

ALBERTO AVRITZER

~~UNIX~~ UM SISTEMA OPERACIONAL COM SUPORTE PARA MULTIPROCESSAMEN-
TO E TEMPO COMPARTILHADO

Tese apresentada ao Instituto de
Ciências Exatas da Universidade Fe-
deral de Minas Gerais como requi-
sito parcial para a obtenção do
título de *Mestre* em Ciência da
Computação.

BELO HORIZONTE
1983



orientador Roberto da Silva Bisnha, pela sua
motivação e pelo exemplo, com o qual se tornou
um trabalho;

Departamento de Engenharia de Computação pelo apoio
que me deu e que me ajudou a poder ter este

meu mestre, pelo qual

Prof. Ivan Martins Filho, pelo exemplo de quem

Salva pela dedicação e

Centro Experimental de Engenharia de Materiais e

aos meus pais



AGRADECIMENTOS

- Ao meu orientador Roberto da Silva Bigonha, pela dedicação, pela motivação e pelo carinho com que leu as diversas versões deste trabalho;
- Ao Departamento de Ciência da Computação pelo apoio financeiro, sem o qual este trabalho não poderia ter sido realizado;
- Aos meus mestres, pela motivação;
- Ao Prof. Ivan Moura Campos, pela sugestão do nome μ Zúnix;
- À Djalva pela datilografia cuidadosa;
- Ao Centro Esportivo Universitário, pelos momentos de relaxamento.

"Se nada ficar destas páginas, algo, pelo menos, esperamos que permaneça: a nossa confiança no povo. Nossa fé nos homens e na criação de um mundo em que seja menos difícil amar."

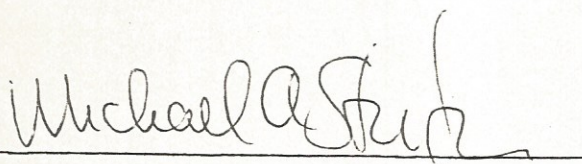
(Paulo Freire, Pedagogia do Oprimido)

FOLHA DE APROVAÇÃO

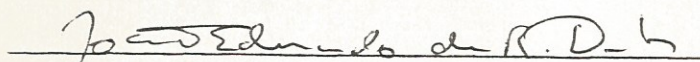
UNIX: UM SISTEMA OPERACIONAL COM SUPORTE PARA MULTIPROCESSAMEN-
TO E TEMPO COMPARTILHADO"

ALBERTO AVRITZER

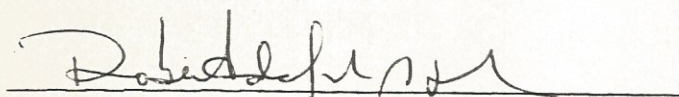
Tese defendida e aprovada pela banca examinadora constituída dos
senhores:



Prof. Michael Stanton



Prof. João Eduardo de Rezende Dantas



Orientador:

Prof. Roberto da Silva Bigonha

DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO DO ICEx

Belo Horizonte, 26 de setembro de 1983

SINOPSE

Este trabalho contém a especificação do Sistema Operacional μ Zúnix do ponto de vista do usuário e de implementação.

O μ Zúnix é um Sistema Operacional baseado no Unix da Bell, que pode operar em ambientes com multiprocessamento.

A linguagem de especificação da implementação é MODULA-2. Cada módulo do sistema está acompanhado de uma ampla documentação.

Os operadores e objetos dependentes de máquina estão concentrados em um módulo à parte.

ABSTRACT

This work presents the μ Zunix Operating System specification from the implementation and user's point of view.

μ Zunix is a Unix-like Operating System which can operate in a multiprocessing environment.

MODULA-2 is the specification language. Every MODULA-2 module in the system is accompanied by its own documentation.

The machine dependencies of the System are highlighted in a separate module.

S U M Á R I O

PÁGS.

I	- INTRODUÇÃO.....	2
II	- UNIX.....	5
	1. Por que o Unix?.....	5
	2. O Sistema de Arquivos.....	8
	2.1 - Descrição Geral.....	8
	2.2 - Arquivos.....	9
	2.3 - Diretórios.....	11
	2.4 - Dispositivos de Entrada e Saída.....	12
	2.5 - Montagem de Dispositivos.....	12
	2.6 - Proteção.....	13
	2.7 - Primitivas do Núcleo que Realizam Entrada e Saída.....	14
	2.7.1 - open e create.....	14
	2.7.2 - read e write.....	15
	2.7.3 - lseek.....	16
	2.7.4 - close e unlink.....	16
	3. Processos.....	18
	3.1 - Introdução.....	18
	3.2 - Criação de Processos.....	18
	3.3 - Comunicação entre Processos.....	18
	3.4 - Sincronização de Processos.....	19
	3.5 - Terminação de Processos.....	20
	4. Execução de Programas.....	21
III	- SHELL DO UNIX.....	23
	1. Introdução.....	23
	2. Construções Básicas.....	23
	2.1 - Comandos Simples.....	23
	2.2 - Entrada e Saída Padrão.....	24
	2.3 - Redirecionamento de Entrada e Saída.....	24

