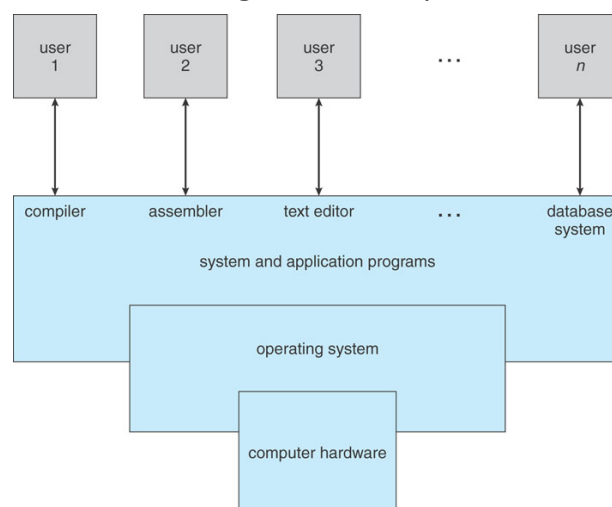


Device Drivers

Arquitetura de Computadores II

António de Brito Ferrari

Sistema de Operação – o software que gere o hardware
Device Drivers – os módulos do sistema de operação que gerem os dispositivos de I/O



ABF - AC2_Device Drivers

2

Device Drivers

- Formam parte do núcleo (*kernel*) do Sistema de Operação
- Fazem com que os dispositivos periféricos respondam a um interface de programação bem definido, escondendo os detalhes de como o periférico opera
- Os utilizadores invocam os serviços de um periférico através de um conjunto de funções standard, independentes do dispositivo específico – o device driver traduz essas funcionalidades em operações específicas do hardware

ABF - AC2_Device Drivers

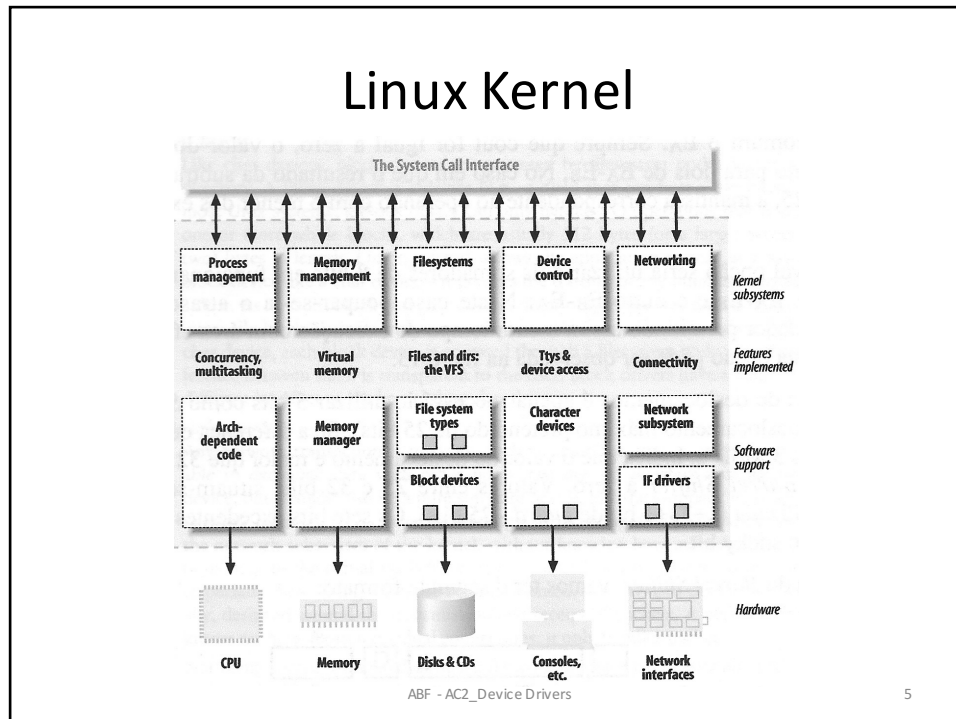
3

Device Drivers (2)

- A escrita de um *device driver* requer um conhecimento aprofundado de como o hardware e o software têm de cooperar na realização das funções (serviços) de uma dada plataforma
- Como os *device drivers* têm acesso a funções a nível do hardware, eles operam em modo privilegiado e podem provocar um desastre se contiverem erros.
- Mesmo *device drivers* operando em modo utilizador podem bloquear um sistema se tiverem erros de programação, enquanto em geral os programas de utilizador nos sistemas de operação atuais podem ser parados sem afetar gravemente o resto do sistema.

