

System Programming

실행결과, Loc, opcode, Modification record,

Assignment: Assembler & Linker Practice

```
QUIZ    START    0
        EXTREF   INIT, FUNCA, FUNCB

FIRST   +JSUB    INIT
        LDX      =X'000000'
        +LDT     #MAXLEN
ILOOP   LDCH     STRIN,X
        +JSUB    FUNCA
        TIXR     T
        JLT      ILOOP
        LDX      =X'000000'
OLOOP   +JSUB    FUNCB
        ADD      #32
        STCH     STROUT,X
        TIXR     T
        JLT      OLOOP
HALT    J        HALT
        LTORG
STRIN   BYTE     C'STRESSED'
MAXLEN  EQU      8
STROUT  RESB     MAXLEN
        END      QUIZ
```

그림 1 Quiz.asm

```
LIBS    START    0
        EXTDEF   INIT, FUNCA, FUNCB

INIT    LDA      #BUFFER
        STA      PTR
        RSUB

FUNCA    STCH     @PTR
        LDA      PTR
        ADD      #1
        STA      PTR
        RSUB

FUNCB    LDA      PTR
        SUB      =X'000001'
        STA      PTR
        LDCH     @PTR
        RSUB

PTR      RESW     1
BUFFER   RESW     100
        END      LIBS
```

그림 2 LIBS.asm

1. 과제 설명

다음은 Quiz 2에 출제되었던 코드이다. 주어진 코드들을 SICTOOL의 Assembler 기능을 활용하여 오브젝트 코드(.obj)로 변환한 뒤, 생성된 오브젝트 파일들을 Linker를 통해 연결하여 실행 가능한 프로그램을 완성하고, 그 결과를 확인하시오.

2. 요구 사항

1. SICTOOL의 Assembler 기능을 사용하여 각 코드 파일을 .obj 파일로 생성할 것
2. Linker를 이용해 두 개 이상의 오브젝트 파일을 연결하여 최종 실행 파일을 생성할 것
3. 프로그램 실행 결과를 스크린샷으로 캡처하여 보고서에 포함할 것

3. 제출물 안내

- 제출 마감: 4월 18일(금요일) 23:59까지
- 제출 형식: 하나의 압축 파일로 제출
 - 압축 파일 이름: 자신의 학번.zip
 - 압축 파일 내 구성:
 - .obj 파일 3개: QUIZ.obj, LIBS.obj, ANSWER.obj
 - 보고서(PDF): 학번_이름.pdf 형식
 - 예시: 202224208_김진수.pdf

※ 파일명 및 제출 형식을 지키지 않을 경우 감점 처리될 수 있습니다.

4. 보고서 작성 가이드

- 형식: 자유 형식 (PDF, A4 기준 2페이지 이내)
 - 내용 포함:
 - 실행 과정 및 결과 화면(스크린샷)
 - 코드 구조 분석 및 동작 설명

How to run sic assembler

1. 오브젝트 파일 생성 (Assembler 실행)

터미널에 아래 명령어를 입력하여 각 소스코드를 .obj 파일로 변환:

```
java -cp sictools.jar sic.Asm 소스코드1 소스코드2
```

```
gvm:~/SicTools$ java -cp out/make/sictools.jar sic.Asm ~/QUIZ.asm ~/LIBS.asm
```

- 입력 시, 각 소스코드에 해당하는 .obj 파일이 생성됨.
- cat 명령어를 사용하여 생성된 .obj 파일의 내용 확인 가능.

```
User name :~$ cat QUIZ.obj
HQUIZ 00000000003F
RFUNCA FUNCB INIT
T0000001E4B1000000720257510000853A0214B100000B8503B2FF40720124B100000
T00001E1919002057A013B8503B2FF13F2FFD00000005354524553534544
M00000105+INIT
M000000805
M00000F05+FUNCA
M00001B05+FUNCB
E000000
```

2. 오브젝트 파일 연결 (Linker 실행)

두 개 이상의 오브젝트 파일을 하나로 링크하기 위해 다음 명령어를 입력:

```
java -cp sictools.jar sic.Link -o 결과파일이름 오브젝트파일1 오브젝트파일2
```

```
gvm:~/SicTools$ java -cp out/make/sictools.jar sic.Link -o ANSWER.obj ~/QUIZ.obj ~/LIBS.obj
```

- 입력 시, 링크가 완료된 하나의 .obj 파일이 생성됨.
- 역시 cat 명령어로 파일 내용 확인 가능.

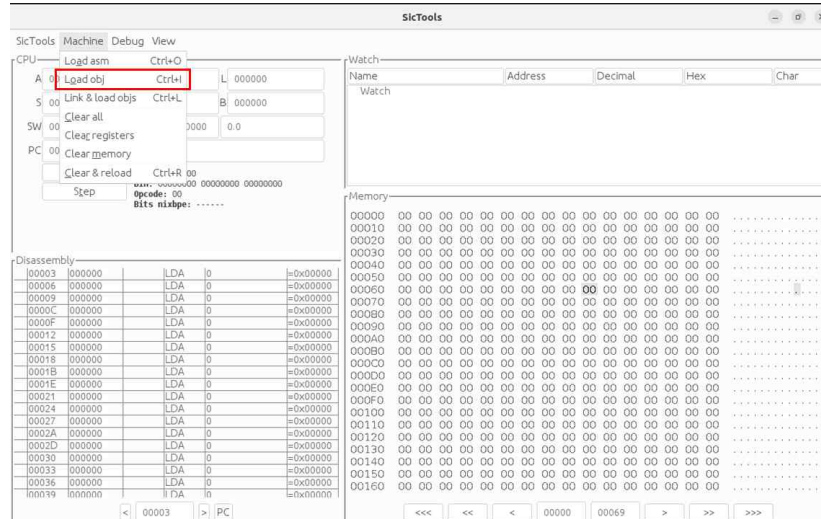
```
User name :~/SicTools$ cat ANSWER.obj
HANSWER000000000198
T0000001E4B10003F0720257510000853A0214B100048B8503B2FF40720124B100057
T00001E1919002057A013B8503B2FF13F2FFD00000005354524553534544
T00003F1E0120270F20214F0000056201B0320181900010F20124F0000003200C1F2138
T00005D090F20065220034F0000
T00019503000001
M00000105
M000000805
M00000F05
M00001B05
E000000
```

3. SicTool 실행 및 프로그램 로드

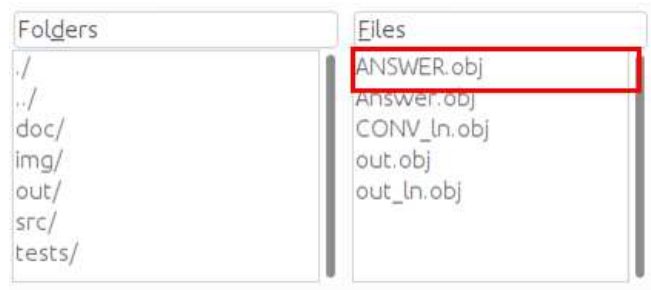
1. SicTool을 실행.

```
@vm: ~/SicTools$ java -jar out/make/sictools.jar
```

2. 상단 메뉴에서 Machine → Load obj 를 클릭.



3. 앞에서 생성한 링크된 오브젝트 파일(ANSWER.obj 등) 을 선택.



4. 좌측의 Disassembly 창과 우측의 Memory 창에 값이 정상적으로 로드되면 성공.

