TEXAS INSTRUMENTS

Improving Man's Effectiveness Through Electronics

Model 990 Computer

Prototyping System Documentation

PART NO. 943380 -0001

Digital Systems Division

TEXAS INSTRUMENTS INCORPORATED

DATE 04/26/77

LIST OF MATERIAL

PART NUMBER LM0943[illegible]001 REV J

PRINT ITEM NUMBER	QUANTITY PER ASSEMBLY	UNIT OF ISSUE	DWG. SIZE	PART NUMBER	DESCRIPTION	VENDOR PART NUMBER
0001	00001.000	EA		0946247-9701	MNL,PROTO SYS UNPACK/INSTALL GUIDE-990	
0001A					THIS MANUAL IS THE	
0001B					UNPACKING & INSTALLATION	
0001C					GUIDE FOR THE 990 PROTO-	
0001D					TYPING SYSTEM(945202-XXXX).	
0001E					*	
0001F					THIS MANUAL SHOULD BE	
0001G					PACKED AS THE TOPMOST ITEM	
0001H					IN THE PROTOTYPING SYSTEM	
0001I					SOFTWARE BOX(943380-0001).	
0001J					*	
0001K					THE SOFTWARE BOX SHOULD BE	
0001L					LABELLED (ON TOP OF BOX)	
0001M					AS FOLLOWS:	
0001N					THE PROTOTYPING SYSTEM	
0001O					UNPACKING & INSTALLATION	
0001P					GUIDE IS INCLUDED IN THIS	
0001Q					BOX. BE SURE TO READ THIS	
0001R					MANUAL BEFORE ATTEMPTING	
0001S					TO INSTALL HARDWARE AND	

DRAFTSMAN / DATE	CKD. DRAFTSMAN / DATE	DESIGN ENGINEER / DATE	TITLE
	Larry R. Arrants 4-27-77		SP,PROTOTYPING SYS SOFTWARE(CCO)-SYS990

PD. ~~MFG.~~ / DATE	APPD. PROJECT ENGINEER / DATE	RELEASED / DATE	PROJECT NO.	PART NUMBER	REV
M.White 4-29-77	[illegible]	[illegible] 5/4/77	7506	LM0943380-0001	J

I 13849

PRINT ITEM NUMBER	QUANTITY PER ASSEMBLY	UNIT OF ISSUE	DWG. SIZE	PART NUMBER	DESCRIPTION	VENDOR PART NUMBER
0001T					SOFTWARE.	
0002	00001.000	EA		0945255-9701	MNL,990 PROTOTYPING SYSTEM USER'S GUIDE	
0003	00001.000	EA		0943441-9701	MNL,TMS9900 MICROPROC ASSEM LANG GUIDE	
0004	00001.000	EA		0945250-9701	MNL,990 COMPUTER FAMILY SYSTEMS HANDBOOK	
0005	00005.000	EA		0943440-9701	MANUAL,990 COMPUTER PROGRAMMING CARD	
0006	00002.000	EA		0943348-0001	MODEL 990/TMS9900 ASSY LANG CODING FORM	
0007	00001.000	EA		0945328-9901	PD,SAMPLE PROGRAM 1-PX990	
0008	00001.000	EA		0945329-9901	PD,SAMPLE PROGRAM 2-PX990	
0009	00001.000	EA		0943380-0019	SP,PROTOTYPING SYS SOFTWARE(AL)-SYS990	
0009A					REF INDIVIDUAL LISTS OF	
0009B					MATERIALS FOR CONTENTS OF	
0009C					ITEMS 10 THRU 12.	
0010	00001.000	EA		0943380-0021	SP,MASTER CASSETTE,PROTO SYS S/W-SYS990	
0011	00001.000	EA		0945328-0001	SP,SAMPLE PROGRAM 1,CASS SOURCE-PX990	
0012	00001.000	EA		0945329-0001	SP,SAMPLE PROGRAM 2,CASS SOURCE-PX990	
0013	00001.000	EA		0943350-2401	CCS,STANDARD CONTROL INFO FILE-PX990	
0014	00001.000	EA		0943388-0001	SP,CO-RES PROTO SYS ADD-ON(CCO)-SYS990	
0014A					THIS IS AN ADD-ON PACKAGE	
0014B					TO BE USED WITH SYSTEMS OF	
0014C					12K OR MORE MEMORY. THE	

PRINT ITEM NUMBER	QUANTITY PER ASSEMBLY	UNIT OF ISSUE	DWG. SIZE	PART NUMBER	DESCRIPTION	VENDOR PART NUMBER
0014D					STRUCTURE OF THE PACKAGE	
0014E					IS SIMILAR TO THAT OF THE	
0014F					8K SYSTEM PACKAGE(943380-1)	
0014G					*	
0014H					PEF LIST OF MATERIALS FOR	
0014I					CONTENTS OF 943388-0001.	
0015	00003.000	EA		0960333-0001	MAGNETIC TAPE CASSETTE	

DRAFTSMAN	DATE	CKD. DRAFTSMAN	DATE	DESIGN ENGINEER	DATE	TITLE
						SP,PROTOTYPING SYS SOFTWARE(CCC)-SYS990
PD.-MFG.	DATE	APPD. PROJECT ENGINEER	DATE	RELEASED	DATE	PROJECT NO.

PART NUMBER LM0943380-0001 REV J

TEXAS INSTRUMENTS CORPORATED

DATE 04/26/77

LIST OF MATERIAL

PART NUMBER LM0943[illegible]0021 REV J

PRINT ITEM NUMBER	QUANTITY PER ASSEMBLY	UNIT OF ISSUE	DWG. SIZE	PART NUMBER	DESCRIPTION	VENDOR PART NUMBER
0001	00001.000	EA		0943395-2201	CDS,PROLOGUE,MASTER CASS,PROTO SW-SYS990	
0001A					SIDE A,ITEM 1	
0001B					FOLLOW WITH 'END OF FILE'	
0002	00001.000	EA		0945392-1201	CDO,UPFLD,UP FRONT LOADER-PX990	
0002A					SIDE A,ITEM 2	
0002B					FOLLOW WITH 'END OF FILE'	
0003	00001.000	EA		0943518-1201	CDFL,PX9MTP,ROOT,PROTO SYS DBG MON-PX990	
0003A					SIDE A,ITEM 3	
0003B					ABS FORMAT:LOAD ADDR=>2000	
0003C					FOLLOW WITH 'END OF FILE'	
0004	00001.000	EA		0943518-1204	CDFL,LAL,OVERLAY,PROTO SYS DBG MON-PX990	
0004A					SIDE A,ITEM 4	
0004B					FOLLOW WITH 'END OF FILE'	
0005	00001.000	EA		0943518-1203	CDFL,TRACE,OVERLAY,PROTO DEBUG MON-PX990	
0005A					SIDE A,ITEM 5	
0005B					FOLLOW WITH 'END OF FILE'	
0006	00001.000	EA		0943518-1202	CDFL,ABS,OVERLAY,PROTO SYS DBG MON-PX990	
0006A					SIDE A,ITEM 6	
0006B					FOLLOW WITH 'END OF FILE'	
0007	00001.000	EA		0943518-1205	CDFL,PRPRG,OVERLAY,PROTO DEBUG MON-PX990	

DRAFTSMAN / DATE	CKD. DRAFTSMAN / DATE	DESIGN ENGINEER / DATE	TITLE
	Larry R. Arrants 4-27-77		SP,MASTER CASSETTE,PROTO SYS S/W-SYS990

PD.-MFG. / DATE	APPD. PROJECT ENGINEER / DATE	RELEASED / DATE	PROJECT NO.	PART NUMBER	REV
[illegible] 4-29-77	[illegible] 4/27/77	Larry [illegible] 5/4/77	7506	LM0943380-0021	J

1 13849

PRINT ITEM NUMBER	QUANTITY PER ASSEMBLY	UNIT OF ISSUE	DWG. SIZE	PART NUMBER	DESCRIPTION	VENDOR PART NUMBER
0007A					SIDE A,ITEM 7	
0007B					FOLLOW WITH 'END OF FILE'	
0008	00001.000	EA		0945396-1201	CDO,PROMHP,PROM PROGRAMMER HELPER-PX990	
0008A					SIDE A,ITEM 8	
0008B					FOLLOW WITH 'END OF FILE'	
0009	00001.000	EA		0943518-1206	CDFL,BNPF,OVERLAY,PROTO DEBUG MON-PX990	
0009A					SIDE A,ITEM 9	
0009B					FOLLOW WITH 'END OF FILE'	
0010	00001.000	EA		0943518-1207	CDFL,HILO,OVERLAY,PROTO DEBUG MON-PX990	
0010A					SIDE A,ITEM 10	
0010B					FOLLOW WITH 'END OF FILE'	
0011	00001.000	EA		0945392-1201	CDO,UPFLD,UP FRONT LOADER-PX990	
0011A					SIDE A,ITEM 11	
0011B					FOLLOW WITH 'END OF FILE'	
0012	00001.000	EA		0945394-1201	CDO,PX9EDT,TEXT EDITOR-PX990	
0012A					SIDE A,ITEM 12	
0012B					ABS FORMAT:LOAD ADDR=>00A0	
0012C					FOLLOW WITH 'END OF FILE'	
0013	00001.000	EA		0945392-1201	CDO,UPFLD,UP FRONT LOADER-PX990	
0013A					SIDE A,ITEM 13	