ABF - AC2_Device Drivers

4



Interface com os programas utilizadores

- Os *Device Drivers* simplificam a programação, atuando como um tradutor entre o dispositivo h/w e as aplicações ou o sistema de operação que o utilizam – os programadores podem escrever código numa linguagem de alto nível independentemente do h/w específico que é usado
- Exemplo: uma aplicação para usar uma porta série pode ter simplesmente duas funções, *send data* e *receive data*.

O *device driver* que implementa estas funções comunicará com a porta série específica instalada no computador. Embora os comandos necessários para controlar uma UART sejam muito diferentes dos usados para controlar uma porta USB mas cada device driver específico abstrai os detalhes num interface s/w igual (ou semelhante)

ABF - AC2_Device Drivers

6

Operações de Entrada/Saída

- Controlador de I/O (Módulo de I/O)
 - transfere dados entre os periféricos que controla e o seu buffer de memória local
- Device Driver
 - 1 para cada controlador de I/O
 - Conhece o funcionamento do controlador de I/O e apresenta um interface uniforme com o periférico ao resto do Sistema de Operação
 - Permite que o S.O. e outros programas acessem as funções do h/w sem ter de conhecer os detalhes específicos dos dispositivos que estão a ser usados

ABF - AC2_Device Drivers

7

Device Drivers

- Usualmente o fornecedor do sistema de operação escreve o *logical device driver* (LDD) enquanto o *physical device driver* (PDD) é implementado pelo fornecedor do periférico
- No universo dos sistemas embebidos e dos sistemas de operação *open source* os *device drivers* são tipicamente escritos pelos desenvolvedores de aplicações.

ABF - AC2_Device Drivers

8

Operação de I/O

1. Para iniciar uma operação de I/O o device driver escreve nos registos apropriados do controlador do dispositivo
2. O controlador examina o conteúdo dos registos para decidir que ação executar
3. O controlador transfere os dados do periférico para o seu buffer local (**Read**).
4. Quando a transferência está completa o controlador informa o *driver* através de uma interrupção
5. O *device driver* devolve o controle ao Sistema de Operação, retornando os dados ou um apontador para os dados (ou informação de *status* se a operação tiver sido de **write**)

