

NOSSO COMPUTADOR
HIPOTÉTICO
X
UM COMPUTADOR
REAL

READ

WRITE

STORECONST

LOAD

STORE

JUMP

ADD, MULT, DIV

READ E WRITE:

NÃO EXISTEM

CPU NÃO COMUNICA-SE
COM PERIFÉRICOS

INTERFACES

A stylized, dark teal silhouette of a mountain range is positioned in the bottom right corner of the slide, partially overlapping the text 'INTERFACES'.

DRIVE

DRIVER

INTERFACES



DRIVE: DISPOSITIVO FÍSICO

INTERFACE: PLACA
CONTROLADORA DO DRIVE

DRIVER:

???????

DRIVER:

É O SOFTWARE QUE
CONVERSA COM A PLACA
CONTROLADORA DO
DISPOSITIVO.

NORMALMENTE É DESENVOLVIDO
PELO FABRICANTE DO
DISPOSITIVO E INCORPORADO AO

SISTEMA OPERACIONAL.

A stylized, dark teal silhouette of a mountain range is positioned in the bottom right corner of the slide, partially overlapping the text area.

COMO TUDO ISSO É
FEITO ? ? ? ?

NORMALMENTE NÃO É PERMITIDO A UM
PROGRAMA DE USUÁRIO FAZER
OPERAÇÕES DE INPUT/ OUTPUT. ESSAS
OPERAÇÕES SÃO DE RESPONSABILIDADE
DO **SO**.

MAS COMO O **SO** FICA SABENDO QUE O
USUÁRIO QUER FAZER I/O
????????????????????

ATRAVÉS DE UMA CHAMADA DE SISTEMA (**SYSTEM CALL**, **TRAP**) FEITA PELO PROGRAMA DO USUÁRIO.

SEMPRE QUE UM PROGRAMA DE USUÁRIO NECESSITA DE UM SERVIÇO DO **SO** ELE O SOLICITA ATRAVÉS DE UM **SYSTEM CALL**.

MAS O QUE É UM SYSTEM
CALL, COMO ISSO PODE SER
FEITO ???????

NADA MAIS É DO QUE
INSTRUÇÃO DE MÁQUINA:
INSTRUÇÃO QUE CAUSA
INTERRUPÇÃO.

SÃO CHAMADAS DE INTERRUPÇÕES
POR SOFTWARE!

OH LOCO SO!

ENTÃO NÃO EXISTEM READ,
WRITE, NADA DISSO ???

EXISTEM SIM MAS EM FORMA DE
PROCEDURES OU CLASSES JÁ
PRONTOS, NORMALMENTE,
DISPONIBILIZADOS NA BIBLIOTECA
DA LINGUAGEM QUE FOI UTILIZADA
PARA CONSTRUIR O PROGRAMA DO
USUÁRIO.

NO INTERIOR DE UM

READ

OU DE UM

SYSTEM.OUT.PRINTLN

CERTAMENTE EXISTIRÁ

UM SYSTEM CALL

UM **SYSTEM CALL** É ENTÃO UMA INSTRUÇÃO DE MÁQUINA (POR EXEMPLO: INT 21) QUE QUANDO EXECUTADA PELA **CPU** FAZ COM QUE O CONTROLE PASSE PARA O **SO**, OU SEJA, A **CPU** DEIXA DE EXECUTAR O PROGRAMA DO USUÁRIO QUE SOLICITOU O SERVIÇO E VAI EXECUTAR UM **TRECHO DO SO**.

QUE TRECHO SERIA ESSE ????