PRINT ITEM NUMBER	QUANTITY PER ASSEMBLY	UNIT OF ISSUE	DWG. SIZE	PART NUMBER	DESCRIPTION	VENDOR PART NUMBER
0013B					FOLLOW WITH 'END OF FILE'	
0014	00001.000	EA		0945393-1201	CDO,PX9ASM,ASSEMBLER-PX990	
0014A					SIDE A,ITEM 14	
0014B					ABS FORMAT:LOAD ADDR=>0000	
0014C					FOLLOW WITH 'END OF FILE'	
0015	00001.000	EA		0943518-1201	CDFL,PX9MTP,ROOT,PROTO SYS DBG MON-PX990	
0015A					SIDE A,ITEM 15	
0015B					FOLLOW WITH 'END OF FILE'	
0015C					LABEL SIDE A AS FOLLOWS:	
0015D					P/N 943380-0021: CONTENTS=	
0015E					PX9MTP+OVERLAYS,PX9EDT,	
0015F					PX9ASM REF MANUAL	
0015G					945255-9701,SECT 2 FOR	
0015H					LOADING INSTRUCTIONS	

AFTSMAN DATE	CKD. DRAFTSMAN DATE	DESIGN ENGINEER DATE	TITLE
			SP,MASTER CASSETTE,PROTO SYS S/W-SYS990
PD. MFG. DATE	APPD. PROJECT ENGINEER DATE	RELEASED DATE	PROJECT NO. / PART NUMBER / REV
			LM0943380-0021 J

TEXAS INSTRUMENTS
INCORPORATED

DATE 05/22/76

LIST OF MATERIAL

PART NUMBER LM 945328-0001 REV A

PRINT ITEM NUMBER	QUANTITY PER ASSEMBLY	UNIT OF ISSUE	DWG. SIZE	PART NUMBER	DESCRIPTION	VENDOR PART NUMBER
C001	00001.000	EA		945328-2201	CDS,TDRIVE,SAMPLE PROGRAM 1-PX990	
C001A					CASSETTE SIDE A,ITEM 1	
C001B					FOLLOW WITH 'END OF FILE'	
0002	00001.000	EA		945328-2202	CDS,TSKIDT,SAMPLE PROGRAM 1-PX990	
C002A					CASSETTE SIDE A,ITEM 2	
C002B					FOLLOW WITH 'END OF FILE'	
0003	00001.000	EA		945328-2203	CDS,OUTIDT,SAMPLE PROGRAM 1-PX990	
C003A					CASSETTE SIDE A,ITEM 3	
0003B					FOLLOW WITH 'END OF FILE'	
C004	00001.000	EA		945328-2204	CDS,P1IDT,SAMPLE PROGRAM 1-PX990	
C004A					CASSETTE SIDE A,ITEM 4	
C004B					FOLLOW WITH 'END OF FILE'	
C005	00001.000	EA		945328-2205	CDS,P2IDT,SAMPLE PROGRAM 1-PX990	
C005A					CASSETTE SIDE A,ITEM 5	
C005B					FOLLOW WITH 'END OF FILE'	
C006	00001.000	EA		945328-2206	CDS,P3IDT,SAMPLE PROGRAM 1-PX990	
C006A					CASSETTE SIDE A,ITEM 6	
C006B					FOLLOW WITH 'END OF FILE'	
C006C					LABEL SIDE A AS FOLLOWS:	
C006D					P/N 945328-0001	

DRAFTSMAN	DATE	CKD. DRAFTSMAN	DATE	DESIGN ENGINEER	DATE	TITLE
[illegible]	5/22/76	[illegible]	5/22/76			SP,SAMPLE PROGRAM 1,CASS SOURCE-PX990

PD.-MFG.	DATE	APPD. PROJECT ENGINEER	DATE	RELEASED	DATE	PROJECT NO.		PART NUMBER	REV
		[illegible]	6-4-76	[illegible]	6/11/76	7506		LM 945328-0001	A

I. 13849

TEXAS INSTRUMENTS
INCORPORATED

DATE 05/22/76

LIST OF MATERIAL

PART NUMBER LM 94[redacted]-0001 REV 1

PRINT ITEM NUMBER	QUANTITY PER ASSEMBLY	UNIT OF ISSUE	DWG. SIZE	PART NUMBER	DESCRIPTION	VENDOR PART NUMBER
C0C6E					CONTENTS:SAMPLE 1	
C0C6F					REF PC -9901 FOR INSTRUCTIONS	

DRAFTSMAN	DATE	CKD. DRAFTSMAN	DATE	DESIGN ENGINEER	DATE	TITLE
						SP, SAMPLE PROGRAM 1, CASS SOURCE-PX990
PD.-MFG.	DATE	APPD. PROJECT ENGINEER	DATE	RELEASED	DATE	PROJECT NO.

PART NUMBER LM 945328-0001 REV A

I. 13849

TEXAS INSTRUMENTS
INCORPORATED

DATE 05/22/76

LIST OF MATERIAL

LM PART NUMBER 945329-0001 REV *

PRINT ITEM NUMBER	QUANTITY PER ASSEMBLY	UNIT OF ISSUE	DWG. SIZE	PART NUMBER	DESCRIPTION	VENDOR PART NUMBER
C0C1	00001.000	EA		945329-2201	CDS,XREF, SAMPLE PROGRAM 2-PX990	
C001A					CASSETTE SIDE A,ITEM 1	
C001B					FOLLOW WITH 'END OF FILE'	
C0C2	00001.000	EA		945329-2202	CDS,PARSEM,SAMPLE PROGRAM 2-PX990	
C002A					CASSETTE SIDE A,ITEM 2	
C002B					FOLLOW WITH 'END OF FILE'	
0003	00001.000	EA		945329-2203	CDS,CTYPM, SAMPLE PROGRAM 2-PX990	
C003A					CASSETTE SIDE A,ITEM 3	
C003B					FOLLOW WITH 'END OF FILE'	
C0C4	00001.000	EA		945329-2204	CDS,PRTPM, SAMPLE PROGRAM 2-PX990	
C0C4A					CASSETTE SIDE A,ITEM 4	
C0C4B					FOLLOW WITH 'END OF FILE'	
C0C5	00001.000	EA		945329-2205	CDS,CSYMM, SAMPLE PROGRAM 2-PX990	
C0C5A					CASSETTE SIDE A,ITEM 5	
C0C5B					FOLLOW WITH 'END OF FILE'	
C0C6	00001.000	EA		945329-2206	CDS,SYMREM,SAMPLE PROGRAM 2-PX990	
C0C6A					CASSETTE SIDE A,ITEM 6	
C0C6B					FOLLOW WITH 'END OF FILE'	
C0C7	00001.000	EA		945329-2207	CDS,SYNCEM,SAMPLE PROGRAM 2-PX990	
C0C7A					CASSETTE SIDE A,ITEM 7	

DRAFTSMAN	DATE	CKD. DRAFTSMAN	DATE	DESIGN ENGINEER	DATE	TITLE
[illegible]	5/22/76	[illegible]	5/22/76			SP,SAMPLE PROGRAM 2,CASS SOURCE-PX990

PD.-MFG.	DATE	APPD. PROJECT ENGINEER	DATE	RELEASED	DATE	PROJECT NO.		PART NUMBER	REV
		M A Moti	6-4-76	[illegible]	6/11/76	7506		LM 945329-0001	*

I. 13849

TEXAS INSTRUMENTS INCORPORATED

DATE 05/22/76

LIST OF MATERIAL

LM PART NUMBER 94[illegible]-0001 REV *

PRINT ITEM NUMBER	QUANTITY PER ASSEMBLY	UNIT OF ISSUE	DWG. SIZE	PART NUMBER	DESCRIPTION	VENDOR PART NUMBER
C0C7B					FOLLOW WITH 'END OF FILE'	
C0C7C					LABEL SIDE A AS FOLLOWS:	
C0C7D					P/N 945329-0001	
C0C7E					CONTENTS:SAMPLE 2	
C0C7F					REF PD -99C1 FOR INSTRUCTIONS	

DRAFTSMAN	DATE	CKD. DRAFTSMAN	DATE	DESIGN ENGINEER	DATE	TITLE
						SP,SAMPLE PROGRAM 2,CASS SOURCE-PX990
PD.-MFG.	DATE	APPD. PROJECT ENGINEER	DATE	RELEASED	DATE	PROJECT NO.