ABF - AC2_Device Drivers

9

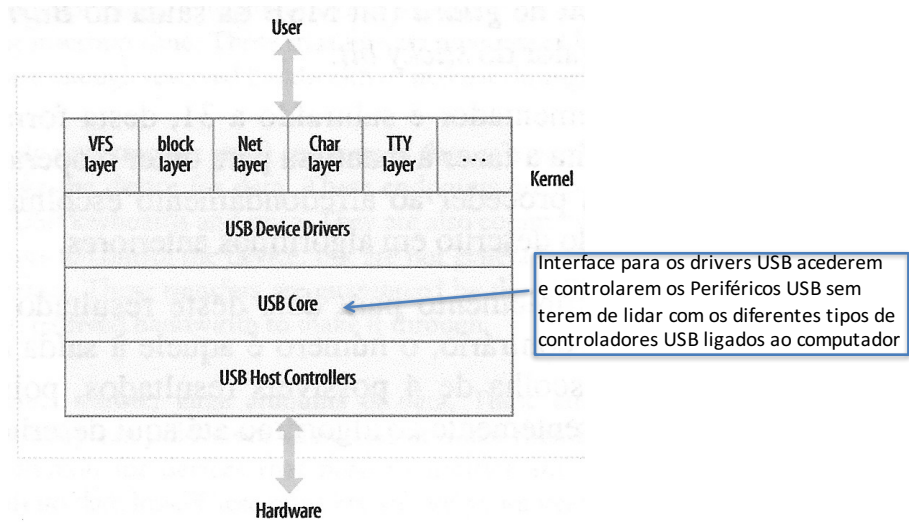
Exemplo: UART Device Driver

- Estrutura de dados – FIFO (*First In First Out*) buffers para transmissão e receção
 - FIFOs são usualmente implementados como **buffers circulares**
- Transmissão
 - **comDrv_putc ()** – escreve os caracteres a enviar no buffer de transmissão
- Receção
 - **comDrv_getc ()** – lê os caracteres colocados no buffer de receção

ABF - AC2_Device Drivers

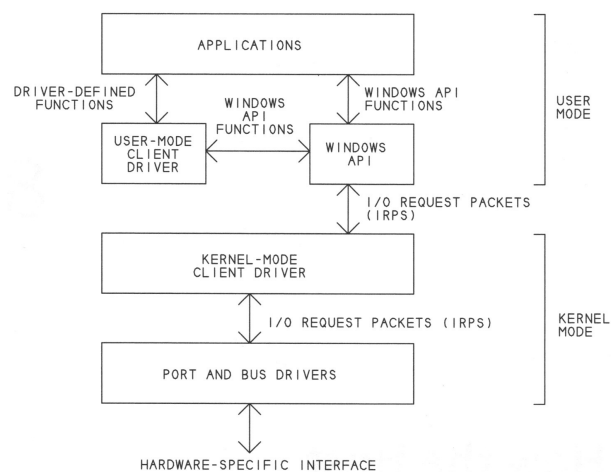
10

Linux USB device drivers



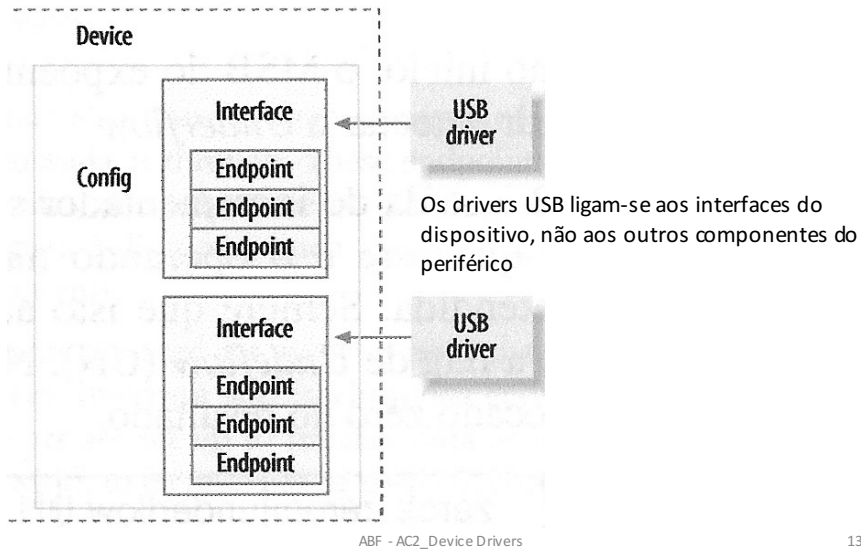
11

USB drivers no Windows



12

Drivers e periféricos USB



13

O Sistema de Ficheiros

- O nível mais baixo do *file system* é formado pelos *device drivers* e *interrupt handlers* para transferir a informação entre o disco e a memória
- O *device driver* traduz comandos do tipo “ler bloco 123” na escrita de bits determinados em posições especiais da memória do controlador do disco

ABF - AC2_Device Drivers

14

Trabalhos práticos

- Os trabalhos 9, 10 e 11 incluem a escrita de device drivers para UART, I2C e SPI
- Gestão de Periféricos e *Device Drivers* fazem parte do programa de Sistemas de Operação