2016	Sprin	29	
		(216)

1		THE STATE OF			The second of th
Line	Location	Symbol	opcode	101	
10	0000	Test	START	EXP	Objectude
20	0000	FIRST	LDA	40	
30	0003	8 - Y - Can	STX	#3	010003
40	0006			THREE	132013
SO	0009		LDX	=X'00'	072010
60	000D		+LDS	THREE	GF100016
70	000F		ADDR	A,X	9001
80	0013	RESULT	+STA	REJULIX	OFGOOD
90			RESW	SA SAN	_
100	0016	THREE	RESW		7 -
110	0019	110	END	FIRST	_
110	0019	X	= X,00,	and	00

Start

Location fill stof Start, end, base) exist

Here

O Directive (START, end, base) exist

Oddress occupy Grardaina

(i) stree opcode HI

Sury HI + (+LDS) or Lout HI R

(APDR) 3H36 profi its SIC(XE

(ii) SIC Simple HI Ard at instruction

3 Byte on gree But 4 Byte - SIC(XE

HI 1,2,3 or 4 byte instruction and

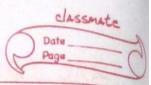
format for goal

(i) stree operand like having 't' in prefix

(LEMH) then its 4 byte (+LDS)

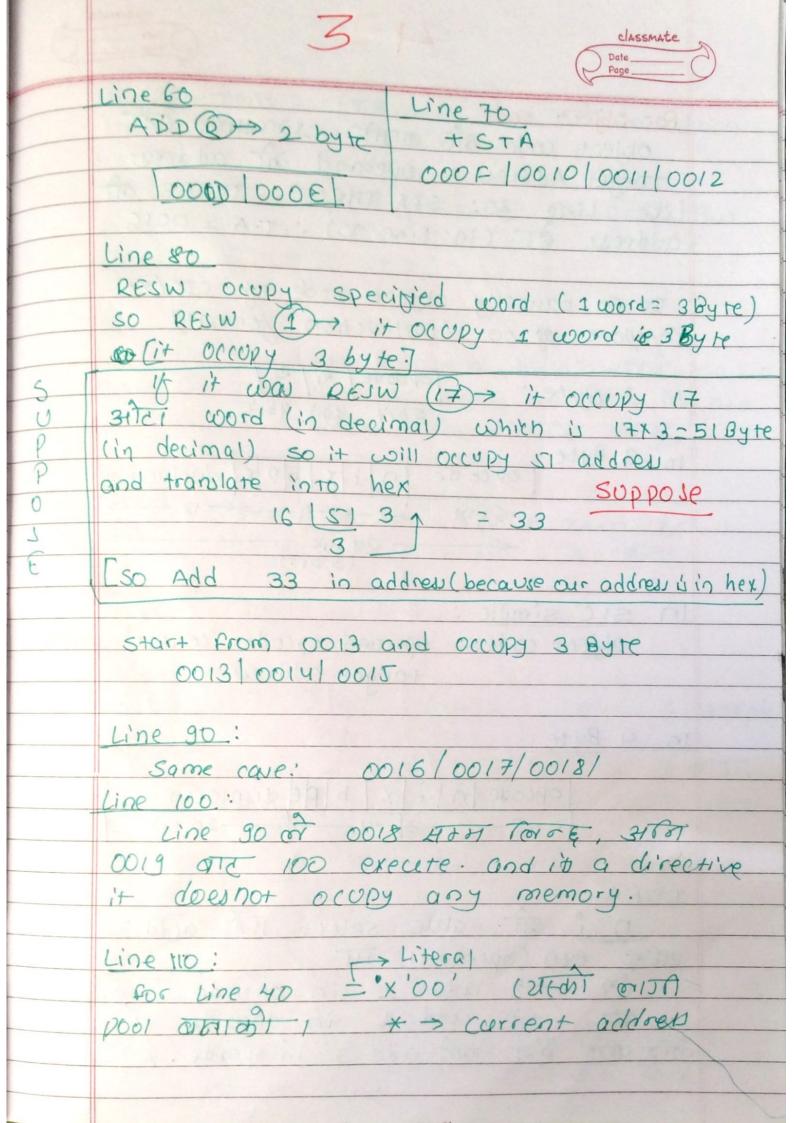
stree operand like having 'R' in suffix

(Last) ATT then its 2 byte 2 log of its 3 byte.