2.4 - Filtros e Linhas de Montagem.....	25
2.5 - Utilitários.....	26
2.6 - Separador de Comandos e Processamento Múltiplo de Tarefas.....	27
2.7 - Comandos Condicionais.....	28
2.8 - Comandos de Repetição.....	29
2.9 - Comando Case.....	32
3. Procedimentos.....	34
3.1 - Documentos Locais.....	35
3.2 - Variáveis do Shell.....	36
3.3 - Passagem de Parâmetros.....	38
3.4 - Avaliação de Parâmetros.....	39
4. Avaliação de Comandos.....	40
5. Geração de Nomes de Arquivos.....	41
6. Ordem de Avaliação e Abolição do Significado de Metacaracteres.....	43
7. A Gramática do Shell.....	45
8. Metacaracteres e Palavras Reservadas.....	47
- ESPECIFICAÇÃO DA IMPLEMENTAÇÃO DO μ ZUNIX.....	50
1. Introdução.....	50
2. Estruturas de Dados Principais.....	51
3. Estruturas Relacionadas com o Sistema de Arquivos.....	52
3.1 - Inodo.....	52
3.2 - File.....	54
3.3 - User.....	54
3.4 - Superbloco.....	55
3.5 - Mount.....	56
4. O Sistema de Buffers.....	57
5. Organização dos Dados nos Dispositivos Blocados..	59
6. Manipulação dos Dispositivos Seriais.....	59
7. Estruturas Relacionadas à Gerência de Processos..	60
7.1 - Proc.....	60

7.2 - Text.....	62
7.3 - User.....	63
8. O Mecanismo de Sincronização do Unix.....	63
8.1 - Sleep e Wakeup.....	63
8.2 - Tratamento de Interrupção.....	64
9. Multiprocessamento do μ Zūnix.....	65
9.1 - Introdução.....	65
9.2 - O Mecanismo de Sincronização.....	66
9.2.1 - Sleep e Wakeup.....	66
10. Implementação do Multiprocessamento do μ Zūnix....	68
10.1- Ferramentas Necessárias.....	68
10.1.1- Test and set.....	68
10.1.2- Sleep.....	68
10.2- Implementação de Regiões Críticas.....	69
11. Políticas Adotadas.....	70
11.1- Alocação Dinâmica de Memória.....	70
11.2- Gerência de Memória.....	70
11.3- Gerência do Processador.....	71
- DEPENDÊNCIA DE MÁQUINA.....	73
1. Introdução.....	73
2. Rotinas que devem ser Implementadas em Linguagem de Máquina.....	74
3. Constantes Dependentes de Máquina.....	76
4. Drivers.....	77
5. Tratamento da Interrupção de Relógio.....	77
6. Inicialização do Sistema.....	78
VI - ESPECIFICAÇÃO DO NÚCLEO DO μ ZŪNIX EM MÓDULO-2.....	80
1. Introdução.....	80
2. Módulo inod.....	83
3. Módulo fil.....	86
4. Módulo user.....	88
5. Módulo mountt.....	94

6. Módulo <i>filesys</i>	96
7. Módulo <i>buff</i>	98
8. Módulo <i>tty</i>	102
9. Módulo <i>sysrdw</i>	105
10. Módulo <i>sysopen</i>	122
11. Módulo <i>sysmount</i>	151
12. Módulo <i>syspipe</i>	165
13. Módulo <i>sysbuilt</i>	183
14. Módulo <i>igett</i>	198
15. Módulo <i>name</i>	220
16. Módulo <i>subr</i>	233
17. Módulo <i>alloca</i>	242
18. Módulo <i>fio</i>	270
19. Módulo <i>bio</i>	297
20. Módulo <i>textt</i>	334
21. Módulo <i>process</i>	336
22. Módulo <i>sysproc</i>	338
23. Módulo <i>slp</i>	363
24. Módulo <i>mem</i>	392
25. Módulo <i>txt</i>	401
26. Módulo <i>sig</i>	418
27. Módulo <i>tmout</i>	431
28. Módulo <i>system</i>	436
29. Módulo <i>param</i>	440
III - 0 μSHELL	445
1. Descrição Geral.....	445
2. A Máquina Abstrata do μ Shell.....	447
2.1 - Estrutura de Dados.....	447
2.2 - Instruções da Máquina Abstrata.....	448
2.3 - Exemplos de Geração de Código Intermediário.....	452
3. Algoritmos em Modula-2 do μ Shell.....	455
3.1 - Introdução.....	455

PÁGS.

3.2 - Módulo analisador.....	456
3.3 - Módulo inter.....	518
CONCLUSÃO.....	555
REFERÊNCIAS BIBLIOGRÁFICAS.....	559



LISTA DE FIGURAS

	PÁGS.
Figura 1 - Estrutura do Sistema de Arquivos.....	8
Figura 2 - Imagem do Processo.....	51
Figura 3 - Estruturas de Dados Mantidos na Memória.....	52
Figura 4 - Estrutura dos Arquivos em Dispositivos Blocados	236
Figura 5 - Passagem de Parâmetros para a Primitiva exec...	343