LM PART NUMBER 945329-0001 REV *

13849

TEXAS INSTRUMENTS INCORPORATED

DATE 04/26/77

LIST MATERIAL

PART NUMBER LM0943388-0001

PRINT ITEM NUMBER	QUANTITY PER ASSEMBLY	UNIT OF ISSUE	DWG. SIZE	PART NUMBER	DESCRIPTION	VENDOR PART NUMBE
0001	00001.000	EA		0946243-9701	MNL,PROTO MON(PXRATE) SYS OP GUIDE-990	
0002	00001.000	EA		0948931-0001	SP,CO-RES PROTO SYS LISTINGS(AL)-PXR990	
0002A					REF INDIVIDUAL LISTS OF	
0002B					MATERIALS FOR CONTENTS OF	
0002C					ITEMS 3 THRU 5.	
0003	00001.000	EA		0943388-0021	SP,CO-RES PROTO SYS,CASS 1 OF 3-SYS990	
0004	00001.000	EA		0943388-0022	SP,CO-RES PROTO SYS,CASS 2 OF 3-SYS990	
0005	00001.000	EA		0943388-0023	SP,CO-RES PROTO SYS,CASS 3 OF 3-SYS990	

DRAFTSMAN	DATE	CKD. DRAFTSMAN	DATE	DESIGN ENGINEER	DATE	TITLE
		Larry R. Abrams	4/27/77			SP,CO-RES PROTO SYS ADD-ON(CCO)-SYS990
APPD.-MFG.	DATE	APPD. PROJECT ENGINEER	DATE	RELEASED	DATE	PROJECT NO.
						7506

PART NUMBER LM0943388-0001 A

T.I. 13849

TEXAS INSTRUMENTS
INCORPORATED

DATE 04/26/77

LIST OF MATERIAL

PART NUMBER LM0943388-0021

PRINT ITEM NUMBER	QUANTITY PER ASSEMBLY	UNIT OF ISSUE	DWG. SIZE	PART NUMBER	DESCRIPTION	VENDOR PART NUMBE
0001	00001.000	EA		0945392-1201	CDO,UPFLD,UP FRONT LOADER-PX990	
0001A					SIDE A,ITEM 1	
0001B					FOLLOW WITH 'END OF FILE'	
0002	00001.000	EA		0948929-1207	CDFL,PXRATE/PXRLAL,CO-RES PROTO-PXR990	
0002A					SIDE A,ITEM 2	
0002B					ABS FORMAT:LOAD ADDR=>0A42	
0002C					FOLLOW WITH 'END OF FILE'	
0003	00001.000	EA		0948929-1207	CDFL,PXRATE/PXRLAL,CO-RES PROTO-PXR990	
0003A					SIDE A,ITEM 3	
0003B					ABS FORMAT:LOAD ADDR=>2A42	
0003C					FOLLOW WITH 'END OF FILE'	
0003D					*	
0003E					LABEL SIDE A AS FOLLOWS:	
0003F					943388-0021 REV SIDE A	
0003G					CASSETTE 1 OF 3	
0003H					CONTENTS=12K & 16K PXRATE	
0003I					REF MANUAL 946243-9701 SECT	
0003J					2 FOR INSTRUCTIONS.	
0004	00001.000	EA		0945392-1201	CDO,UPFLD,UP FRONT LOADER-PX990	
0004A					SIDE B,ITEM 1	

DRAFTSMAN	DATE	CKD. DRAFTSMAN	DATE	DESIGN ENGINEER	DATE	TITLE
		Larry R. Aerents 4-27-77				SP,CO-RES PROTO SYS,CASS 1 OF 3-SYS990
APPD.-MFG.	DATE	APPD. PROJECT ENGINEER	DATE	RELEASED	DATE	PROJECT NO. 7506

PART NUMBER LM0943388-0021 A

T.I. 13849

PRINT ITEM NUMBER	QUANTITY PER ASSEMBLY	UNIT OF ISSUE	DWG. SIZE	PART NUMBER	DESCRIPTION	VENDOR PART NUMBE
0004B					FOLLOW WITH 'END OF FILE'	
0005	00001.000	EA		0948929-1207	CDFL,PXRATF/PXRLAL,CO-RES PROTO-PXR990	
0005A					SIDE B,ITEM 2	
0005B					ABS FORMAT:LOAD ADDR=>4A42	
0005C					FOLLOW WITH 'END OF FILE'	
0006	00001.000	EA		0948929-1207	CDFL,PXRATE/PXRLAL,CO-RES PROTO-PXR990	
0006A					SIDE B,ITEM 3	
0006B					ABS FORMAT:LOAD ADDR=>6A42	
0006C					FOLLOW WITH 'END OF FILE'	
0006D					*	
0006E					LABEL SIDE B AS FOLLOWS:	
0006F					943388-0021 REV SIDE B	
0006G					CASSETTE 1 OF 3	
0006H					CONTENTS=20K & 24K PXRATE	
0006I					REF MANUAL 946243-9701 SECT	
0006J					2 FOR INSTRUCTIONS.	

DRAFTSMAN	DATE	CKD. DRAFTSMAN	DATE	DESIGN ENGINEER	DATE	TITLE
						SP,CO-RES PROTO SYS,CASS 1 OF 3-SYS99(
APPD.-MFG.	DATE	APPD. PROJECT ENGINEER	DATE	RELEASED	DATE	PROJECT NO. / PART NUMBER
						LM0943388-0021A

TEXAS INSTRUMENTS
INCORPORATED

DATE 04/26/77

LIST OF MATERIAL

PART NUMBER LM0943388-0022

PRINT ITEM NUMBER	QUANTITY PER ASSEMBLY	UNIT OF ISSUE	DWG. SIZE	PART NUMBER	DESCRIPTION	VENDOR PART NUMBER
0001	00001.000	EA		0948929-1202	CDFL,PXRLAL,OVLY,CO-RES PROTO MON-PXR990	
0001A					SIDE A,ITEM 1	
0001B					FOLLOW WITH 'END OF FILE'	
0002	00001.000	EA		0948929-1203	CDFL,PXRABS,OVLY,CO-RES PROTO MON-PXR990	
0002A					SIDE A,ITEM 2	
0002B					FOLLOW WITH 'END OF FILE'	
0003	00001.000	EA		0948929-1204	CDFL,PXRPROM,OVY,CO-RES PROTO MON-PXR990	
0003A					SIDE A,ITEM 3	
0003B					FOLLOW WITH 'END OF FILE'	
0004	00001.000	EA		0945396-1201	CDO,PROMHP,PROM PROGRAMMER HELPER-PX990	
0004A					SIDE A,ITEM 4	
0004B					FOLLOW WITH 'END OF FILE'	
0005	00001.000	EA		0948929-1205	CDFL,PXRBNPF,OVY,CO-RES PROTO MON-PXR990	
0005A					SIDE A,ITEM 5	
0005B					FOLLOW WITH 'END OF FILE'	
0006	00001.000	EA		0948929-1206	CDFL,PXRHILO,OVY,CO-RES PROTO MON-PXR990	
0006A					SIDE A,ITEM 6	
0006B					FOLLOW WITH 'END OF FILE'	
0006C					*	
0006D					LABEL SIDE A AS FOLLOWS:	

DRAFTSMAN / DATE	CKD. DRAFTSMAN / DATE	DESIGN ENGINEER / DATE	TITLE
	Larry R. Arrants 4-27-77		SP,CO-RES PROTO SYS,CASS 2 OF 3-SYS990
APPD.-MFG. / DATE	APPD. PROJECT ENGINEER / DATE	RELEASED / DATE	PROJECT NO. / PART NUMBER
			7506 / LM0943388-0022

T.I. 13847

PRINT ITEM NUMBER	QUANTITY PER ASSEMBLY	UNIT OF ISSUE	DWG. SIZE	PART NUMBER	DESCRIPTION	VENDOR PART NUMBE
0006E					943388-0022 REV SIDE A	
0006F					CASSETTE 2 OF 3	
0006G					CONTENTS=PXRATE OVERLAYS	
0006H					REF MANUAL 946243-9701 SECT	
0006I					2 FOR INSTRUCTIONS.	
0007	00001.000	EA		0948929-1207	CDFL,PXRATE/PXRLAL,CO-RES PROTO-PXR990	
0007A					SIDE B,ITEM 1	
0007B					FOLLOW WITH 'END OF FILE'	
0007C					*	
0007D					LABEL SIDE B AS FOLLOWS:	
0007E					943388-0022 REV SIDE B	
0007F					CASSETTE 2 OF 3	
0007G					CONTENTS=RELOCATABLE PXRATE	
0007H					REF MANUAL 946243-9701 SECT	
0007I					2 FOR INSTRUCTIONS.	

DRAFTSMAN	DATE	CKD. DRAFTSMAN	DATE	DESIGN ENGINEER	DATE	TITLE
						SP,CO-RES PROTO SYS,CASS 2 OF 3-SYS990
APPD.-MFG.	DATE	APPD. PROJECT ENGINEER	DATE	RELEASED	DATE	PROJECT NO. / PART NUMBER
						LM0943388-0022 A

TEXAS INSTRUMENTS
INCORPORATED

DATE 04/26/77

LIST OF MATERIAL

PART NUMBER LM0943388-0023

PRINT ITEM NUMBER	QUANTITY PER ASSEMBLY	UNIT OF ISSUE	DWG. SIZE	PART NUMBER	DESCRIPTION	VENDOR PART NUMB
0001	00001.000	EA		0936145-2201	CDS,BASCON,SAMPLE PROGRAM 3-PXR990	
0001A					SIDE A,ITEM 1	
0001B					FOLLOW WITH 'END OF FILE'	
0001C					*	
0001D					LABEL SIDE A AS FOLLOWS:	
0001E					943388-0023 REV SIDE A	
0001F					CASSETTE 3 OF 3	
0001G					CONTENTS=SAMPLE 3	
0001H					REF MANUAL 946243-9701 SECT	
0001I					4 FOR INSTRUCTIONS.	
0002	00001.000	EA		0936146-2201	CDS,OPTEST,SAMPLE PROGRAM 4-PXR990	
0002A					SIDE B,ITEM 1	
0002B					FOLLOW WITH 'END OF FILE'	
0002C					*	
0002D					LABEL SIDE B AS FOLLOWS:	
0002E					943388-0023 REV SIDE B	
0002F					CASSETTE 3 OF 3	
0002G					CONTENTS=SAMPLE 4	
0002H					REF MANUAL 946243-9701 SECT	
0002I					4 FOR INSTRUCTIONS.	

