tion

tation

ned

ch- 10

. Array এর declare,

1D ~~2D~~

Addressing mode syntax

কেমন? ~~addressin~~(Syntax)

কোনো code দরকার

90.2|10.2

10.2-a

Bin multiplication

Ch-9 (v.v.i)

9.1 - MUL and IMUN

Effect of MUL/IMUL on status flag:

SF, ZF, HF, PF undefined

CF/OF:
After . .

Ex - 9.1

9.3 - Divo

. Arr

1D

Addr

[illegible]

[illegible]

9.0.2

10. 2

Fig 7.2 (136)

→ Left shif করলে carry flag এর value কী হবে ?

Ex - 7.7 :

Ex - 7.8 :

Application of LS is multiplication

2 → 10 .
100 → 4 (2×2)
1000 → 8 (2×2×2)

→ SHL, SAL same
7.2.2 –
→ RSI
Ex 7. (1.36) Divisi–

Fig 7.5 (Rotate left)

Rotate
7.6 (page 140)

Fig 7.7 RCL

~~F~~ Fig 7.8 –

Fig 7.4 → Binary

8 → parition (Fir bes

9 → Optimal phase
replacement alg

10 → least recently
used

11 → RSA algorit
(encryption)

6 → Bounded Buff

7 → Counting Ser

linux command –

Ch – 8

8.1 (152)

Fig 8.1
POP POPF

8.2 Stack application →
→ ~~inputt~~ input reverse
→ ~~8.4~~ –
8.4 → call and Ret

rder
rce and
ieve
to
al block
d to mapped
cal block

sponsible
ds to
fetch

e code
access
id

age

printing
108

Jumps

r register
re

Example-6.1 (V.V.I) এবার ques

6.3 - Unconditional jump

6.4.1 (.112)

Example-6.2 (V.V.I

Ex-6.6

Ex-6.8 (For)

Ex-6.9 (While loop)

chapter summery

Ch-7

Logic, Shif and Rotate

$\dot{A} \oplus B = \bar{A}B + \bar{A}B$

7.1.1 → Destination

Ex

From these

1. The AND can be
   clear → and 0
   preserve → and 1

2. OR

3. XOR V.V.I

Ex-7.2

Ex-7.4

→ Clearing a Register (134)

→ Testing a Register for zer

CMP CX, D

→ NO

→ 7.2.1 - V.V.I

Shift left

(math) (135)

$$\begin{array}{l} X(0) \\ X(1) \\ X(2) \\ X(3) \\ X(4) = \\ X(5) \\ X(6) \\ X(7) \end{array} \begin{array}{|llll} W_8^0 & W & W & W \\ W_8^0 & W & W & W \\ W_8^0 & W & W & W \\ W_8^0 & W & W & W \\ W_8^0 & W & W & W \\ W_8^0 & W & W & W \\ W_8^0 & W & W & W \\ W_8^0 & W & W & W \end{array}$$

Fig 7.2 (.136)

→Left shif করলে carry flag এর value কী হবে ?

Ex-7.7:

Ex-7.8:

Application of LS in multiplication

2 → 10

100 → 4 (2×2)

1000 → 8 (2×2×2)

→SHL, SAL same

7.2.2-

→RSI

Ex7.(1.36) Divisi--

Fig 7.5 (Rotate left)

Rotate
7.6 (page 140)

Fig 7.7 RCL

Fig 7.8-

Fig 7.4 → Binary

8-
9
10
1

Ch-
8.1
Fig 8
pop
8.2
8.4

— x —

20-08-23 Assembly Language

Ch-6

6.1 - Jump

6.2 - JNZ → Jump if not zero printloop

→ The cmp instruction pag 108
instruction গুলার কাজ

6.2. Conditional Jumps
Table 6.1 – Conditional Jupmps
signed, unsigned

→ Exercise চাওয়া হলো দুই টা register and value. কীভাবে compare করা যায়?

- Generally Jump, conditional এর statements গুলা.

Table - 4.2 v.v.I

→ emulator 8086 (download)

→ 4.5.2 (আগেরও ত3)

ADD, SUB, (Fig)

INC (x6)

Fig 4.7 (প্রত্যেক incremand negation)

→ 4.7.1 – কী কী মডেল কী ডিজাইন.

→ 4.7.2 –

→ 4.7.3

→ 4.7.4

→ 4.9 – First program (page 82)

· program Listing. PGM 4.

→ Fig - 4.8 (v.v.I)

→ Summeoy

Ch - 5

→ কেন আছে কেন নাই

→ flags 95, 96

→ 5.2 - Ex

· overflow - 4 রকম - কখন কোন overflow.

→ Ex - 5.1 (v.v.I (98) page)

→ Flag Register operation

Control Falg

— x —

16-08-23 - OS

File management

File Def - attribute -

- → Nam
- → type
- → technique
- → size

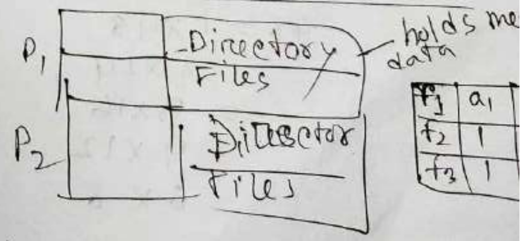
Operat

- → open
- → creation
- → ~~transead~~ / truncate
- → Reposition
- →

File access mode – 3 types

1. Sequential Accen
2. Direct Access
3. Indexed Access

Directing Structure



Directory

(i) single level

→ A ~~[illegible]~~

→ Assembly languag e pro (27)

Program Listing PGM1.

DW → Define word.

page (28)

Chapter summery (V.V.I)

Ch-2

→ Bin

→ Hex

→ 2.2 - V.V. I

→ 2.4

→ One's comple V.V.I

→ Two's 〃 (Ex)

— 2.8 Ex

→ Subracti

→ Summery

Ch-3

→ 3.1 -

→ Microprocess 8086 and 8088

page 51 (IT Govt. memory)

→ Bitwise, clamification

Generation wise

→ 3.2.1

→ 3.2.2

→ প্রতিটা Register এর particu

কাজ

→ সব কিছুর কাজ (53)

→ Ax

→ DX

* Segment - offset addrem

Ex - 3.2

Fig 3.3 এর, প্রত্যেক টা

কত্ত কোনটার কি-

Summery

Ch-4

→ 4.1 statements

name operation operand com

→ 4.1.3 (Field Desc.

→ 4.2

Number

Chars

variabu

Byte variabl

→ name DB

→ 4.3.2 (by)

→ 4.3.3

→ 4.4 (Equ).

→ Fig 4.1 MOV Ax

→ Exchng.

Ch - 1

→ Bytes and words

→ Bit position

→ Fig 1.4 - low byte-hi g

→ Fig 1.5
Operation of BUS

* Unidireectional Birectional

→* Fiig 1.6 - V.V.I
Details → Flag (Everythings)

→ machon lang and amsemb
langrag (Difference)