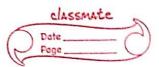


Line 10: its openand: START address start fre Location column Hi START is direction it does not take any address '0000' START of onthe Uto solvers address town so line 20 Start from 0000 But it operand LDA WHAT 't FOT EN 'R' FOT EN SO IT 3 byte Add it occupy 0000/0001/0002/ Line 20 of 0000, 0001, 0002, occupy stail So Line 30 Storts from 0003 at usa (STX) 3 byte it occupy 0003/0004/0005 Line 40 start from 0006 and occupy 3 by te opto 0008 0006 0007 0008 Line 50 10008 0003 000A 0000

0009/000A|000B|000C

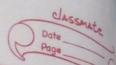


Page
For object code
Object code day out a exposure
Target Addrew = operand on addrew
like the Address = operand on address
address et (in Line 90): T.A = 0016
Compare at an libertain to start (XE
Compare of sic/xE
Compare of on (instruction format)
In 2 Byte: OPcode R1/82
861+ 461+ 461+
In 3 Byte
opcode nix bpe d'upper
$6614 + 42 \rightarrow 44 \rightarrow 12$
24 bit (3 bit)
In sic simple:
Object code = operand opcode 'concat' with
target address
In 4 Byte
opcode n'i x b pe dimen
to the distriction of the distri
20 bit >
T.DEA:
O i day realist
zifce exp (premails select sint attent
- (a) Hodala) HI
add 2 in
add 1 in opwde Trule
ale ala sin opwde / 1
m opace.

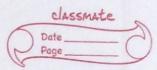


	Fage
	displacement = (TA - PC) 31/47 instruction
	of the lone step below)
	some instruction like (line 70) RESULT, X:
	some instruction like (line 70) RESULT, X:
	21 Fal - 11 E X = 1
	b P
	0 .0 -> 4 byte instruction (+ LDS) when b=0,p=0
	1 0 2 3 byte AT - CIFE SURU AT P=1, b=0
	0 1 7 21007, (1/ displa 400, 2100)
	0 -> 3 byte instruction on e zero & [-4]
	1 4 byte instruction on e zero gran
	1 - (solle (113) 110 (401) 20) C T E. E.
-7	Line 10
	directive (START) No object code
>	Line 20
	LDA #3 # (see rule 1) add I in opcode
	L> 00
	+1 6 2 5 th & 2000, direct 3 0015
	01 put sint as a displacement
	$(50 01 \times \times \times \times)$
	0100031 write 3 interno of 4 bit
	1
	Line 30 STX THREE (see rule 1) add 3
	b 10 x b p e
	7
	12 12 12 12 12 13 by to el, 4 by to
	Ear (no +, no R)





disp = TA (see address of THREE pe (see line 40's address Ly I step below's address disp = 0016 -0003 ob 13] -> Take last 3 digit for 3 Byte instruction (you Recause NO 4, NO R. so object code: 132013 Line 40 +LDS =x'00' -> its literal because of '=' 1>60 x b p e | = x'00' ont 65 000 01 address on TEZ FIT @ 1 वनाकी ह, 110 मा C+3=D+2=E+1=f disp = TA 0019 6F2010
-PC -0009 WRONG
00007 its 4 Byte No PC, NO Base So X=0, b,p=0,0 and c=1 disp = Target Address of 20 bit (5 Bicharacter) 00019
5 21 zero add gareko saci dall [6F100019]

