



44 D 4c L 20 Space 41 A 49 I 46 F 45 E 56 V
53 S 0a LF

Program 1

LDA FIVE
STA ALPHA
LDCH CHARZ
STCH C1

ALPHA	RESW	1
FIVE	WORD	5
CHARZ	BYTE	C'Z'
C1	RESB	1

```
manoj@manoj-VirtualBox:~/Desktop$ mkdir CSN-252
manoj@manoj-VirtualBox:~/Desktop$ cd CSN-252
manoj@manoj-VirtualBox:~/Desktop/CSN-252$ vi prog1.c
manoj@manoj-VirtualBox:~/Desktop/CSN-252$ od -x prog1.c
0000000 444c 2041 4946 4556 530a 4154 4120 504c
0000020 4148 4c0a 4344 2048 4843 5241 0a5a 5453
0000040 4843 4320 0a31 4c41 4850 2041 4552 5753
0000060 3120 460a 5649 2045 4f57 4452 3520 430a
0000100 4148 5a52 4220 5459 2045 2743 275a 430a
0000120 2031 4552 4253 3120 000a
0000131
manoj@manoj-VirtualBox:~/Desktop/CSN-252$
```

Program 1

```

MYPROG    START      1000
FIRST     LDA        FIVE
          STA        ALPHA
          LDCH       CHARZ
          STCH       C1
          RSUB
ALPHA     RESW        1
FIVE      WORD        5
CHARZ     BYTE       C'Z'
C1        RESB        1
          END        FIRST

```

START Specify **name** and **starting address** for the program

END Indicates the end of the program and (optionally)
specify the **first executable instruction in the program**

```

manoj@manoj-VirtualBox:~/Desktop$ mkdir CSN-252
manoj@manoj-VirtualBox:~/Desktop$ cd CSN-252
manoj@manoj-VirtualBox:~/Desktop/CSN-252$ vi prog1.c
manoj@manoj-VirtualBox:~/Desktop/CSN-252$ od -x prog1.c
00000000 444c 2041 4946 4556 530a 4154 4120 504c
00000020 4148 4c0a 4344 2048 4843 5241 0a5a 5453
00000040 4843 4320 0a31 4c41 4850 2041 4552 5753
00000060 3120 460a 5649 2045 4f57 4452 3520 430a
0000100 4148 5a52 4220 5459 2045 2743 275a 430a
0000120 2031 4552 4253 3120 000a
0000131
manoj@manoj-VirtualBox:~/Desktop/CSN-252$

```

↓
Assembler

```

0000 00000F
0003 0C000C
.
.
000C xxxxxx
000F 000005
.

```

Program 2

	LDA	ALPHA
	ADD	INCR
	SUB	ONE
	STA	BETA
	LDA	GAMMA
	ADD	INCR
	SUB	ONE
	STA	DELTA
	:	
ONE	WORD	1
ALPHA	RESW	1
BETA	RESW	1
GAMMA	RESW	1
DELTA	RESW	1
INCR	RESW	1

- avoid the need to fetch INCR from memory each time it is used in the calculation
- use of immediate addressing for the constant 1

Program 3 to copy one 11-byte character string to another

	LDX	ZERO
MOVCH	LDCH	STR1, X
	STCH	STR2, X
	TIX	ELEVEN
	JLT	MOVCH
	:	
STR1	BYTE	C'TEST STRING'
STR2	RESB	11
ZERO	WORD	0
ELEVEN	WORD	11

TIX: $X \leftarrow (X) + 1$; $(X) : (m..m+2)$ C (<, =, >)

Contents of memory Loc 0, 1 and 2 in prog 1

LDA FIVE

0000 0000 0000 0000 0000 1111

TA = 000F

(A) = 5

Contents of memory Loc 0, 1 and 2 in prog 3

STCH STR2, X

0101 0100 1

(X) = 0000

TA = 00

SIC/XE



- **Memory** – maximum memory available is 2^{20} bytes (1 megabyte)
- **Registers** (additional)

B	3	Base register
S	4	GPR
T	5	GPR
F	6	Floating-point accumulator

- Floating-point accumulator is 48 bit