DRAFTSMAN	DATE	CKD. DRAFTSMAN	DATE	DESIGN ENGINEER	DATE	TITLE
		Larry R. Arnants 4-27-77				SP,CO-RES PROTO SYS,CASS 3 OF 3-SYS990
APPD.-MFG.	DATE	APPD. PROJECT ENGINEER	DATE	RELEASED	DATE	PROJECT NO. 7506

PART NUMBER LM0943388-0023 A

T.I. 13849

A

APPLICATION	
NEXT ASSY	USED ON
	7506

REVISIONS			
LTR	DESCRIPTION	DATE	APPROVED

REV	
SHEET	
REV STATUS OF SHEETS	REV
	SHEET

UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES
TOLERANCES
ANGLES ±1°
3 PLACE DECIMAL ±.010
2 PLACE DECIMAL ±.02

IDENTIFYING NUMBERS SHOWN IN PARENTHESES FOR REFERENCE ONLY

INTERPRET DWG IN ACCORDANCE WITH MIL-STD-100

DWN | DATE
CHK
ENGR
QA
APVD
CONTR NO
DESIGN ACTIVITY RELEASE

TEXAS INSTRUMENTS
INCORPORATED
Equipment Group *Dallas, Texas*

PD, SAMPLE PROGRAM 1 FOR 990-733 ASR SYSTEM SOFTWARE AND PROTOTYPING SYSTEM SOFTWARE-PX990

SIZE	CODE IDENT NO	DRAWING NO
A	96214	945328-9901

SCALE | REV. A | SHEET 1 of 34

TI-0923-C

VERIFICATION OF
MODEL 990 COMPUTER
733 ASR PROGRAM DEVELOPMENT SYSTEM
OPERATION GUIDE

1.0 PURPOSE

The procedure and results listed below are designed to verify the operations and capabilities described in the "Model 990 Computer 733 ASR Program Development System Operation Guide". It starts with assembly of a sample source program which has been keyed to tape cassette prior to the described procedures, and progresses through Text Editing, Reassembly, Linking and Loading and finally Execution.

2.0 SAMPLE PROGRAM

2.1 DESCRIPTION

The sample program is interactive, communicating with the operator through the 733 ASR Keyboard and printer. It recognizes numbers (1-10) written in 5 different forms (4 languages and arabic numerals) and prints the language it was written in and its numerical value. It then, in response to operator input, prints the name of the number in any of the 5 recognizable forms (Spanish, English, Russian, German, and Digits). The program notifies the operator if it does not recognize the name of the number or the language requested and allows re-entry of the number or choice of another number or language.

2.2 CONSTRUCTION

The program is divided into 6 source modules.

2.2.1 TDRIVE - The main program which calls required programs to access data, output messages and process inputs.

2.2.2 TSKIDT - Program contains all the data used by the other modules.

2.2.3 OUTIDT - Processes all outputs messages and clears output message location.

2.2.4 P1IDT, P2IDT, P3IDT - Process the three requested inputs and builds output messages.

2.3 OBJECTIVE

Sample Program was modularized to demonstrate use of the link and loader. A syntactic error was introduced in TDRIVE, module one, to demonstrate assembly function and use of text editor to correct program errors. Finally, the program was written interactively to graphically depict that it is executing properly.

2.4 SUPPLEMENTARY DESCRIPTIONS

2.4.1 A flow chart of the verification procedure can be found in Appendix A.

3.0 EQUIPMENT REQUIREMENTS

To run this test the following equipment and tape cassettes are required.

3.1 EQUIPMENT

To run this test a 733 program development system is required.

3.2 TAPE CASSETTES

SYSTEM CASSETTES	USER CASSETTES
PX9MTR (ROOT) = MONITOR ROOT	SAMPLE PROGRAM 1 (5 modules)
LAL = LINK AND LOADER OVERLAY	BLANK CASSETTE ① - BC1
PX9ASM = ASSEMBLER	BLANK CASSETTE ② - BC2
PX9EDT = TEXT EDITOR	BLANK CASSETTE ③ - BC3

4.0 CASSETTE LOADING AND UNLOADING PROCEDURES

Through-out these procedures, instructions will be given to load and ready a cassette or unload and remove a cassette.

4.1 LOAD AND READY A CASSETTE

Mount cassette tape in Cassette unit 1 or 2 (left or right unit) - CS1 or CS2 as instructed; close door; press unit rewind switch; watch for "END" LED to light; then press load/fast forward switch, wait for READY LED to light.

4.2 UNLOAD A CASSETTE

Press rewind switch on cassette unit indicated; wait for end LED to light; open cassette unit door and remove tape cassette.

4.3 All system Cassettes (listed above may be mounted in a rack for easy access. After use they should be returned to their labeled holder to facilitate later use.

5.0 KEY TO ABBREVIATIONS USED IN THE VERIFICATION PROCEDURES

KB: 733 ASR Keyboard

PRT: 733 ASR Printer

CS1: 733 ASR LEFT CASSETTE UNIT

CS2: 733 ASR RIGHT CASSETTE UNIT

[CR] CARRIAGE RETURN

[ ] CONTROL KEY/TAB ON KEYBOARD

5.0A VERIFICATION PROCEDURES

5.1 STEPS

The procedures are divided into 6 STEPS:

1. STEP A - LOADING MONITOR
2. STEP B - ASSEMBLY
3. STEP C - TEXT EDITING
4. STEP D - REASSEMBLY
5. STEP E - LINK AND LOADING
6. STEP F - EXECUTION

5.2 PREPARATION

To perform the following procedures you will need:

5.2.1 THE SAMPLE SOURCE PROGRAM 1 ON TAPE CASSETTE (6 MODULES)

5.2.2 THREE (3) BLANK CASSETTES: WITH NEW BLANK LABELS

LABEL THEM: BC①, BC2 and BC3 respectively. They will be referred to by those names in the procedures.

5.3 DESCRIPTION

After verifying the hardware is properly configured and connected, turn the power switches on the computer and the 733ASR to ON.

5.3 DESCRIPTION (continued)

A. LOADING MONITOR

DEVICE	INSTRUCTIONS	DEVICE	RESPONSE
733 ASR/CS1	1. LOAD AND READY PXMTR (ROOT) CASSETTE	73RASR	
	2. VERIFY SYSTEM POWER IS ON.		3. CS1 READY LED ON
	4. TURN PLAYBACK/RECORD SWITCH TO PLAYBACK		5. CS1 PLAYBACK LED ON
	6. PRESS: HALT/SIE BUTTON	FRONT PANEL	7. RUN LED: OFF
	8. PRESS CLEAR BUTTON		
	9. PRESS LOAD BUTTON	CS1	10. PXMTR (ROOT) LOADED
			11. WHEN LOADING IS COMPLETED, PX9 MTR (ROOT) GAINS CONTROL AND OUTPUTS ITS PROMPTING CHARACTER: A PERIOD (.)
CS1	12. UNLOAD PX9MTR(ROOT) AND RETURN CASSETTE TO RACK POSITION		
	13. PROCEED TO STEP B		

