Section ①

embedded system : Computing system has limited

- Mem
- procssor
- i/0 interface.

We have 2 way to implement this system :

A] system board.

- Ic Mem / Ic i/0 / Ic processor
- PCB
- Software.

B] system On chip.

- Ready chip From Factory.

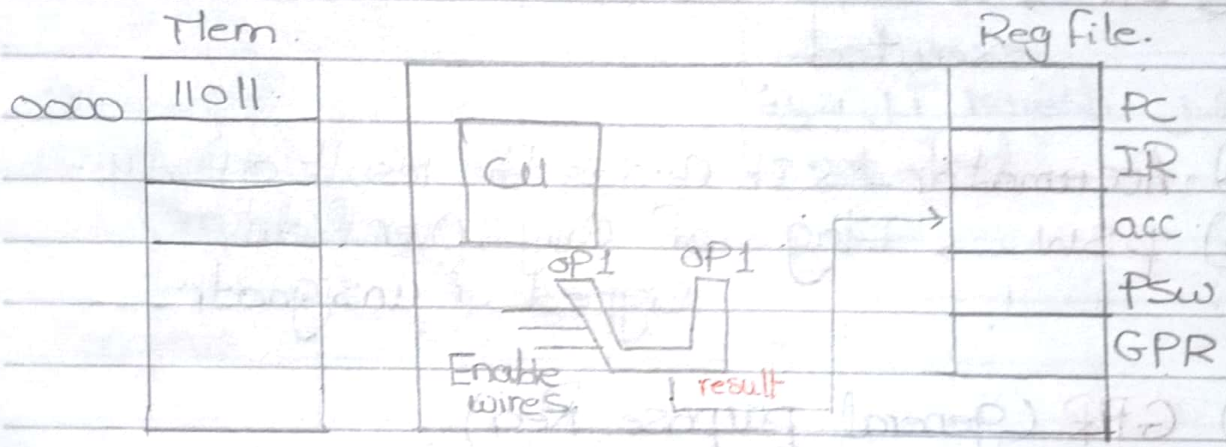
Advantage | disatvantage of each board.

	A	B
① Power Consumption	high	low.
② Size..	high	low
③ Cost	high.	low.
④ Configration.	development. we Can modify the board, So it Can be developed.	production. we Can't make any modification so, it is used in large production

* Micro Controller → Mem. } Carry Code.
→ processor
→ I/O interface.

* Micro processor.
(the processor function is to execute Code)

Processor diagram:-



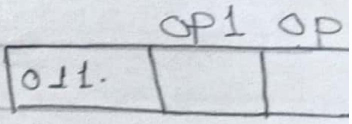
steps

(1) Control Unit (CU) → هيشوف انهو سطر عليه الدور
processor ← to Mem ويسحبه من الـ

program Counter ← (PC) هيستخدم
After that pc will increment to the the next adress to be executed.

(2) ID → decoding.

* Each Code has its Own instruction Format



The ID will Compare this format by the instruction set & will enable ALU according to its function → add, -

③ ALU → function execution & store the result in the accumalator.

④ wb → writing back in Mem.

Note ① PC → it Carries the next address of inst. to be executed.

② IR → it Carries the return adress after Calling function "Simillar to ra in MIps" Linker Reg

③ Accumator → it Carries the result of ALU.

④ PSW → Flag of Carry, Over flow, signed or unsigned, ...

⑤ GPR (general purpose Reg)

↳ It's limited.

↳ it's used to Save local Variable, which is Used many time in the prog.

(GPR ⟵ Variable بنفسه بيسجل User ممكن)

Typs of mem:

* RAM (Volatile) → All data is lost when electricity is off.

SRAM
depends On transistor
when transistor ON
1 → channel has carriers ON.

0 → OFF
So channel is Closed

DRAM
depends On Cap —||— C

1 → when C is charged.
0 → when C is empty

محتاج اعمل ال refresh كل شوية
عشان ال C ← not ideal.