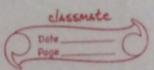


Ly 2x'00' > its literal became of '02' Ly 04 x b pe = x'00' of addrew +3 001.0 Last AT 6, 110 HI 07 2 disp=TA 0019 PC 0009 072010 Line 50 +LDS THREE its 4 byte so, b2p=0 +3 0 0 0 1 (x=1 ante Thece, x yild) 6F 1 may: disp= Target address of THREE is 0016 but its 4 byte, we need so byte of displacement [see page 4 (4 byte instruction format] so mare 0016 of 5 bit by adding one 0. -> 00016 So Object code 6F100016 Line 60: ADDR AIX (P) > 2 byte 100016 Line 60: ADDR AIX (P) > 2 byte 100016	Line 40
#3 0 0 1 0 LOUT HI E, 110 HI OF 2 disp=TA 0019 PC 0009 072010 Line SD. +LDS THREE its 4 byte so, b2p=0 \$\frac{1}{3}\$ 0 0 0 1 (\frac{1}{2} - \frac{1}{2} \text{THECE} \text{X Med}) \\ \text{GF 1} \text{300} \text{disp=Target address of THREE ie 0016 \\ but its 4 byte, we need so byte of displacement \\ \text{Esee page 4 (4 byte instruction format]} \\ \text{SO make 0016 of 5 bit by adding one 0.} \\ \text{20016} \\ \text{SO make 0016 of 5 bit by adding one 0.} \\ \text{20016} \\ \text{20016} \\ \text{20016} \\ \text{200016} \\ \te	1DX = x'00' = :+ 13000'
DISP = TA 0019 PC 0009	12 Oct 1 113 (literal perame of '02'
DISP = TA 0019 PC 0009	+2 pe = x'00' on address
Disp=TA 0019 PC 0009 072010 Line SD. +LDS THREE it 4 byte SO, blp=0 b 6C x b pe and e=1, 00 x=0 +3 0 0 0 1 (x=1 wife three, x min) GF 1 min) disp= Target address of three ie 0016 but its 4 byte, we need 20 byte of displacement [see page 4 (4 byte Instruction format] So mare 0016 of 5 bit by adding one 0. -> 00016 So Object code 6F100016 Line 60: ADDR AIX R > 2 byte 10 poute R 1 4 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	
Line SO. Line S	2
Line SD. + LDS THREE it 4 byte so, blp=0 b 6C x b pe and e=1, no x=0 +3 0 0 0 1 (x=1 unto thece, x min) 6F 1 mu) disp= Target address of three ie 0016 but its 4 byte, we need 20 byte of displacement [see page 4 (4 byte Instruction Rormat] so make 0016 of 5 bit by adding one 0. -> 00016 So Object code 6F100016 Line60: ADDR AIX (D-> 2 byte 100016 AI	CO3P = 1A 0019
Line SD. + LDS THREE it 4 byte so, blp=0 b 6C x b pe and e=1, no x=0 +3 0 0 0 1 (x=1 unto thece, x min) 6F 1 mu) disp= Target address of three ie 0016 but its 4 byte, we need 20 byte of displacement [see page 4 (4 byte Instruction Rormat] so make 0016 of 5 bit by adding one 0. -> 00016 So Object code 6F100016 Line60: ADDR AIX (D-> 2 byte 100016 AI	PC 0,009 [072010]
Line SO. + LDS THREE it 4 byte so, blp=0 by 6C x b pe and e=1, anx=0 +3 0 0 0 1 (x=1 unter three, x yild) GF 1 mu) disp = Target address of three is 0016 but its 4 byte, we need so byte of displacement [see page 4 (4 byte instruction rormat] so make 0016 of 5 bit by adding one 0. -> 00016 So Object code 6F100016 Line60: ADDR AIX D-2 byte 100016 Line60: ADDR AIX D-2 byte 100016 Line60: ADDR AIX D-2 byte 100016	0[010] 072010
HLDS THREE Ly 6C X b pe and e=1, one x =0 13 0 0 0 1 (x=1 wife three, x yid) 6F 1 one) disp= Target address of three is 0016 but its 4 byte, we need 20 byte of displacement [see page 4 (4 byte Instruction Rormat] So make 0016 of 5 bit by adding one 0. -> 00016 So Object code 6F100016 Line60: ADDR AIX R>2 byte crude R, R, Ly 90 40 61 Aco 90 01	
t3 0 0 0 1 (X=1 wife THREE, X MIN) GF 1 2 2000 disp= Target address of THREE is 0016 but its 4 byte, we need 20 byte of displacement [see page 4 (4 byte instruction rormat] So make 0016 of 5 bit by adding one 0. Tool6 So Object code 6F100016 Line60: ADDR AIX R 2 byte 10pwte R, R, R, L, 90 40 b, 1 ACO 90 01	