B. ASSEMBLY

6 TEST PROGRAM SOURCE MODULES ARE ASSEMBLED. MODULE 1: TDRIVE HAS SYNTACTIC ERROR.

DEVICE	INSTRUCTIONS	DEVICE	RESPONSE
CS1	1. LOAD AND READY PX9 ASM CASSETTE		
		733ASR	3. CS1 READY LED ON
KB	4. ENTER: LU [CR]		5. ASSEMBLER LOADED INTO MEMORY UNDER PX9MTR (ROOT)
		PRT	6. MONITOR PROMPTING CHARACTER PRINTED AT COMPLETION OF LOAD (.).
CS1	7. UNLOAD PX9ASM/UFL AND RETURN TO RACK POSITION		
CS1	8. LOAD AND READY TEST SOURCE PROGRAM (6 MODULES) CASSETTE	733ASR	9. CS1 READY LED ON
CS2	10. LOAD AND READY BC1 CASSETTE		11. CS2 READY LED ON
KB	12. ENTER: EX [CR]	PRT	13. PRINT: PXASM VERSION 1.0 15 MARCH 1976 ADD 4K MEM BLOCKS CONFIGURED?
KB	14. ENTER: 0 [CR]	PRT	15. PRINTS: PREDEFINED REGISTERS?
KB	16. ENTER: Y [CR]	PRT	17. PRINTS: ASM/TERM?
KB	18. ENTER: A [CR] (MODULE 1: TDRIVE)		19. ASSEMBLY COMMENCES
		PRT	20. PRINTS: ASSEMBLY LISTING MODULE 1 NOTE: ^ SYNTAX ERROR IN MODULE 1
		CS2	21. OUTPUT OBJECT MODULE 1
		C	22. PRINTS: ASM/TERM?
KB	23. ENTER: A [CR] (MODULE 2: TSKIDT)		23. ASSEMBLY RESUMES: ASSEMBLY LISTING AND OBJECT CODE OUTPUT (MODULE 2)
		PRT	24. PRINTS: ASM/TERM?
KB	25. ENTER: A [CR] (MODULE 3; OUTIDT)		
KB	26. REPEAT ABOVE RESPONSE UNTIL P3IDT, MODULE 6 IS ASSEMBLED	PRT	27. PRINTS: ASM/TERM?
KB	27. ENTER: T [CR] (AFTER P3IDT ASSEMBLY)	PRT	28. MONITOR PROMPTING CHARACTER PRINTED (.).

```
0003                        IDT  'TDRIVE'
0004                  * TITLE:    TDRIVE
0005                  *           TRANSLATION DRIVER
0006                  * REVISION: 03/15/76
0007                  *           ORIGINAL
0008                  * COMPUTER: 990,ASSEMBLY
0009                  * ABSTRACT:
0010                  *           THIS IS THE MAIN PROGRAM: IT CALLS THE*
0011                  *           REQUIRED SUBPROGRAMS TO OUTPUT MSGS AND *
0012                  *           PROCESS INPUT DATA.  THE PROGRAM REQUEST *
0013                  *           A NUMBER BE ENTERED AND ACCEPTS BOTH DIGITS *
0014                  *           AND NAMES OF NUMBERS (1-10) WRITTEN IN *
0015                  *           FIVE DIFFERENT LANGUAGES:  ENGLISH, SPANISH *
0016                  *           GERMAN, RUSSIAN AND DIGITS.*
0017                  *           IT RETURNS THE NAME OF THE LANGUAGE THE *
0018                  *          NUMBER WAS WRITTEN IN AND ITS VALUE. *
0019                  *           NEXT IT ALLOWS YOU TO TRANSLATE THE *
0020                  *           NUMBER OF ANY OF THE 5 LANGUAGES. *
0021                  *           FINALLY IT REQUEST WHETHER YOU WISH TO *
0022                  *           CONTINUE, AND IF SO, ALLOWS YOU TO ENTER *
0023                  *           ANOTHER NUMBER AND REPEATS THE ENTIRE *
0024                  *           PROCESS.  ERROR MSGS ARE PRINTED FOR *
0025                  *           UNRECOGNIZABLE NUMBERS AND LANGUAGES *
0026                  *
0027  0000            WD    BSS  32
0028                        DXOP SVC,15
0029                        REF  OUTIDT,OUTPRC,MSG1,TSKIDT
0030                        REF  INPUT,P1IDT,P2IDT,INPRC1
0031                        REF  INPRC2,MSG2,MSG3,MSG4,MSG5,MSG6
0032                        REF  MSG7,ENDPT,P3IDT,INPRC3
0033  0020  02E0  START     LWPI WD
      0022  0000'
0034  0024  0420            BLWP @OUTPRC              ENTER NUMBER
      0026  0000
0035  0028  0000            DATA MSG1
0036  002A  2FE0            SVC  @INPUT               INPUT NUMBER EG. EINS
      002C  0000
0037  002E  0420            BLWP @INPRC1              PROCESS NUMBER ENTERED
      0030  0000
** ERR 2 - STMT 0038
0038  0032  10E6            JMP  MSG3                 ****SYNTACTIC ERROR*****
0039  0034  0420            BLWP @OUTPRC
      0036  0026'
0040  0038  0000            DATA MSG2
0041                  *  NO WAS WRITTEN IN "GERMAN", ITS VALUE IS "2"
0042  003A  0420            BLWP @OUTPRC
      003C  0036'
0043  003E  0000            DATA MSG5                 ENTER LANGUAGE DESIRED
0044  0040  2FE0            SVC  @INPUT
      0042  002C'
0045  0044  0420            BLWP @INPRC2              PROCESS LANGUAGE ENTER VALIDAT
      0046  0000
0046  0048  10--            JMP  @MSG6                INVALID LANGUAGE GO TO WRITE L
0047  004A  0420            BLWP @OUTPRC              "GERMAN" NAME FOR THIS NO IS "
```

```
.       004C   003C'
0048    004E   0000          DATA MSG7
0049    0050   0420  DMSG4   BLWP @OUTPRC
        0052   004C'
0050    0054   0000          DATA MSG4
0051    0056   2FE0          SVC  @INPUT          ENTER Y OR N!
.       0058   0042'
0052    005A   0420          BLWP @INPRC3
        005C   0000
0053    005E   10E0          JMP  START
0054    0060   2FE0          SVC  @ENDPT
        0062   0000
0055    0064   0420  DMSG3   BLWP   @OUTPRC
        0066   0052'
0056    0068   0032'         DATA MSG3
0057    006A   10F2          JMP  DMSG4
0058    006C   0420  DMSG6   BLWP   @OUTPRC
        006E   0066'
        0048**1011
0059    0070   0000          DATA MSG6
0060    0072   10EE          JMP  DMSG4
0061                         END  START
```

