Computer Science

Mathematical activities

Theoretical activities

Computer Science
def.

Sub-disciplines:

- Architecture
- Software
- Theory

Digital Devices

Memory

Input

Output

Computer Softwares

General purpose application softwares (ex)

Special purpose application softwares (ex)

Programming languages, compilers, interpreters (ex)

Operating systems (ex)

High-level languages:
- not machine oriented
- problem oriented
- “English grammar”

Low-level languages:
- machine oriented
- assembly language (difficult and long)

Translating and running programs

Sequence: source code - compiler - object code

- EDITORS: enter and edit programs.
- COMPILERS AND INTERPRETERS: translate into machine code
- LINKAGE EDITORS: link to executable code
- DEBUGGERS: find logic errors

Microprocessors

Sequence: silicon slicing - pollution - baking

Advantages:

- speed (100.000-2 million/sec.)
- familiarity
- availability of equipment

Windows vs MacIntosh
or
Microsoft vs Apple

Affordability
Selection
Gaming
Big screen connectivity
Popularity
Software compatibility
3D rendering

Stability
Security
Reliability
Multimedia
Compatibility
Typography
Color matching
Speed test

Visual Basic

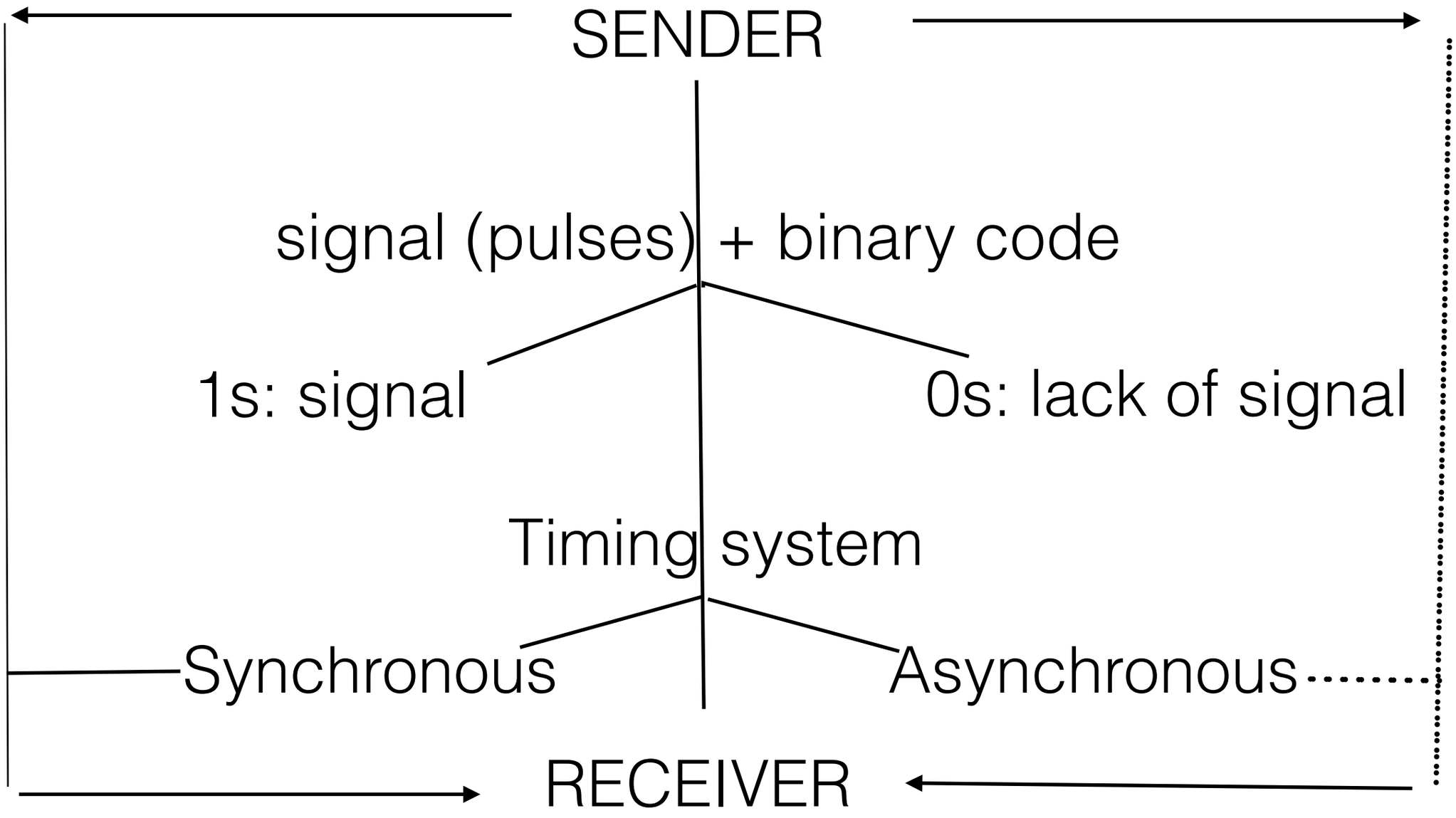
Objects: combinations of code and data treated as a unit

They support:

- PROPERTIES: attribute or aspect of behaviour
- METHODS: action
- EVENTS: external stimulus
- FORMS: foundation to create the interface

Design time vs Run time

Digital communication



LANs and WANs

LAN: Local Area Network

- same building or room
- sharing of devices
- server (memory)

Wan: Wide Area Network

- bigger geographical areas
- external links (satellites, mocrowaves, etc.)
- private owners
- you have to pay for connection

ISO-OSI protocols

International Organization for Standardization

7 layers:

7. Application
6. Presentation
5. Session
4. Transport
3. Network
2. Data link
1. Physical