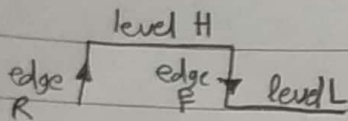


Clock CycleInstruction Cycle

في نفس السايكل
 Fetch
 Decode
 execute
 write back

PIC 16F877A

RISC 35

Flash ^{Loc.} 8K * ^{WZ} 14

EEPROM 256

SRAM 368 * 8

PIC 18F4550 16384 code density in flash (عدد التعليمات التي يمكن تخزينها في الذاكرة) ^{يقدر}

CISC 75+8

Flash 32K

EEPROM 256

SRAM 2K

STM 32F103C8T6

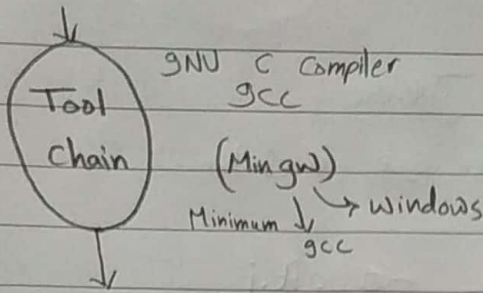
AKM ARM → Document for instruction set

RISC

Flash 64K

SRAM 70K

Source File File.c



Executable File File.hex

.exe

.elf

.crt

.lib

File.c "Source File"

Preprocessor

File.i

"Intermediate File"

Compiler

File.asm

"Assembly File"

Assembler

File.obj

"Object File"

Linker

File.hex

.exe

"Executable File"

Preprocessor Directives

include

define

undef

if

else

elif

ifdef

ifndef

endif

error

warning

#Pragma

من تبع Preprocessor

Compiler, Assembler, and Linker are "Target specific"

Profiler →

Optimizer →
Compiler
Assembler
Linker

memory

execution cycle

execution cycle

Function

code

tool

cmd

gcc File.c -o out.exe

a.exe default name
لوگ ڈیفالٹ نام

gcc -E File.c -o out.i

gcc -S File.c -o out.asm

gcc -c File.c -o out.obj
small

cls → clear cmd کی اسکرین کو

Prompt ahmed:
ahmed;

gcc → Tool Chain
gcc -E →
gcc -S →
gcc -c →

make → makefile

Rule: Dependencies
name:

<tab> Command

Batch Script

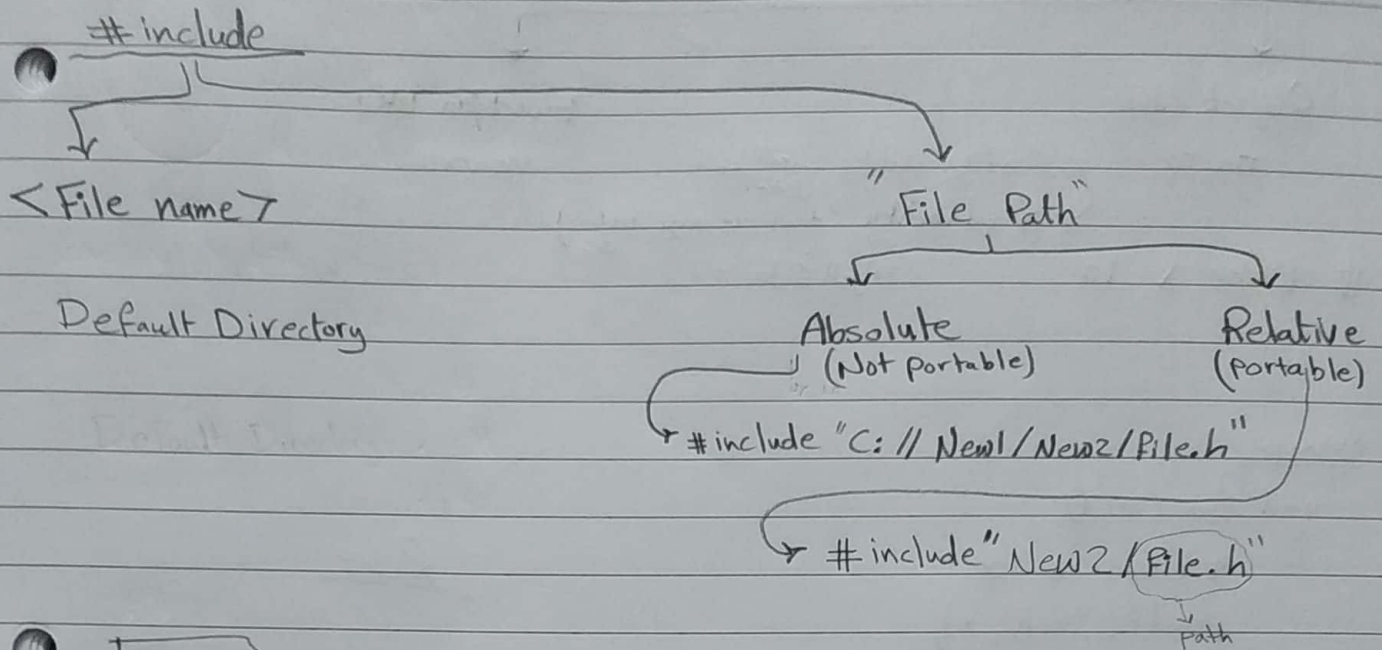
make Build example

بکٹ کائی

cmd کی makefile

Preprocessor

Preprocessor میں بیوقوف "C" کی طرح error کو ال File میں سے
میں موجود



File 5.h

1-Abs: #include "C://New1/New4/New5/File5.h"

2-Rel: #include "News/Files.h"

File 4.h

1-Abs: #include "C://New1/New4/File4.h"

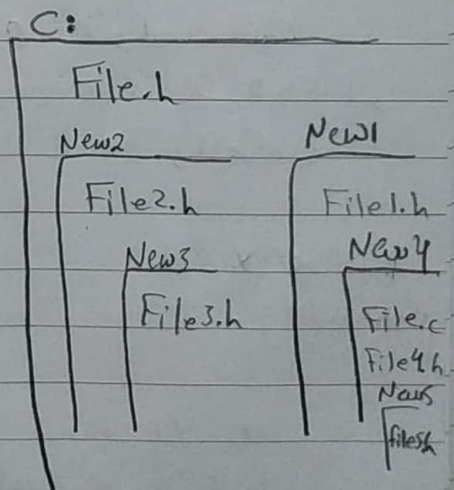
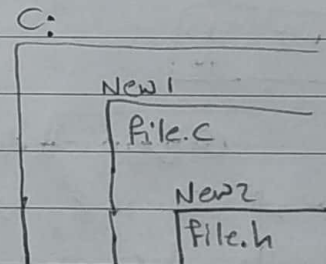
2-Rel: #include "File 4.h"

File 1.h

Rel: #include "../File1.h"

File.h

Rel: #include "../../File.h"



#define "Macro" #define like macro also

Object like

macro

→ Configurability (دالة التغير والتعديل)
→ Readability (don't use magic numbers) (سهولة القراءة)

Function like

macro

#define x 10

→ سهولة القراءة

example

void func(void)

{ #define x 10

printf("%d", x);

}

main {

printf("%d", x);

}

result	10
	10

Rules

٢- لو عندي multi macro هيفتاك
في ال expansion من في ال name

rule ال #define معرفتي

يعني رايه scope لكن هو بيشتوي
كل اللي بعده وبيحل replace.

٣- ال Preprocessor معرفتي Scope
ويفطبق من اول ما اتكتب

ال Preprocessor Case sensitive

بيحل replace لل small لو

small letter والعكس

#define x 1

main()

{

int yx=10;

printf("%d", y1);

}

result	compilation error
--------	-------------------

ال macro لا يفهم بين ال برقي ولا space ولا dot

```
#define x 10
```

```
main  
{
```

```
#define x 20
```

```
printf("%d", x);
```

```
}
```

update defined
x to 20

result 20

Warning

الـ macro يسكن من جزئيت

```
#define
```

X
name

10

expansion

↑
multimacro

```
#define x 10
```

```
#define y x
```

```
main()
```

```
{
```

```
printf("%d", y);
```

```
}
```

result 10

```
#define x z
```

```
#define z 10
```

```
main()
```

```
{
```

```
printf("%d", x);
```

```
}
```

result 10

```
#define x z
```

```
#define z x
```

close circuit macro

```
main()
```

```
{ printf("%d", x);
```

```
printf("%d", z);
```

```
}
```