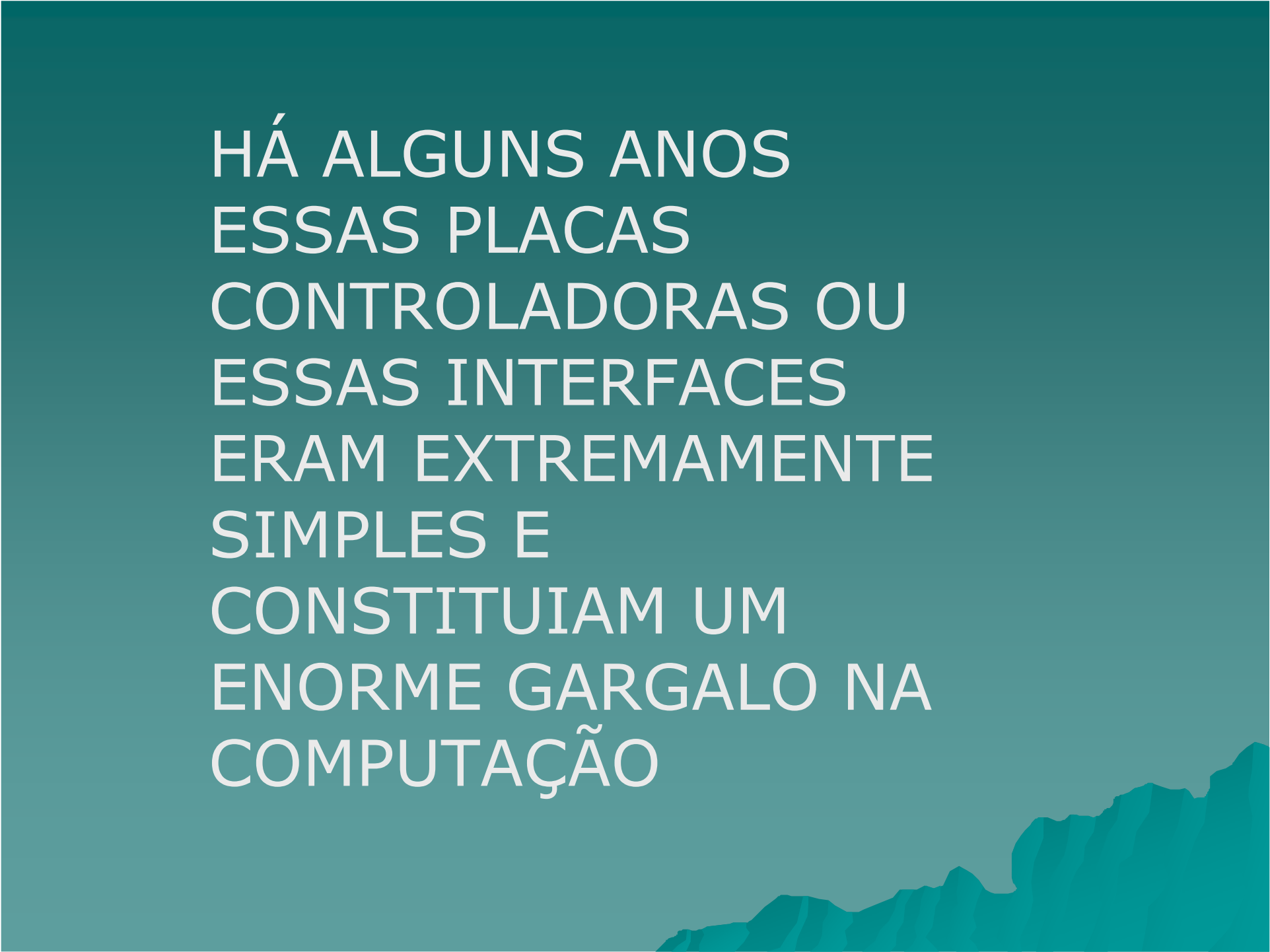
ESSE TRECHO CORRESPONDE AO
TRATADOR DAQUELA
INTERRUPÇÃO OCORRIDA E
NORMALMENTE É O **DRIVER** DO
DISPOSITIVO ENVOLVIDO.

