software;
- *Secure boot*

BIOS



BIOS

- Assim que a fonte de alimentação energiza o processador, ele busca a primeira instrução para execução;
- O endereço dessa instrução é exatamente o endereço do BIOS;
- A primeira rotina do BIOS é a execução do POST.

POST

- *Power On Self Test*;
- Uma rotina executada pelo BIOS para verificação dos dispositivos de *hardware*;
- Verifica a quantidade de memória e o estado de funcionamento da mesma;
- Verifica a existência de unidades de armazenamento;
- Também verifica demais dispositivos de *hardware*;
- Retorna códigos de erro através de *beeps*.

CMOS

- *Complementary Metal-oxide-semiconductor*;
- Termo histórico, já que a grande maioria dos dispositivos atualmente usam a tecnologia CMOS;
- Se refere a uma memória volátil que armazena os dados de configuração e informações sobre o *hardware* instalado;
- Porque volátil?
- RTC + Armazenamento;
- Atualmente está incluso no *chipset*.

Master Boot Record

- O BIOS busca o disco que armazena o sistema operacional;
- O MBR é executado;
- Espaço em disco responsável por armazenar informações de como as partições, contendo sistemas de arquivos, estão organizadas na mídia em questão.
- Além disso a MBR contém código executável com funções para inicialização do sistema operacional.
- Possui no mínimo uma tabela de partições e o código de chamada do sistema operacional;

UEFI

- Na UEFI não é necessário usar a MBR;
- O próprio UEFI é capaz de montar os discos;
- Dentro do disco ela busca os arquivos de inicialização do sistema.

Inicialização do S.O.

- O MBR executa as funções de inicialização do sistema operacional;
- A rotina de inicialização do S.O. acessa o BIOS para obter informação dos dispositivos ligados no sistema;
- Se há mudanças desde o último boot tenta instalar os novos drivers;
- Cria os processos de *background*;
- Inicia a interface com o usuário.

Boot do Windows

- Após o boot do hardware;
- O primeiro passo do Windows é o BootMgr;
- Testa se o sistema foi hibernado ou estava em estado de *stand-by*;
- Se estava, utiliza o WinResume.exe;
- Senão utiliza o WinLoad.exe.

Boot do Windows

- Componentes de software do boot:
- Kernel
 - ntoskrnl.exe
- HAL
 - hal.dll
- SYSTEM hive
 - Informações do registro
- Drivers de boot
 - Win32k.sys

Boot do Windows

- Inicializar o kernel
- Inicializar o HAL
- Carregar todos os drivers
- Iniciar o primeiro processo do usuário:
 - smss.exe
- Alguns outros modos de inicialização são:
 - Modo de recuperação;
 - Modo seguro

Boot do Linux

- A MBR executará o *bootloader*, também conhecido como gerenciador de boot;
 - GRUB;
 - LILO;
- O bootloader então carrega o kernel do S.O.

Boot do Linux

- Normalmente esse passo é escrito em assembly;
- A carga do kernel inclui:
 - Criar a pilha de execução;
 - Identificar o tipo de CPU;
 - Calcular o total de memória RAM;
 - Desabilitar interrupções;
 - Habilitar a MMU(*memory management unit*);
 - Ajustar o clock;
 - Chamar o código *main()* do kernel.

Boot do Linux

- O código em C do kernel é então executado;
- Cria um espaço para armazenar as mensagens de inicialização;
- Cria as tabelas do kernel;
- Inicia a auto-configuração;
 - Usando os arquivos de configuração, testa cada dispositivo de E/S;
- Carrega dinamicamente os drivers dos dispositivos encontrados.

Boot do Linux

- O sistema de arquivos é montado;
- O processo 1 pode ser então criado;
- *init()*;

Gerenciadores de boot

- GRUB
 - Conhece os sistemas de arquivos;
 - É o mais comum em distribuições Linux;
- LILO
 - Criado pela Intel;
 - Necessita de um mapa do início do disco para efetuar a inicialização;
- EFISTUB, systemd-boot, etc.

Dúvidas?

