**Mã hóa chương trình**

**Xem 3 video trong series Easy:**

* “Bộ nhớ nội ROM, RAM”
* “Tính toán tốc độ thực thi”
* “Tính dung lượng bộ nhớ, xác định Machine code của lệnh”

**Bài tập:**

1- Tính dung lượng bộ nhớ cho chương trình trong bài tập 1 (tuần trước)

2- Xác định mã máy (machine code) cho từng lệnh, và cho toàn chương trình trong bài tập 1

3- Tính thời gian thực thi của từng lệnh và của toàn chương trình trong bài tập 1

**[Lập bảng chung cho 3 câu]**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Lệnh ASM (bạn viết) | Lệnh ASM (compiler ánh xạ)  (xem Disassembly) | Dung lượng(Byte) | Kiểu toán hạng | Machine code (Encoding) | Machine Cycle (us) |
| ORG 000h | x | x | x | x | x |
| LOOP | x | x | x | x | x |
| MOV A, P3 | MOV A, P3 | 2 | MOV A, direct | E5B0  11100101 direct | 1 (1us) |
| ANL A, #0Fh | ANL A, #0x0F | 2 | ANL A, immediate | 540F   |  |  |  | | --- | --- | --- | | 01010100 |  | immediate | | 1 (1us) |
| MOV R2, A | MOV R2, A | 2 | MOV Rn, A | FA  11111nnn | 1 (1us) |
| MOV A, P3 | MOV A, P3 | 2 | MOV A, direct | E5B0  11100101 direct | 1 (1us) |
| SWAP A | SWAP A | 1 | SWAP A | C4  11000100 | 1 (1us) |
| ANL A, #0Fh | ANL A, #0x0F | 2 | ANL A, immediate | 540F   |  |  |  | | --- | --- | --- | | 01010100 |  | immediate | | 1 (1us) |
| ADD A, R2 | ADD A, R2 | 1 | ADD A, Rn | 2A  00101nnn | 1 (1us) |
| MOV P1, A | MOV P1 | 2 | MOV direct, A | F590   |  |  |  | | --- | --- | --- | | 11110101 |  | direct | | 1 (1us) |
| JMP LOOP | SJMP LOOP(C:0000) | 2 | SJMP LABEL | 80EA  10000000 offset | 2  (2us) |
| END | x | x | x | x | x |

4- Bài (Không bắt buộc), cộng điểm nếu "tự làm & giải thích được, không google/chatgpt/..."

+ Sửa hàm Delay1S cho chính xác 1S

+ Viết chương trình blink Led, T= 2s (1s sáng, 1s tắt)

(\*)Gợi ý: Xem đáp án (gần như có sẵn) trong chế độ Debug.  
(Nếu bạn muốn tập tư duy, thì tự mở file tài liệu và tự tính đáp án. Sau đó hãy mở Debug để xem lại kq đúng không!)

5- Bài (không bắt buộc), phân tích ý nghĩa của file hex. (file bin, file elf)

(\*)Đoạn chương trình DELAY1S:

DELAY1S:

MOV R3,#5

S1BACK:

CALL DELAY200ms

DJNZ R3, S1BACK

RET                ; ket thuc c.trinh con.

DELAY200ms:

MOV R2, #200

LAP2:        MOV R1, #200

LAP1:        NOP                 ;lenh delay 1MC

NOP

NOP

DJNZ R1, LAP1         ;DECREASE, jump if not zero

DJNZ R2, LAP2

RET