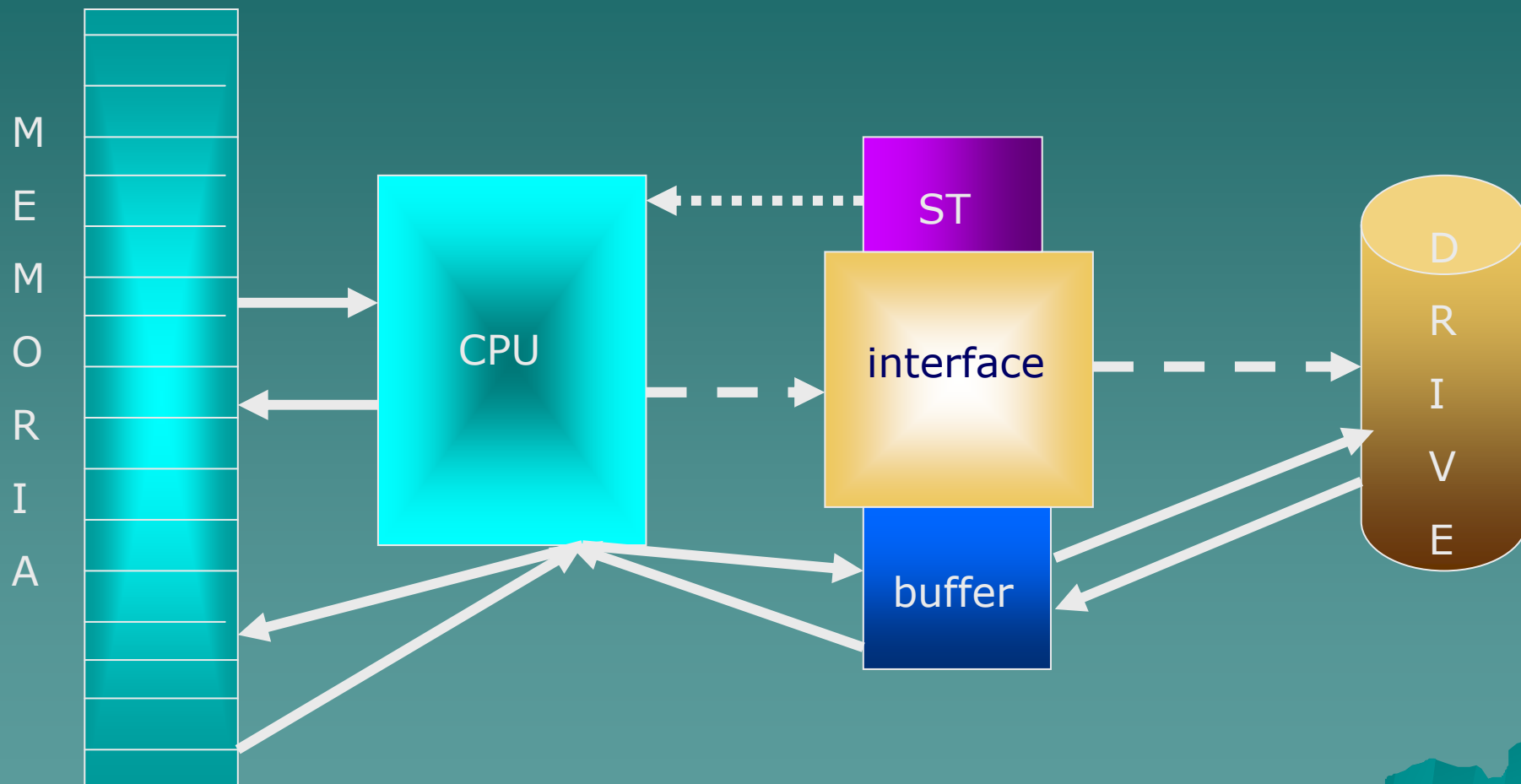
OU SEJA, É AQUELE SOFTWARE
FORNECIDO PELO FABRICANTE DO
DISPOSITIVO QUE FOI
INCORPORADO AO **SO** QUANDO DA
SUA INSTALAÇÃO.