Compare	SRAM	DRAM
Power Consumption	↑	↓
Speed	↓	↑
Cost	↑	↓

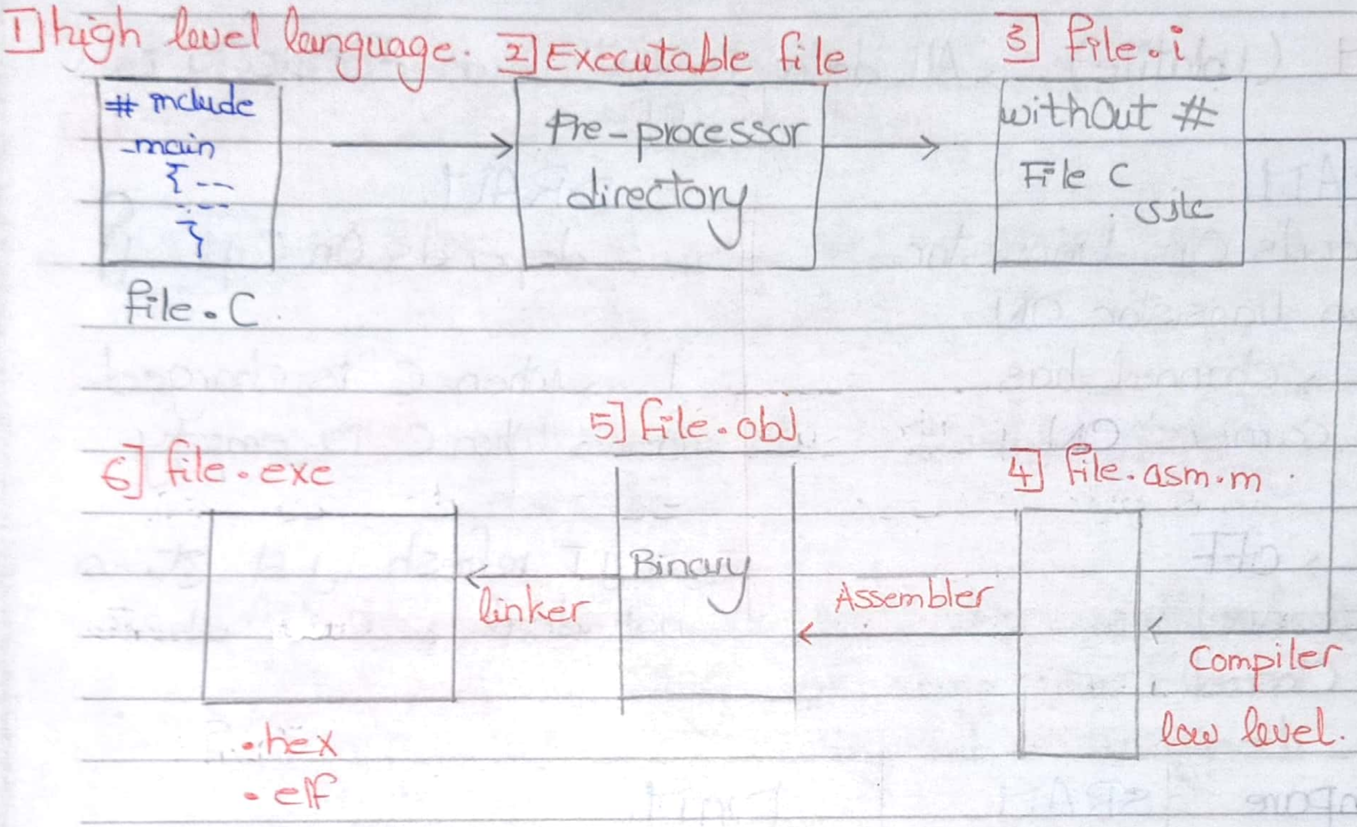
* ROM → Non Volatile (Floating) → Flash
→ ep^2 ROM.