t3 0 0 0 1 (X=1 wife THREE, X MIN) GF 1 2 2000 disp= Target address of THREE is 0016 but its 4 byte, we need 20 byte of displacement [see page 4 (4 byte instruction rormat] So make 0016 of 5 bit by adding one 0. Tool6 So Object code 6F100016 Line60: ADDR AIX R 2 byte 10pwte R, R, R, L, 90 40 b, 1 ACO 90 01	+ LDS THREE its 4 byte so, 62020
disp = Target address of THREE ie 0016 but its 4 byte, we need 20 byte of displacement [see page 4 (4 byte instruction rormat] So make 0016 of 5 bit by adding one 0. -> 00016 So Object code 6F100016 Line60: ADDR AIX R > 2 byte 100016 Line60: ADDR AIX R > 2 byte 100016 Line60: ADDR AIX R > 2 byte 100016 ADDR AIX R > 2 byte 100016 Line60: ADDR AIX R > 2 byte 100016 Line60: ADDR AIX R > 2 byte 100016 ADDR AIX R > 2 byte 100016 Line60: ADDR AIX R > 2 byte 100016 Line60: ADDR AIX R > 2 byte 100016 ADDR AIX R > 2 byte 100016 ADDR AIX R > 2 byte 100016	
disp = Target address of THREE ie 0016 but its 4 byte, we need 20 byte of displacement [See page 4 (4 byte Instruction Rormat] So make 0016 of 5 bit by adding one 0. -> 00016 So Object code 6F100016 Line60: ADDR AIX (P) - 2 byte 100016 ADDR AIX (P) - 2 byte 100016 Line60: ADDR AIX (P) - 2 byte 100016	+3 0 0 0 1 (X=1 wife THREE, X 2010)
disp = Target address of THREE ie 0016 but its 4 byte, we need 20 byte of displacement [See page 4 (4 byte instruction format] So make 0016 of 5 bit by adding one 0. -> 00016 So Object code [6F100016] Line60: ADDR AIX (R) -> 2 byte Optobe RI RI Line60: ADDR AIX (R) -> 2 byte ADDR AIX RI Line60: ADDR AIX (R) -> 2 byte ADDR AIX RI Line60: ADDR AIX (R) -> 2 byte ADDR AIX RI Line60: ADDR AIX (R) -> 2 byte ADDR AIX RI Line60: ADDR AIX (R) -> 2 byte ADDR AIX RI Line60: ADDR AIX (R) -> 2 byte ADDR AIX RI Line60: ADDR AIX (R) -> 2 byte ADDR AIX RI Line60: ADDR AIX (R) -> 2 byte ADDR AIX RI Line60: ADDR AIX (R) -> 2 byte ADDR AIX RI Line60: ADDR AIX (R) -> 2 byte ADDR AIX RI Line60: ADDR AIX (R) -> 2 byte ADDR AIX RI Line60: ADDR AIX (R) -> 2 byte ADDR AIX RI Line60: ADDR AIX (R) -> 2 byte ADDR AIX RI Line60: ADDR AIX (R) -> 2 byte ADDR AIX RI Line60: ADDR AIX (R) -> 2 byte ADDR AIX RI Line60: ADDR AIX (R) -> 2 byte ADDR AIX RI Line60: ADDR AIX (R) -> 2 byte ADDR AIX RI Line60: ADDR AIX (R) -> 2 byte ADDR AIX RI Line60: ADDR AIX (R) -> 2 byte ADDR AIX RI Line60: ADDR AIX (R) -> 2 byte ADDR AIX RI Line60: ADDR AIX (R) -> 2 byte ADDR AIX RI Line60: ADDR AIX (R) -> 2 byte ADDR AIX RI Line60: ADDR AIX (R) -> 2 byte ADDR AIX RI Line60: ADDR AIX (R) -> 2 byte ADDR AIX RI Line60: ADDR AIX (R) -> 2 byte ADDR AIX RI Line60: ADDR AIX (R) -> 2 byte ADDR AIX RI Line60: ADDR AIX (R) -> 2 byte ADDR AIX RI ADDR AIX (R) -> 2 byte ADDR AIX RI ADDR AIX (R) -> 2 byte ADDR AIX	
Line 60: ADDR AIX R > 2 byte Opwde RI RV Line 90 40 41 ACO 90 01	
Line 60: ADDR AIX R > 2 byte Opwde R R P P P P P P P P	disp = Target address of THREE ie 0016
So make 0016 of 5 bit by adding one 0. The constant of the bound of the by adding one 0. So object code 6F100016 Line 60: ADDR AIX R 2 byte 10pole R R R R LA 90 40 bit ADDR AIX 90 0 0 1 XEL	but its 4 byte, we need 20 byte of displacement
So make 0016 of 5 bit by adding one 0. \rightarrow 00016 So Object code [6F100016] Line60: ADDR AIX (P) \rightarrow 2 byte Optobe R, R, R, L, P, P, P, P, P, P, P	[see page 4 (4 byte Instruction Rormat]
So Object code $6F100016$ Line 60: ADDR AIX $R \rightarrow 2$ by the $ 0 $ whe $ 0 $ ADDR AIX $ 0 $ ADD	
Line 60: ADDR AIX $R \rightarrow 2$ by the lopuste R , R A	
Line 60: ADDR AIX $R \rightarrow 2$ by the lopuste R , R A	So Object code (6F100016)
ADDR AIX R > 2 byte / CPWde /RI/RI/ L7 90 40 L31 ACO 90 01	
XEL	Line 60:
XEL	ADDR AIX R -> 2 byte Optobe RI RI
	4 90 40 41 ACO 90 01
0 1 2 3 4 5 6 8 9 9 19007	
0(1) 2 3 4 8 6 8 9	A (8) L B S + F PC SW 19007/
	0/1/2345689