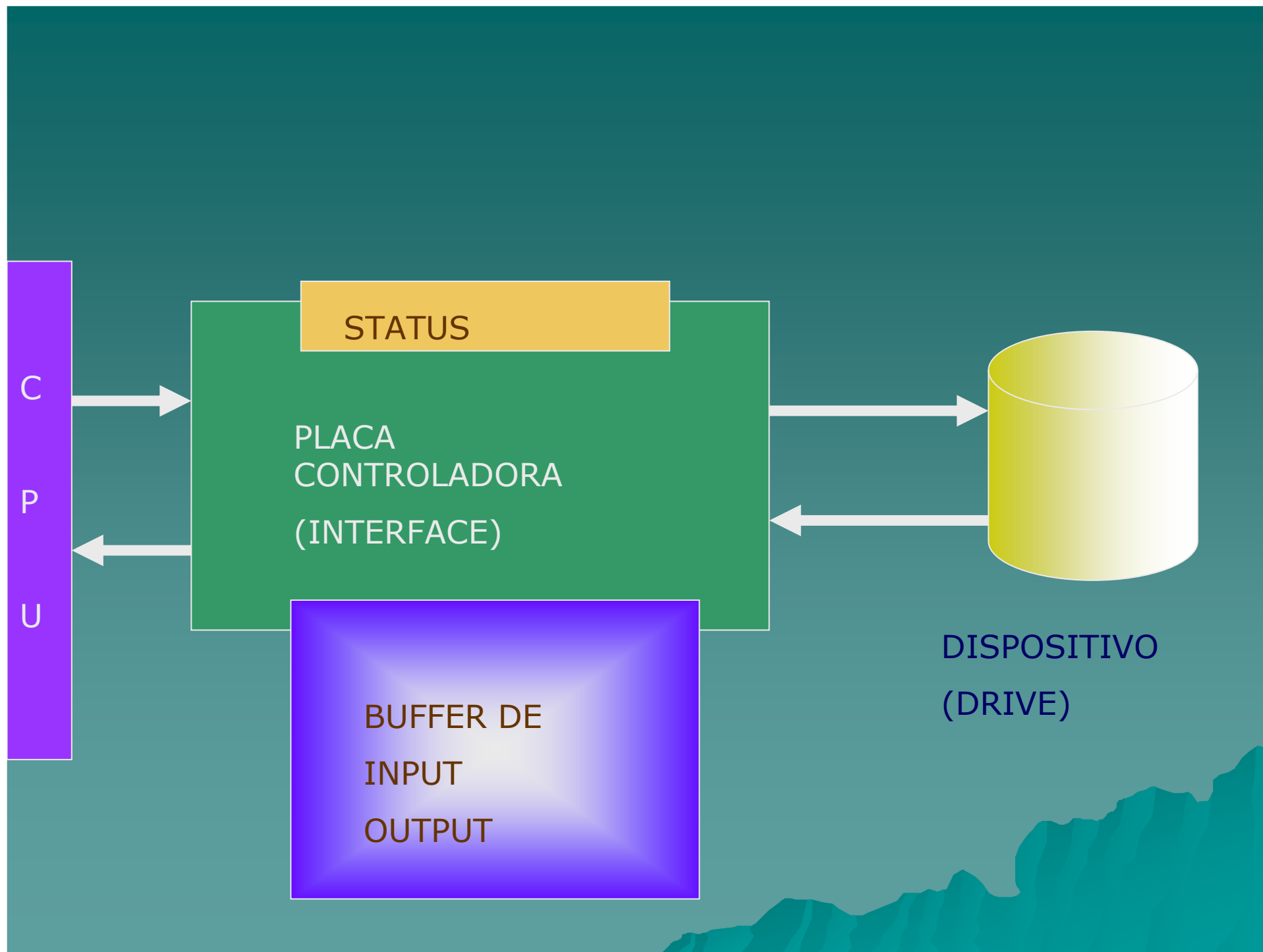
É SOMENTE ESSA A FUNÇÃO
DO DRIVER ?????

