

unit 4

by Chandra Kant

Submission date: 18-Jun-2018 02:16AM (UTC-0500)

Submission ID: 976633742

File name: Unit 4- Individual Project.docx

Word count: 587

Character count: 4216

Unit 4 Individual Project

Chandra Kant

CE242-1802B

CTU

Table of Contents

Business Organizational Background	3
Modern Computer Organizations and Design	4
Digital Logic Design	5
Computer Memory & Data Representation	6
Assembly Language Programming	7
C Program Compiler design	7
Four Stages of C Program Compilation	7
The Role of Assembly Language in C Program Compilation	7
Example of Assembly Language	7
Execution of Compiled Program	8
Input/ Output Organization	9
Summary & Final Recommendation	10

Business Organizational Background

TBD

Modern Computer Organizations and Design

TBD

Digital Logic Design

TBD

Computer Memory & Data Representation

TBD

Assembly Language Programming

Computers only understand machine language which is in binary form, consisting of 1's and 0's. Since it would be very hard to write a program in binary form, computers are programmed in assembly language or high-level programming languages. High-level programming languages are human-readable and therefore easy to write. Examples of such languages are Java, C, and FORTRAN. On the other hand, assembly language is closer to machine language but instead of numbers; names are used. Unlike high-level languages, assembly language is machine/processor specific and not portable. Assembly language comes in handy in cases of special features and where high execution speed or less memory use is required.

C Program Compiler design

Four Stages of C Program Compilation

- i) Preprocessing – In this stage, processing of lines of code to strip comments and join lines separated by '\n' is done. Lines starting with # are then processed as preprocessor directives.
- ii) Compilation – This is the second stage where the translation of preprocessed lines of code to assembly language is carried out. Different assemblers can be used to translate the code to assembly language targeted to a particular processor. Some compilers use integrated assemblers which translate the code directly to machine language.
- iii) Assembly – This is the stage where assembly language is translated to machine language.
- iv) Linking – In this final stage, a linker arranges machine code to enable successful linking of function calls and library functions. An executable program is now compiled.

The Role of Assembly Language in C Program Compilation

In a C program compiler, assembly language acts as an intermediate between the written source code and machine-readable language. Assemblers are written in assembly language for this purpose. They can either translate preprocessed source code to assembly language or in some cases directly to machine/object code.

Example of Assembly Language

Here is an example of a basic assembly language program to print 'Hello there.'

```

2
section      .text
global _start
_start:      ; entry point
    mov     edx, len      ; text length
    mov     ecx, msg      ; text to be written
    mov     ebx, 1        ; stdout
    mov     eax, 4        ; sys_write
    int     0x80          ; calling kernel

    mov     eax, 1        ; sys_exit
    int     0x80          ; calling kernel

section      .data
msg db 'Hello there', 0xa ; the string to show
len equ $ - msg          ; string length

```

Execution of Compiled Program

Execution of executable object code is carried out by a loader. The work of the loader is to load the object code to memory then run it, starting from the first instruction. Object files can be in 3 possible forms, either as a relocatable, executable or shared object file. Relocatable object files can be combined to an executable object file. Shared object files are loaded into memory and linked at runtime while executable object files are loaded and executed directly.

Executable object files are generated by the linker and then loaded into memory and run by the loader.

Input/ Output Organization

TBD

Summary & Final Recommendation

TBD

unit 4

ORIGINALITY REPORT

16%	7%	1%	15%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Colorado Technical University Online Student Paper	9%
2	docplayer.hu Internet Source	5%
3	dev-archive.aminer.org Internet Source	1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		