0001 ERRORS
LAST ERR - STMT 0038

ASM/TERM? A

```
0003                        IDT  'TSKIDT'
0004                  * TITLE:   TSKIDT
0005                  *          DATA DEFINITIONS
0006                  * REVISION: 03/15/76
0007                  *          ORIGINAL
0008                  * COMPUTER: 990,ASSEMBLY
0009                  * ABSTRACT:
0010                  *          TASK IDT PROVIDES DATA DEFINITINS TO ALL*
0011                  *          PROGRAMS *
0012                  *
0013                        DEF  NSAVE,MSG1,MSG2,LANGUA,DIGIT
0014                        DEF  MSG3,MSG4,MSG5,MSG6
0015                        DEF  MSG7,YAZEEK,NUMBER
0016                        DEF  NUMS,LNGTH,TRANS,YES,NO
0017                        DEF  OUTPUT,PRNBUF,OUTCNT,ENDPT
0018                        DEF  INPUT,INBUF,BTD,DEC,INCNT
0019  0000          NSAVE BSS  200
0020        00CA' MSG1    EQU  $+2
0021  00C8  000F          DATA 15                 *MSG1: ENTER NUMBER "VOSYEM"
0022  00CA    0A          BYTE >A,>D
      00CB    0D
0023  00CC    45          TEXT 'ENTER NUMBER '
0024                  *
0025                        EVEN
0026        00DC' MSG2    EQU  $+2                *MSG2: NUMBER WAS WRITTEN IN "
0027  00DA  0034          DATA 52
0028  00DC    0A          BYTE >A,>D
      00DD    0D
0029  00DE    4E          TEXT 'NUMBER WAS WRITTEN IN '
0030  00F4          LANGUA BSS  7
0031  00FB    2C          TEXT ', ITS VALUE IS '
0032  010A          DIGIT  BSS  6
0033                        EVEN
0034        0112' MSG3    EQU  $+2
0035  0110  0022          DATA 34
0036  0112    0A          BYTE >A,>D
      0113    0D
0037  0114    4E          TEXT 'NUMBER WAS NOT PROPERLY WRITTEN '
0038                  *
0039                        EVEN
0040        0136' MSG4     EQU $+2
0041  0134  0023          DATA 35
0042  0136    0A          BYTE >A,>D
      0137    0D
0043  0138    44          TEXT 'DO YOU WISH TO CONTINUE? (Y OR N)'
0044                  *
0045                        EVEN
0046        015C' MSG5    EQU  $+2
0047  015A  0019          DATA 25
0048  015C    0A          BYTE >A,>D
      015D    0D
0049  015E    45          TEXT 'ENTER LANGUAGE DESIRED '
0050                  *
0051                        EVEN
```

```
0052          0178' MSG6    EQU  $+2
0053  0176    0018          DATA 24
0054  0178      0A          BYTE >A,>D
      0179      0D
0055  017A      4C                TEXT  'LANGUAGE NOT AVAILABLE'
0056                        *
0057                        EVEN
0058          0192' MSG7    EQU  $+2
0059  0190    0029          DATA 41
0060  0192      0A          BYTE >A,>D
      0193      0D
0061  0194          YAZEEK  BSS  7
0062  019B      20          TEXT ' NAME FOR THIS NUMBER IS '
0063  01B4          NUMBER  BSS  7
0064          01BB' NUMS    EQU  $
0065  01BB      4F  ENUM    TEXT 'ONETWOTHREEFOUR'
0066  01CA      46          TEXT 'FIVESIXSEVEN'
0067  01D6      45          TEXT 'EIGHTNINETEN'
0068  01E2      31  DNUM    TEXT '12345678910'
0069  01ED      55  SNUM    TEXT 'UNODOSTRESCUATRO'
0070  01FD      43          TEXT 'CINCOSEISSIETE'
0071  020B      4F          TEXT 'OCHONUEVEDIEZ'
0072  0218      4F  RNUM    TEXT 'ODEENDVATREECHATERY'
0073  022B      50          TEXT 'PYATSHESTSYEM'
0074  0238      56          TEXT 'VOSYEMDYEVITDYESIT'
0075  024A      45  GNUM    TEXT 'EINSZWEIDREIVIER'
0076  025A      46          TEXT 'FUNFSECHSSIEBEN'
0077  0269      4F          TEXT 'OCHTSNEUNZEN'
0078                        *
0079          0275' LNGTH   EQU  $
0080  0275      03  ELNGTH BYTE 3,3,5,4,4,3,5,5,4,3
      0276      03
      0277      05
      0278      04
      0279      04
      027A      03
      027B      05
      027C      05
      027D      04
      027E      03
0081  027F      01  DLNGTH BYTE 1,1,1,1,1,1,1,1,1,2
      0280      01
      0281      01
      0282      01
      0283      01
      0284      01
      0285      01
      0286      01
      0287      01
      0288      02
0082  0289      03  SLNGTH BYTE 3,3,4,6,5,4,5,4,5,4
      028A      03
      028B      04
      028C      06
      028D      05
```

```
         028E    04
         028F    05
         0290    04
         0291    05
         0292    04
0083     0293    05   RLNGTH BYTE 5,3,4,7,4,5,4,6,6,6
         0294    03
         0295    04
         0296    07
         0297    04
         0298    05
         0299    04
         029A    06
         029B    06
         029C    06
0084     029D    04   GLNGTH BYTE 4,4,4,4,4,5,6,5,4,3
         029E    04
         029F    04
         02A0    04
         02A1    04
         02A2    05
         02A3    06
         02A4    05
         02A5    04
         02A6    03
0085                  *
0086     02A7    45   TRANS  TEXT 'ENGLISH '
0087     02AF    44          TEXT 'DIGITS  '
0088     02B7    53          TEXT 'SPANISH '
0089     02BF    52          TEXT 'RUSSIAN '
0090     02C7    47          TEXT 'GERMAN  '
0091     02CF    59   YES    TEXT 'Y'
0092     02D0    4E   NO     TEXT 'N'
0093                  *
0094                  * OUT SVC
0095                  *
0096                         EVEN
0097     02D2    00   OUTPUT BYTE 0,0
         02D3    00
0098     02D4    0B   PREO   BYTE >B,6
         02D5    06
0099     02D6  0000           DATA 0
0100     02D8  0000   PRNBUF DATA 0
0101     02DA  0000           DATA 0
0102     02DC  0000   OUTCNT DATA 0
0103     02DE                 BSS  4
0104                  *
0105                  *  RETURN TO MONITOR
0106                  *
0107     02E2    04   ENDPT  BYTE 4,0
         02E3    00
0108                  *
0109                  *  INPUT SVC
0110                  *
0111     02E4    00   INPUT  BYTE 0,0
```

```
         02E5     00
  0112   02E6     09   PRBI    BYTE 9,6
         02E7     06
  0113   02E8   0000           DATA 0
  0114   02EA   ----   BADRS   DATA INBUF
  0115   02EC   0007   BUFSZ   DATA 7
  0116   02EE   0000   INCNT   DATA 0
  0117   02F0                  BSS  4
  0118   02F4          INBUF   BSS  7
         02EA**02F4'
  0119   02FB     0A   BTD     BYTE >A,0
         02FC     00
  0120   02FD          DEC     BSS  6
  0121                         END

0000 ERRORS

ASM/TERM? A
```

```
                              IDT  'OUTIDT'
                    * TITLE:  OUTIDT
                    *         FORMAT AND OUTPUT
                    * REVISION: 03/15/76
                    *         ORIGINAL
                    * COMPUTER: 990,ASSEMBLY
                    * ABSTRACT:
                    *         DEFINES OUTPROCESS FORMAT AND PERFORMS *
                    *         OUTPUT TO THE PRINTER *
                    *
                              DEF  OUTPRC,CLEAR
                              REF  PRNBUF,OUTCNT,TSKIDT,OUTPUT
0000                W0        BSS  32
                              DXOP SVC,15
0020  0000'         OUTPRC    DATA W0
0022  ----                    DATA OP
0024  C81E          OP        MOV  *R14,@PRNBUF       PUT MSG ADRS IN PRB
0026  0000
0022**0024'
0028  C33E                    MOV  *R14+,R12
002A  064C                    DECT R12
002C  C81C                    MOV  *R12,@OUTCNT       PUT LNGTH OF MSG IN PRB
002E  0000
0030  2FE0                    SVC  @OUTPUT
0032  0000
0034  0380                    RTWP
                    *
                    *****CLEAR BUFFER ROUTINE
                    *
0036  0000'         CLEAR     DATA W0
0038  ----                    DATA CL1
003A  C1FE          CL1       MOV  *R14+,R7           R7:LNGTH OF FIELD TO CLEAR
0038**003A'
003C  C23E                    MOV  *R14+,R8           R8:NAME OF FIELD TO CLEAR
003E  04C1                    CLR  R1
0040  DE01          CL        MOVB R1,*R8+
0042  0607                    DEC  R7
0044  16FD                    JNE  CL
0046  0380                    RTWP
                              END

0000 ERRORS

ASM/TERM? A
```

```
0003                        IDT  'P1IDT'
0004                  * TITLE:   P1IDT
0005                  *          GET VALUE AND POINTER
0006                  * REVISION: 03/15/76
0007                  *          ORIGINAL
0008                  * COMPUTER: 990,ASSEMBLY
0009                  * ABSTRACT:
0010                  *          THIS SUBPROGRAM USES THE LENGTH OF NUMBERS *
0011                  *          ENTERED ON THE KEYBOARD TO SEARCH THROUGH *
0012                  *          A TABLE OF LENGTHS WHICH KEYS COMPARISON *
0013                  *          WITH A PARELLEL TABLE OF NUMBERS WRITTEN *
0014                  *          IN DIFFERENT LANGUAGES.  IF NUMBER NOT *
0015                  *          FOUND PROGRAM RETURNS TO ERROR MSG 3 *
0016                  *          WHEN THE NUMBER IS FOUND, PRGM COMPUTES *
0017                  *          VALUE AND PTR TO LANGUAGE.  IT THEN USES *
0018                  *          THIS INFO TO BUILD MSG 2 *
0019                  *
0020  0000            WP1    BSS  32
0021                         DEF  INPRC1
0022                         REF  TSKIDT,NUMS,LNGTH,INBUF,NSAVE
0023                         REF  TRANS,LANGUA,BTD,DEC,DIGIT
0024                         REF  INCNT
0025                         DXOP SVC,15
0026  0020  0000'    INPRC1 DATA WP1
0027  0022  ----           DATA P1
0028  0024  04C1     P1     CLR  R1                  R1=COUNT OF LNGTH COMPARE
      0022*+0024'
0029  0026  04C2            CLR  R2                  R2:PTR TO NUMBER TO COMPARE
0030  0028  0208            LI   R8,100              R8=MAX NUMBER OF COMPPARES
      002A  0064
0031  002C  C020            MOV  @INCNT,R0           R0: NO. OF CHARS READ IN
      002E  0000
0032  0030  06C0            SWPB R0
0033  0032  9840    LCOMP   CB   R0,@LNGTH(R1)       COMPARE NOS. WITH SAME
      0034  0000
0034                  *                              LNGTH AS INPUT, IGNOR OTHERS
0035  0036  13--            JEQ  NUMCPR              IF EQU GO TO COMPARE NUMBER
0036  0038  D1A1    LKMORE  MOVB @LNGTH(R1),R6       ELSE INC PTR TO COMPR
      003A  0034'
0037  003C  0986            SRL  R6,8
0038  003E  A086            A    R6,R2
0039  0040  0581            INC  R1
0040  0042  0608            DEC  R8
0041  0044  16F6            JNE  LCOMP               IF LESS THAN 100 COMPARE CONTI
0042  0046  0380            RTWP                     ELSE NUMBER NOT FOUND RETURN T
0043  0048  C0E0    NUMCPR  MOV  @INCNT,R3           R3:NO..OF CHARS READ IN
      004A  002E'
      0036*+1308
0044  004C  C142            MOV  R2,R5               R5:PTR TO NUMS INDXD
0045  004E  0204            LI   R4,INBUF            R4:ADRS OF NUMBER INPUTTED
      0050  0000
0046  0052  9974    CMP     CB   *R4+,@NUMS(R5)      COMPARE LETTER BY LETTER
      0054  0000
0047  0056  16F0            JNE  LKMORE              IF NOT EQU LOOK FOR OTHER COMP
```

```
  0048   0058   0585          INC   R5
  0049   005A   0603          DEC   R3
  0050   005C   16FA          JNE   CMP
  0051   005E   C1C1  GOODFD  MOV   R1,R7          R7:COUNT OF LNGTHS
  0052   0060   0209          LI    R9,10
         0062   000A
  0053   0064   04C6          CLR   R6
  0054   0066   3D89          DIV   R9,R6          R7MODULO(10)=VALUE OF NO-1
  0055   0068   0587          INC   R7             R7:VALUE OF NO.
  0056   006A   C807          MOV   R7,@NSAVE      R6:INT(COUNT LNGTH/10)=INDEX T
         006C   0000
  0057   006E   0A36  PREP1   SLA   R6,3
  0058   0070   0209          LI    R9,7           R9:CHAR CNT IN TRANSFER
         0072   0007
  0059   0074   020A          LI    R10,TRANS
         0076   0000
  0060   0078   A286          A     R6,R10         R10 PTS TO LANGUAGE
  0061   007A   020C          LI    R12,LANGUA
         007C   0000
  0062   007E   DF3A  L4      MOVB  *R10+,*R12+    TRANSFER NAME TO LANGUA IN MSG
  0063   0080   0609          DEC   R9
  0064   0082   16FD          JNE   L4
  0065   0084   C007          MOV   R7,R0          R0:BINARY VALUE TO CNVRT
  0066                       *                    TO DASCII TO PRT
  0067   0086   2FE0          SVC   @BTD
         0088   0000
  0068   008A   0203          LI    R3,5
         008C   0005
  0069   008E   D8E3  L5      MOVB  @DEC(R3),@DIGIT(R3)  TRANS VALUE TO MSG2
         0090   0000
  .      0092   0000
  0070   0094   0603          DEC   R3
  0071   0096   18FB          JOC   L5
  0072   0098   05CE          INCT  R14            SET RETRN TO OUTPUT MSG 2
  0073   009A   0380          RTWP
  0074                       END

0000 ERRORS

ASM/TERM? A
```

```
0003                        IDT  'P2IDT'
0004                  * TITLE:    P2IDT
0005                  *           COMPUTE OUTPUT SENTENCE
0006                  * REVISION: 03/15/76
0007                  *           ORIGINAL
0008                  * COMPUTER: 990,ASSEMBLY
0009                  * ABSTRACT:
0010                  *           THIS SUBPROGRAM COMPARES LANGUAGE ENTERED *
0011                  *           (INBUF*) WITH TABLE OF RECOGNIZABLE LANGU- *
0012                  *           AGES (TRANS):  USES PTR TO LANGUAGE AND VALUE *
0013                  *           OF NUMBER TO LOCATE COMPUTED VALUES THRU *
0014                  *           'LNGTH' TABLE WHICH POINT TO LOCATIONS AND *
0015                  *           LENGTH OF NAME OF NUMBER (TRANSLATION)-NUMS *
0016                  *           THESE VALUES ARE USED TO OUTPUT NAME OF TRAN-*
0017                  *       SLATED NUMBER OT MSG 7,E.G. "RUSSIAN NAME FOR THIS*
0018                  *           NUMBER IS OIEEN" *
0019                  *
0020  0000            WP2     BSS  32
0021                          DEF  INPRC2
0022                          REF  TSKIDT,INBUF,TRANS
0023                          REF  YAZEEK,NSAVE,LNGTH
0024                          REF  NUMBER,NUMS,CLEAR,OUTIDT
0025                          REF  INCNT
0026  0020  0000'   INPRC2  DATA WP2
0027  0022  ----            DATA P2
0028  0024  0209    P2      LI   R9,5                R9:USED TO CONTROL TRANS TABLE
      0026  0005
      0022** 0024'
0029                  *                              LOOK-UP.
0030  0028  C020            MOV  @INCNT,R0           INCNT:USED TO CONTROL NO.
      002A  0000
0031                  *                              OF LTRS OF LANGUAGE COMPARE
0032  002C  04C3            CLR  R3                  R3:INDEX THRU TRANS TABLE
0033  002E  C083    RT1     MOV  R3,R2               R2:INDX THRU CHAR IN TRANS
0034  0030  04C1            CLR  R1                  R1:INDX THRU CHAR IN INPUT BUF
0035  0032  98A1    NXLTR   CB   @INBUF(R1),@TRANS(R2)
      0034  0000
      0036  0000
0036  0038  13--            JEQ  CNTU
0037  003A  0223            AI   R3,8                BAD COMPARE,INC TO PT TO
      003C  0008
0038                  *                              NEXT TRANS TABLE ENTRY
0039  003E  0609            DEC  R9
0040  0040  16F6            JNE  RT1
0041  0042  0380            RTWP                     NO COMPARES RETRN TO PRC ERROR
0042                  *                              MSG 6
0043  0044  0581    CNTU    INC  R1
      0038** 1305
0044  0046  0582            INC  R2
0045  0048  8001            C    R1,R0
0046  004A  11F3            JLT  NXLTR
0047  004C  04C2    GOOD    CLR  R2
0048  004E  C283            MOV  R3,R10              R10:INDEX TO LANGUAGE
0049  0050  0209            LI   R9,10
```

```
      0052  000A
0050  0054  0933          SRL   R3,3              ADJUST R3 TO PT TO LNGTH TABLE
0051  0056  38C9          MPY   R9,R3
0052  0058  A120          A     @NSAVE,R4
      005A  0000
0053  005C  0604          DEC   R4
0054  005E  C144          MOV   R4,R5             R5:PTR TO LNGTH OF NUMBER
0055  0060  0604          DEC   R4
0056  0062  11--          JLT   ADJUST
0057  0064  C184          MOV   R4,R6             R6:COUNT OF LNGTHS TO BE SUMME
0058                  *                           D TO POINT TO NAME OF NO.
0059  0066  D1E5          MOVB  @LNGTH(R5),R7     R7:LENGTH OF NUMBER
      0068  0000
0060  006A  0987          SRL   R7,8              RT JUSTIFY LENGTH IN REG7
0061  006C  04C9          CLR   R9
0062  006E  D026  SUM     MOVB  @LNGTH(R6),R0     R9:PTR TO NO. IN NUMS TABLE
      0070  0068'
0063  0072  0980          SRL   R0,8
0064  0074  A240          A     R0,R9
0065  0076  0606          DEC   R6
0066  0078  18FA          JOC   SUM
0067  007A  10--          JMP   CLRNUM
0068  007C  04C9  ADJUST  CLR   R9                ADJUST WHEN INBUF=ENGLISH(VALU
      0062**110C
0069  007E  D1E5          MOVB  @LNGTH(R5),R7     R9:0,R7:3-LENGTH OF ONE
      0080  0070'
0070  0082  0987          SRL   R7,8
0071                  *
0072                  *CLEAR NUMBER ENTRY IN MSG 7 BEFORE ENTERING NAME
0073                  *
0074  0084  0420  CLRNUM BLWP  @CLEAR
      0086  0000
      007A**1004
0075  0088  0007          DATA  7
0076  008A  0000          DATA  NUMBER
0077                  *
0078                  ****MOVE NAME OF NUMBER TO MSG7
0079                  *
0080  008C  0200          LI    R0,NUMBER
      008E  008A'
0081  0090  DC29  NMFILL  MOVB  @NUMS(R9),*R0+
      0092  0000
0082  0094  0589          INC   R9
0083  0096  0607          DEC   R7
0084  0098  16FB          JNE   NMFILL
0085                  *
0086                  ***MOVE LANGUAGE NAME TO MSG7
0087                  *
0088  009A  0206          LI    R6,YAZEEK
      009C  0000
0089  009E  022A          AI    R10,TRANS         R10 CONTRAINED PTR TO LANGUAGE
      00A0  0036'
0090  00A2  0207          LI    R7,7
      00A4  0007
0091  00A6  DDBA  TR      MOVB  *R10+,*R6+
```

```
0092  00A8  0607        DEC  R7
0093  00AA  16FD        JNE  TR
0094  00AC  05CE        INCT R14
0095  00AE  0380        RTWP
0096                    END

0000 ERRORS

ASM/TERM? A
```

```
0003                        IDT   'P3IDT'
0004                  * TITLE:    P3IDT
0005                  *           PROCESS REPLY TO QUESTION
0006                  * REVISION: 03/15/76
0007                  *           ORIGINAL
0008                  * COMPUTER: 990,ASSEMBLY
0009                  * ABSTRACT:
0010                  *           PROCESS YES OR NO REPLY--KEYS OFF THE *
0011                  *           LETTER 'Y' IN THE INPUT BUFFER (INBUF).*
0012                  *           TREATS ALL OTHER ENTRYS AS A 'NO'.  TRANSFERS*
0013                  *           CONTROL BACK TO MAIN TO TERMINATE OR *
0014                  *           CONTINUE(Y) *
0015                  *
0016  0000            WP3     BSS   32
0017                          DEF   INPRC3
0018                          REF   TSKIDT,YES,INBUF
0019  0020  0000'     INPRC3  DATA  WP3
0020  0022  ----              DATA  P3
0021  0024  9820      P3      CB    @YES,@INBUF
      0026  0000
      0028  0000
      0022**0024'
0022  002A  13--              JEQ   STRT
0023  002C  05CE              INCT  R14                   CHANGE RETURN TO ENDPT
0024  002E  0380      STRT    RTWP
      002A**1301
0025                          END

0000 ERRORS

ASM/TERM? T
```

```
.LU
.EX
PX9EDT  VERSION 1.0  01DEC75
ADD 4K MEM BLOCKS CONFIGURED? 0

POSITION TAPES, ENTER CR


 ?D42
 ?P38-38
 0038          JMP   MSG3                ****SYNTACTIC ERROR*****
 ?C38-38
 JMP OMSG3 NUMBWE


              BER NOT FOUND,PRT ERROR MSG

 ?Q
END EDIT
TERMINATE/CONTINUE?T
```