NÃO, ELE TAMBÉM TRATARÁ A
INTERRUPÇÃO (DE HARDWARE
AGORA) CAUSADA PELA PLACA
CONTROLADORA DO DISPOSITIVO
QUANDO DA CONCLUSÃO DA
OPERAÇÃO.

HÁ ALGUNS ANOS
ESSAS PLACAS
CONTROLADORAS OU
ESSAS INTERFACES
ERAM EXTREMAMENTE
SIMPLES E
CONSTITUÍAM UM
ENORME GARGALO NA
COMPUTAÇÃO

The background is a solid teal color. In the bottom right corner, there is a stylized silhouette of a mountain range in a slightly darker shade of teal.





HOJE ESSAS PLACAS SÃO
VERDADEIROS
COMPUTADORES
ESPECIALIZADOS EM
INPUT/OUTPUT, COM:
INTERRUPÇÃO E DMA

O QUE MAIS QUE A NOSSA
POBRE MÁQUINA NÃO TEM??

REGISTRADORES: A COITADA SO
TEM UM! O ACUMULADOR (ACC)

QUANTOS REGISTRADORES
POSSUE UMA MÁQUINA REAL???

DEPENDE TOTALMENTE DA
ARQUITETURA DO PROCESSADOR MAS
A MAIORIA:

ACC

PC

SP

ST

RI

ATUALMENTE APRESENTAM-SE DUAS
LINHAS DE ARQUITETURAS DE
COMPUTADORES BEM DISTINTAS:

CISC:

COMPLEX INSTRUCTION
SET COMPUTER

RISC:

REDUCED INSTRUCTION
SET COMPUTER

A stylized, dark teal silhouette of a mountain range is positioned in the bottom right corner of the slide, partially overlapping the text area.

CARACTERÍSTICAS DAS ARQUITETURAS CISC:

- CONJUNTO GRANDE DE INSTRUÇÕES;
- FORMATOS COMPLEXOS DE INSTRUÇÕES;
- MICROPROGRAMAÇÃO
- PERFORMANCE MENOR EM ALGUNS ASPECTOS

CARACTERÍSTICAS DAS ARQUITETURAS RISC:

- CONJUNTO PEQUENO DE INSTRUÇÕES;
- FORMATOS SIMPLES DE INSTRUÇÕES;
- OPERANDOS SEMPRE EM REGISTROS;
- MODOS SIMPLES DE ENDEREÇAMENTO À MEMÓRIA

RISC:

DEC ALPHA

SPARC

MIPS

POWERPC

CISC: FAMÍLIA X86 (desktops)