Bus → wires. ① data bus < read / write

② Address bus

③ Control bus < 0 / 1

Build processor



Notes:. in the pre procssor stage, all # include, #define, #if (any hash "#") is removed and replaced by this file.h in the Code. except #progma

So file.i ⟶ has no #

* profiler ⟶ to get information about Code ⟶ size, __ *

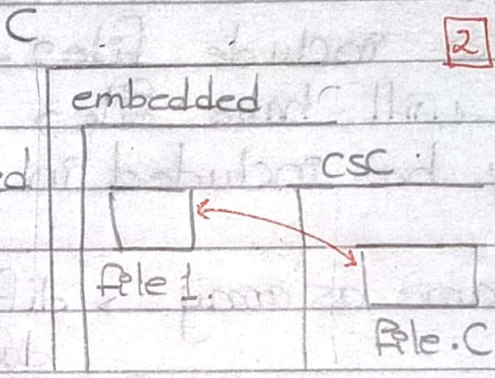
* optimizer ⟶ optimize the Size of mem by reducing unusable lines.

Linker ⟶ it manges all functions in all file in Final File

Types of error ① preprocessor error
② Runtime
③ logical error

* There is two way to include file :-

① Absolute



#include "C://embedded //csc// file.C"

هكتب كل حاجة لحد ما اوصل لل file.

[2] Relative.

#include "csc// file.C"

هكتب مكانه بالنسبة relative

↓ the Other file.

Note the difference :

define Max 10

we left space in Variables.

define Sum(x,y)

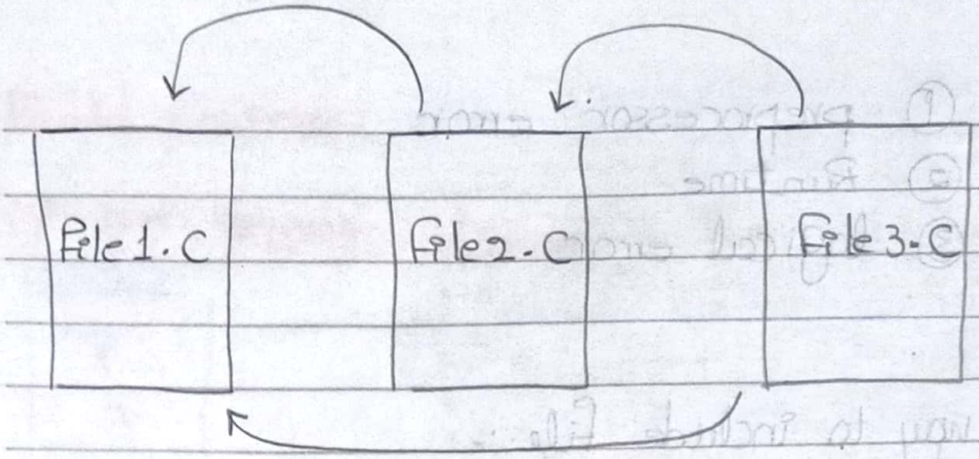
we left no space in function

Note

```
# define MC 1

# if (MC == 1) ____
# elif (MC == 2) ____
# else ____
```

Note : MC 1
So the last two lines will be deleted.
But if # define MC 2
the first line won't be deleted



* If we first include file3.C in file1 & file 2 then we want to include file2.C in file1.
So File 1.C will have file3 double.
(As its already be included in file 2)

struct : (same as array → diff data type بس الحجز)

struct student {
int ID ;
char name [20] ; ← different data type لـ
float degree ;
}

يحجز مكان

Enum
# define Red 0
# define Blue 1
# define Black 2

$\xrightarrow{\text{Equivlent}}$

Enum Coulor {
Red
Blue
Black
}

هو هيحجز لـ Red و الـ Blue 1 والـ Black
0 2

بدل ماكتب سطر اعلاها define.