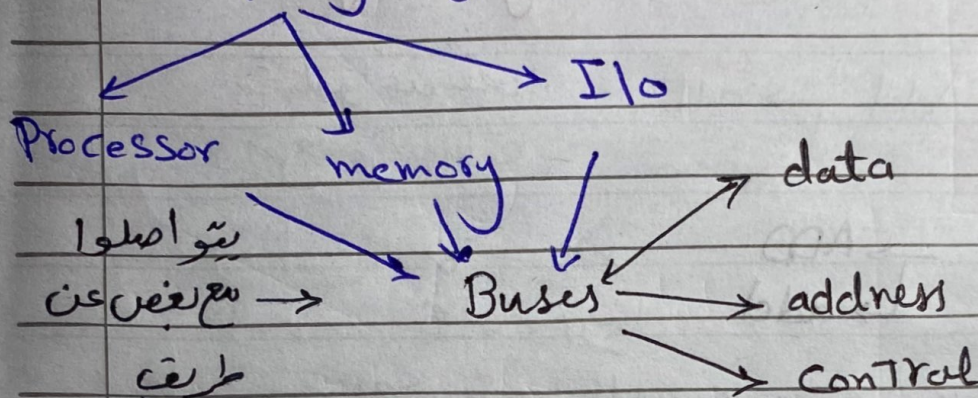


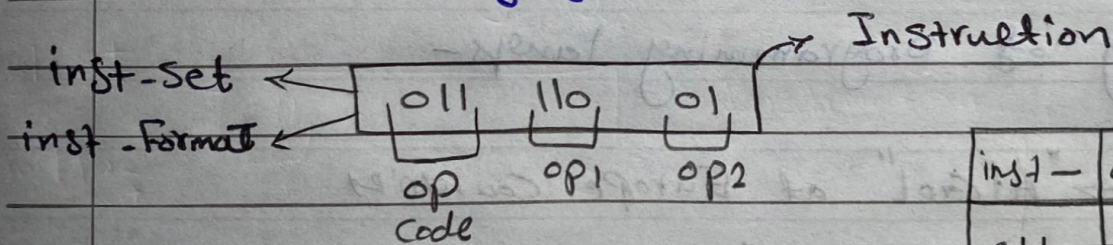
Computing System



MP (Microprocessor) Instruction مخزنة في الذاكرة (memory)
 من الذاكرة (MP) يتم المبرمج كونه قارئ الذاكرة - Instruction
 لغة مبنية في الذاكرة memory

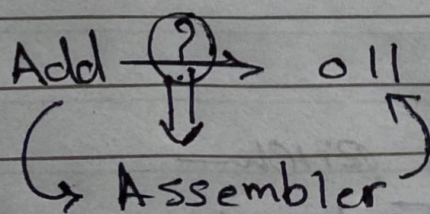
* Language of microprocessor (mp)

[I] Machine Language (0,1)



| inst - | opera - |
|--------|---------|
| 011 | → Add |
| 1 | 1 |
| 1 | 1 |

[2] Assembly lang



Problems related to Assembly lang

- | | | | |
|---|---------|------------------------------------------------------------------------|---------------------------------------------------------|
| ① | op Code | Addl $\begin{cases} \rightarrow 011011 \\ \rightarrow 000 \end{cases}$ | يَجْعَلُوا عَيْنَ رَجُلٍ عَنِ وَجْهِ الْعَمَلِيَّاتِ |
|---|---------|------------------------------------------------------------------------|---------------------------------------------------------|

- (2) Instruction - → ADD سلة في
 ↘ add بخره خلت

- ⑤
- | HP | مقدار | نوع | نوع |
|----|-------|-----|-----|
| HP | ✓ | ✓ | ✓ |
| HP | ✗ | ✗ | ✗ |
- فيديو mp3 يرفعه علياء،
الضرب والتمزيق و خ مبيوفه ماسا

- ④ difficult

Advantages of Assembly lang

- ① Optimised size of execution time
- ② No need to Compiler - (just Assembler)

⊛ history of programming lanegs-

- 1- 1960 → "ALGOL" at Europe countries
- 2- 1967 → "BCPL" → "Martin Rich"
 ↓
 Basic Compiled Programming Language
- 3- 1970 → "B" → Based on Unix
 ↓
 BCPL + Features
 by (AT&T) & (Bell Lab)
- 4- 1972 → "C" → Dennis Ritchie
 ↓
 ALGOL + BCPL + B + Features

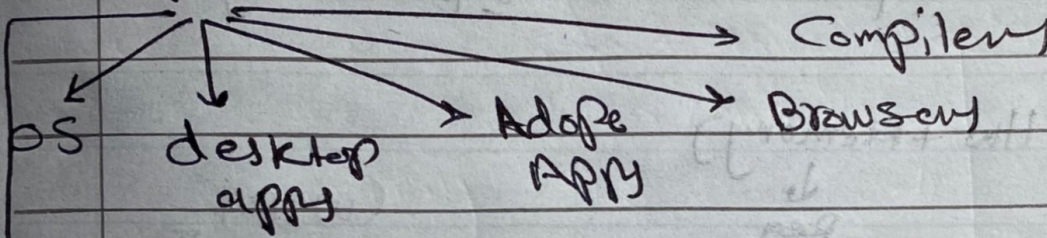
(ANSI)

1989 → C89 Standard

ISO accepted it at 1990

ANSI C = C89

Applications of C



→ (ES) Embedded Systems → AVR -

This is the main language of ES

⊗ Why C Not Assembly ?

1- C is easier & Fast to develop & Simple more than assembler.

2. Portable → platform
high quality compiler

قابل للتنقل (platform) C هي أفضل
الذي يمكنه تشغيله على كل شيء

① Why C for embedded?

(I2C)

Embedded

C80 C80

② Memory allocation

* تغير عمل access

↳ directly "Address"

لا memory لأن الـ C فيها
Pointers

* لغة قريبة من الـ hardware
لا يمكن أن نكتب الـ C بدون الـ hardware
registries والحكم فيها في الـ hardware
Peripherals

Peripherals → (HW + Memory)

↓
Reg-

(WIR)

master

CPU

CPU diso

CPU

master

Bus

R/W

R/W

عبر

RAM

I2C

GPIO

Master → Initiate transaction

STO

Lod

assembly
Instruction

C

Compiler

A

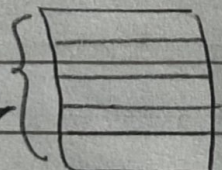
* الـ C lang. Prag. يغير فيها Pointers

لأن الـ C لا يمكن أن نكتب الـ C بدون الـ hardware
Lod & STO

③ Portable Code / Performance

④ Layout → Flash

"linker" →



⊗ Building process / Compilation Process

File.c → executable file
(.exe) or (.hex)

Tool-Chain

↓
GCC → GNU C Compiler
(Free + open Source)

tool Chain

Native

Cross

run on the same

run on another Arch

Architecture

يعني تشغيل ال (.exe) على نفس
ال processor الذي سطره على
tool chain

.c → [] → .bin
AVR Arm

tool chain → instal → PC (intel)
MP
.c → [] → (.exe)

(.c) → [Preprocessor] (.i) → [Compiler] (.s) → [Assembler] (.o) File → object

لإبشيل أي * عند ربط الكود
المكافئ له في المكتبة والمكرو
بيفك الكود أو ينفذ app

(.exe) ← [Linker] ←
Linker script Static Lib

PrintF مثل Standard Libraries

فيها تعريفات

ROX