ARQUIVOS FONTES

ASSEMBLER

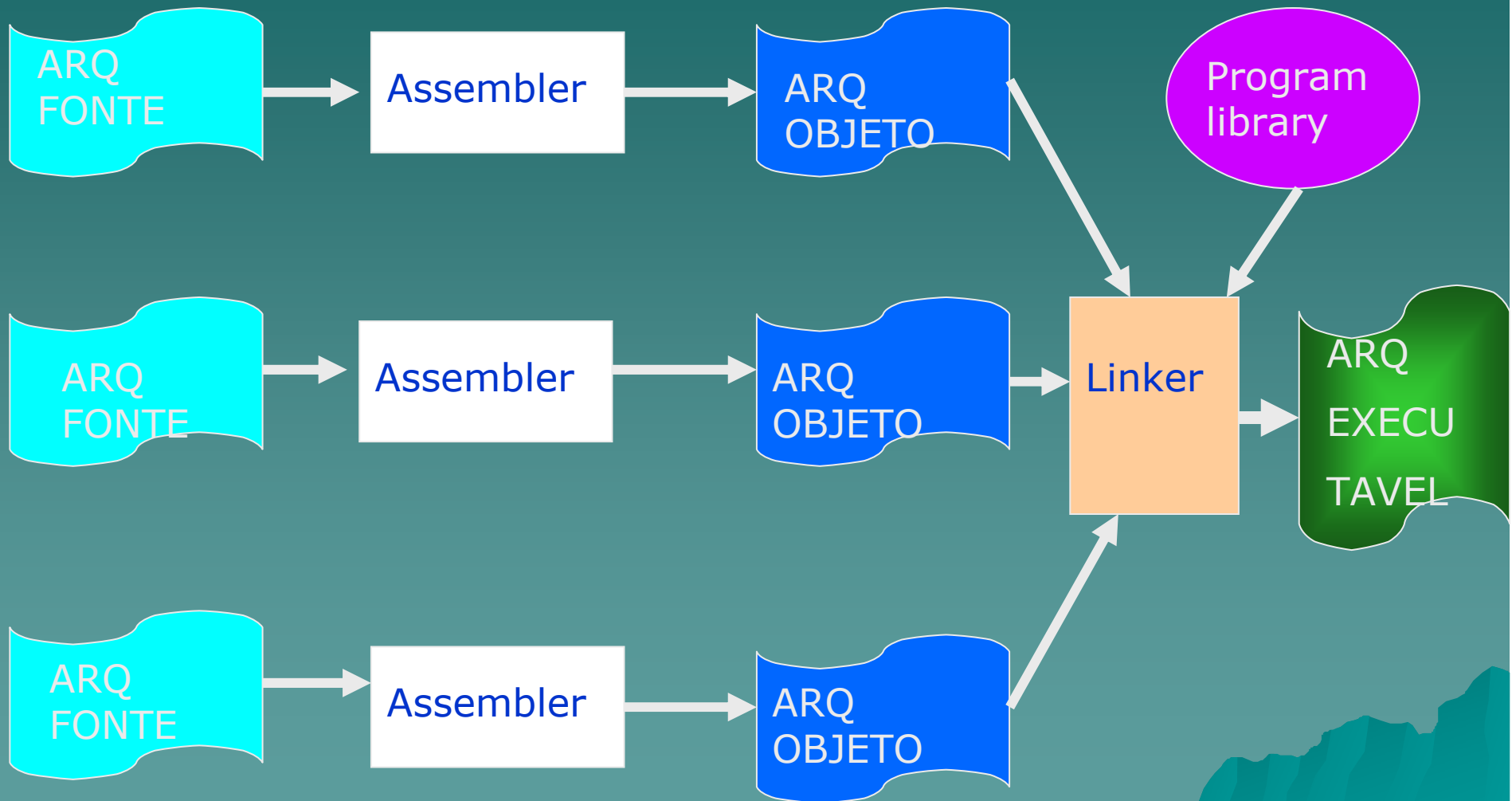
ARQUIVOS OBJETOS

LINKER

BIBLIOTECAS

ARQUIVOS EXECUTÁVEIS





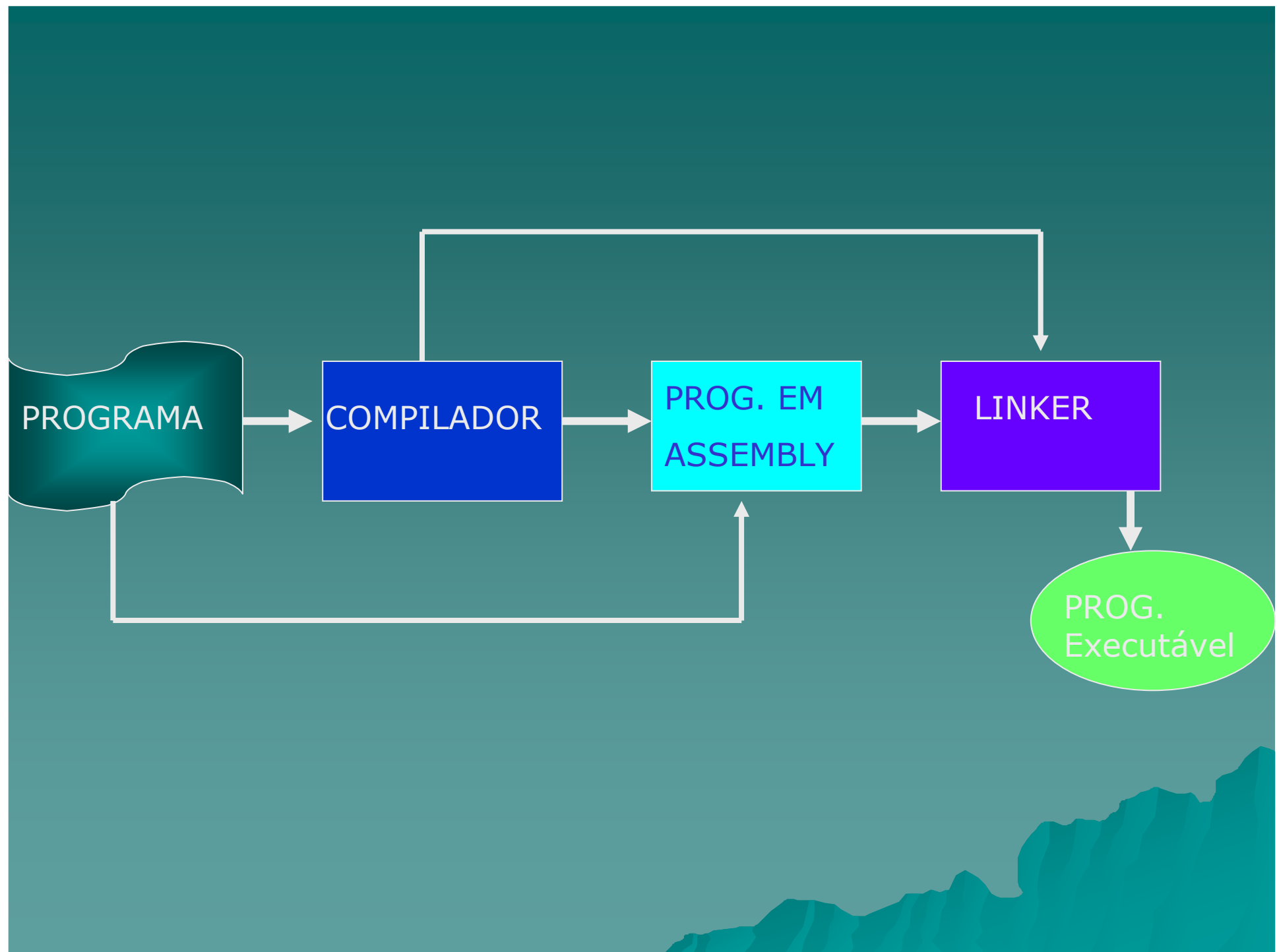
ASSEMBLER:

UM PROGRAMA QUE TRADUZ
UMA VERSÃO SIMBÓLICA
DE UMA INSTRUÇÃO NA SUA
VERSÃO BINÁRIA.

ASSEMBLY:

UM PROGRAMA ESCRITO
UTILIZANDO INSTRUÇÕES
SIMBÓLICAS

A stylized silhouette of a mountain range in a darker shade of teal, located in the bottom right corner of the slide.



0101010101000101010101010101010
100101010101010101010100001010001
0010101000010010101010101010001
0101000010011101010101010101011
0101000101010101010101010101010
0101010101000001001100010001001
0101000100010101010101001010100

.....

.....

.....

0010010000100101110101000101001
0100100001010000100100010001000

O QUE É A MÁQUINA DO
RODRIGO?

RODRIGO CONSTRUIU UM
SIMULADOR PARA A
NOSSA MÁQUINA
HIPOTÉTICA

O QUE É A MÁQUINA
VIRTUAL JAVA: JVM ???