C. TEXT EDIT

TEST PROGRAM SOURCE MODULE: TDRIVE EDITED TO REMOVE SYNTACTIC ERROR

DEVICE	INSTRUCTIONS	DEVICE	RESPONSE
CS1	1. LOAD AND READY PX9EDT CASSETTE	733ASR	2. CS1 READY LED ON
	3. ENTER: LU [CR]		4. TEXT EDITOR LOADED INTO MEMORY. UNDER PX9MTR
		PRT	5. MONITOR PROMPTING CHAR PRINTED AT COMPLETION OF LOAD (.).
CS1	6. UNLOAD PX9EDT AND RETURN TO RACK		
CS1	7. RELOAD AND READY TEST PROGRAM (SOURCE)	733ASR	8. CS1 READY LED ON
CS2	9. LOAD AND READY BC2 CASSETTE	733ASR	10. CS2 READY LED ON.
KB	11. ENTER: EX [CR]	PRT	12. PRINTS: PX9EDT VERSION 1.0 DECEMBER 75 ADD 4K MEM BLOCKS CONFIGURED
KB	13. ENTER: 0 [CR]	PRT	14. PRINTS: POSITION TAPES, ENTER CR
KB	15. ENTER: [CR]	PRT	16. PRINTS: ? (THIS IS THE TEXT EDITOR PROMPTING CHARACTER)
KB	17. ENTER: D40 [CR]	CS1	18. 40 RECORDS ARE READ IN FROM CS1
		PRT	19. PRINTS: ?
KB	20. ENTER: P38-38 [CR]	PRT	21. PRINTS:
		0031 JMP MSG3	**SYNTACTIC ERROR**
	23. ENTER: C38-38 [CR]		
	TAB JMP TAB OMSG3 TAB	NUMBER NOT FOUND, PRT ERR MSG. [CR] [CR]	
			24. PRINTS?
	25. ENTER: Q [CR]	CS1	26. READS IN REST OF MODULE 1
		CS2	27. OUTPUTS TO BC2 BUFFER WITH CORRECTION AND REST OF MODULE 1
		PRT	28. PRINTS: END EDIT TERMINATE/CONTINUE
KB	29. ENTER: T [CR]	PRT	30. PRINTS: . (MONITOR PROMPTING CHAR)

C. TEXT EDIT (CONTINUE)

DEVICE	INSTRUCTIONS	DEVICE	RESPONSE
CS1	31. UNLOAD TEST SOURCE PROGRAM		
	32. UNLOAD BC2; LABEL CASSETTE: SOURCE MODEL 1 CORRECTED		
	33. PROCEED TO NEXT STEP: D).		