Classmate Page

	Line 70
1 2	tine 70 + STA RESULT, X Cit 4 byte 6=0, P=0, es
	12 00 X; p p e 10
	+0 14 00 1
	73 - 9 +0
	OF J
	'+' 2171 ord 4 byte 50
	+ 31 w orn 4 byte so, [displacement = TA & or 21 3 byte of
	-PC SITI ET
0-03	desplacement = TA XIZI (but so bute)
	Line so of out of
Telle	
	disp 200013 [0F90013]
	Line 80, 90, 100 and order [directive]
three east	The territory by to the second of the second
	Line 110 + conver x'00' into hep
. 10 3/3	Oh! its already in hex because x' is her
	so object code
	31 Est De Location done la code program done
	(i) Object trogram done
(111)	Data structure
	SYMTAB I IMILIA
	Test 0000 7 Location and
	First DODD
	RESULT 0013 1 Table old Page
	THREE 0016 1 477

(9)



LITTAB literal expression value length address = X'00' 00 1 0019 (iv) Object code file/program > Haprogram namen start wength (See to be page 1) 4 program name = Test Start Oddress = 0000 length=end-start (019+2) - 0000 [2 chai =x'00' of 0020-0000 L fore 50 add 1 0020 -> TA Start address , length , object codes ---La object code of start addrew (i) slength (too toicky) [see Nore | Book] length chai, (maxm) object code and digit 60 samma include um milxa. 36 = 2= 18 object code HA TEST, 000000, 000020 TA 000000 12 A 010003 A 132013 A 07 2010A 6F100016 A 3001 A 0F9000 TA 000019 A01 A 00 MA 00000 AA 05 MA 000000 A 05 En 000000