

CAPÍTULO 2 - REPRESENTATION OF INFORMATION

Binary System

- the possible values are finite;
- is the simplest form of digital systems
- there are two possible values for the signals
- é usado pq na eletrônica é mais fácil definir com "tem corrente ou não tem corrente", por exemplo.
- Se houver mais níveis de distinção é menos viável;
- é o método mais viável for encoding digital information

Digital Abstraction

- A small variation in an input value is still interpreted correctly
-
-

Bits

- Valor lógico : 0 ou 1
- 2 bits $\leftrightarrow$ 4 patterns 00 / 01 / 10 / 11
 $\hookrightarrow +1$ bit $\hookrightarrow \times 2$, então 3 bits $\leftrightarrow$ 8 patterns
 $\hookrightarrow$ ou seja nº patterns = $2^n \rightarrow$ nº bits
- a cada padrão de bits é atribuída uma informação

8 bits são um byte

$\hookrightarrow$ + pequena unidade de memória

Meio byte é um nibble, nibbles são inúteis

* tabela de multiple-byte units: decimal e binário *

Main Memory

Se tem 8 bits de memória tem $2^8 = 256$ células de memória
Cada byte na memória é identificado pelo seu endereço

TEXTUAL INFORMATION

ASCII → alfabeto para computadores representarem o texto humano

→ Cada caractere é uma 7 bit string

→ representa poucos símbolos pois $2^7 = 128$, só representa

128 caracteres com as strings de 7 bits, então:

Modern computers use UNICODE, a última versão tem mais
de 149 mil símbolos

↳ Sept. 2023

MACHINE-LEVEL INSTRUCTIONS

FORMAT OF INSTRUCTIONS

→ in machine-level each instruction is represented by a bit pattern

OPCODE and some additional info

↳ where operands come

↳ where to store results

opcode

opcode | reg |

" | | |

...

→ On some machines all instructions have the same length, in others they may be different;

→ the opcode of each instruction must be associated to an unique pattern;

R instructions

↳ all data values are located in registers

↳ Syntax: OP rd, rs, rt

add \$t1, \$t2, \$t3

00000 | 01010 | 01011 | 01001 | 00000 | 100000

opcode | rs | rt | rd | shamt | funct

6bits 5bits 5 5 5 6bits

IMAGES

RASTER IMAGES

A digital image is a grid of small points

raster image or bitmap

Cada ponto é um pixel

tem
3
bytes

0	1	1	0
1	0	0	1
1	0	0	1
1	1	1	1
1	0	0	1
1	0	0	1

é representado por um
padrão binário

Dada a sequência binária, sabemos quantos pixels
tem a largura e altura através dos métodos,
que têm de ser stored

Em imagens preto e branco, o preto é por exemplo
1 e branco 0. Para criar cores → ↑ nº bits por pixel

COLOUR DEPTH → número de bits usado to store
each pixel

A foto a cor tem 6 bytes

↑ cor → ↑ bits → ↑ ficheiro

Image Resolution

- ↳ is related to how close the pixels are
 - ↳ is usually measured in dpi (dots per inch)
 - ↳ low resolution image → the pixels are larger
- fewer are needed to fill the space

Audio

Sampling

To process audio, computers need to convert it into a digital format.

The microphone translates sound waves into an electrical signal.

Periodic measurement of the level of that signal are registered

The process that reduces a continuous-signal to a discrete-time signal is called sampling.

The value at a point in time is called sample.



Samples are then converted into binary (w/unique binary code)

- Now the computer can process the digital sound, as a -
- Sequence of bits -

Sampling Rate

↳ measured in Hertz

↳ n° of samples per second

↑ sampling rate → ↑ quality of audio

In bits, the size of the sound signal s :

$$S = f \times n \times t$$

Sample
Rate
(Hz) ✓

↓
Sample
Resolution
(n° of bits)

↘ Sample
duration
(seconds)