D. REASSEMBLY:

BC2: SOURCE MODULE 1 CORRECTED IS ASSEMBLED AND BC3: OBJECT MODULE 1 CORRECTED, IS CREATED.

DEVICE	INSTRUCTIONS	DEVICE	RESPONSE
CS1	1. LOAD AND READY PX9ASM CASSETTE	733ASR	2. CS1 READY LED ON
KB	3. ENTER: LU [CR]		4. ASSEMBLER LOADED IN
		PRT	5. PRINTS: .(MONITOR PROMPTING CHAR)
CS1	6. UNLOAD PX9ASM AND RETURN TO RACK		
CS1	7. LOAD AND READY CASSETTE LABELED SOURCE MODULE 1 CORRECTED	733ASR	8. CS1 READY LED ON
CS2	9. LOAD AND READY BC3 (BLANK CASSETTE)	733ASR	10. CS2 READY LED ON
KB	11. ENTER: EX [CR]	PRT	12. PRINTS: PX9ASM VERSION 15 MARCH 76 ADD 4K MEM BLOCKS CONFIGURED?
KB	13. ENTER: 0 [CR]	PRT	14. PRINTS: PREDEFINED REGISTERS?
KB	15. ENTER: Y [CR]	PRT	16. PRINTS: ASM/TERM?
KB	17. ENTER: A [CR]		18. ASSEMBLY COMMENCES
		PRT	19. ASSEMBLY LISTING PRINTED
		CS2	20. OBJECT MODULE OUTPUT TO BC3
		PRT	21. PRINTS: ASM/TERM?
KB	22. ENTER: T [CR]	PRT	23. PRINTS: . (MONITOR PROMPTING CHAR)
CS1	24. UNLOAD BC3; LABEL CASSETTE: OBJECT MODULE 1 CORRECTED		
	25. PROCEED TO STEP: E		

```
.LU
.EX

PX9ASM  VERSION 1.0  15MAR76
ADD 4K MEM BLOCKS CONFIGURED? 0
PREDEFINED REGISTERS? Y

ASM/TERM? A
```

```
0003                        IDT  'TDRIVE'
0004                *  TITLE:   TDRIVE
0005                *           TRANSLATION DRIVER
0006                *  REVISION: 03/15/76
0007                *           ORIGINAL
0008                *  COMPUTER: 990,ASSEMBLY
0009                *  ABSTRACT:
0010                *           THIS IS THE MAIN PROGRAM: IT CALLS THE*
0011                *           REQUIRED SUBPROGRAMS TO OUTPUT MSGS AND *
0012                *           PROCESS INPUT DATA.  THE PROGRAM REQUEST *
0013                *           A NUMBER BE ENTERED AND ACCEPTS BOTH DIGITS *
0014                *           AND NAMES OF NUMBERS (1-10) WRITTEN IN *
0015                *           FIVE DIFFERENT LANGUAGES:  ENGLISH, SPANISH *
0016                *           GERMAN, RUSSIAN AND DIGITS.*
0017                *           IT RETURNS THE NAME OF THE LANGUAGE THE *
0018                *          NUMBER WAS WRITTEN IN AND ITS VALUE. *
0019                *           NEXT IT ALLOWS YOU TO TRANSLATE THE *
0020                *           NUMBER OF ANY OF THE 5 LANGUAGES. *
0021                *           FINALLY IT REQUEST WHETHER YOU WISH TO *
0022                *           CONTINUE, AND IF SO, ALLOWS YOU TO ENTER *
0023                *           ANOTHER NUMBER AND REPEATS THE ENTIRE *
0024                *           PROCESS.  ERROR MSGS ARE PRINTED FOR *
0025                *           UNRECOGNIZABLE NUMBERS AND LANGUAGES *
0026                *
0027  0000          WD    BSS  32
0028                      DXOP SVC,15
0029                      REF  OUTIDT,OUTPRC,MSG1,TSKIDT
0030                      REF  INPUT,P1IDT,P2IDT,INPRC1
0031                      REF  INPRC2,MSG2,MSG3,MSG4,MSG5,MSG6
0032                      REF  MSG7,ENDPT,P3IDT,INPRC3
0033  0020  02E0    START LWPI WD
      0022  0000'
0034  0024  0420          BLWP @OUTPRC              ENTER NUMBER
      0026  0000
0035  0028  0000          DATA MSG1
0036  002A  2FE0          SVC  @INPUT               INPUT NUMBER EG. EINS
      002C  0000
0037  002E  0420          BLWP @INPRC1              PROCESS NUMBER ENTERED
      0030  0000
0038  0032  10--          JMP  DMSG3                NUMBER NOT FOUND, PRT ERR MSG.
0039  0034  0420          BLWP @OUTPRC
      0036  0026'
0040  0038  0000          DATA MSG2
0041                *  NO WAS WRITTEN IN "GERMAN",ITS VALUE IS "2"
0042  003A  0420          BLWP @OUTPRC
      003C  0036'
0043  003E  0000          DATA MSG5                 ENTER LANGUAGE DESIRED
0044  0040  2FE0          SVC  @INPUT
      0042  002C'
0045  0044  0420          BLWP @INPRC2              PROCESS LANGUAGE ENTER VALIDAT
      0046  0000
0046  0048  10--          JMP  DMSG6                INVALID LANGUAGE GO TO WRITE L
0047  004A  0420          BLWP @OUTPRC              "GERMAN" NAME FOR THIS NO IS "
      004C  003C'
```

```
  0048  004E  0000          DATA MSG7
  0049  0050  0420  DMSG4   BLWP @OUTPRC
        0052  004C'
  0050  0054  0000          DATA MSG4
  0051  0056  2FE0          SVC  @INPUT          ENTER Y OR N!
        0058  0042'
  0052  005A  0420          BLWP @INPRC3
        005C  0000
  0053  005E  10E0          JMP  START
  0054  0060  2FE0          SVC  @ENDPT
        0062  0000
  0055  0064  0420  DMSG3   BLWP   @OUTPRC
        0066  0052'
        0032**1018
  0056  0068  0000          DATA MSG3
  0057  006A  10F2          JMP  DMSG4
  0058  006C  0420  DMSG6   BLWP   @OUTPRC
        006E  0066'
        0048**1011
  0059  0070  0000          DATA MSG6
  0060  0072  10EE          JMP  DMSG4
  0061                      END  START

0000 ERRORS

ASM/TERM? T
```

E. LINK AND LOADING

THE LAL OVERLAY WILL BE LOADED AND THE COMMAND (L.L.) ISSUED TO LINK AND LOAD THE 6 OBJECT MODULES. THE CORRECTED OBJECT MODULE IS LINK AND LOADED FIRST, THEN THE OTHER OBJECT MODULES ARE LINKED AND LOADED.

DEVICE	INSTRUMENT	DEVICE	RESPONSE
CS1	1. LOAD AND READY LAL CASSETTE	733ASR	1. CS1 READY LED ON
KB	2. ENTER: OV [CR]		3. LAL LOADED INTO MONITOR OVERLAY AREA
		PRT	4. PRINTS: LL
		PRT	5. PRINTS: . (MONITOR PROMPTING CHARACTER)
CS1	6. UNLOAD LAL AND RETURN TO RACK		
CS1	7. LOAD AND READY BC3, CORRECTED OBJ MODULE 1		8. CS1 READY LED ON
KB	9. ENTER: LL [CR]	PRT	10. PRINTS: LD PT?
KB	11. ENTER: [CR]	PRT	12. PRINTS: LD BI?
KB	13. ENTER: 100 [CR]	PRT	14. PRINTS: LIST OPT?
	15. ENTER: F [CR]		16. PRINTS: LOAD/END?
	17. ENTER: L [CR]		18. PRINTS: TDRIVE 0100
			19. PRINTS: LOAD/END?
CS1	20. UNLOAD BC3		
	21. LOAD AND READY BC1, OBJ MODULES, BAD MODULE 1	733ASR	22. CS1 READY LED ON
KB	23. ENTER: L [CR]		24. PRINTS: **LL 05** (IGNORE)
			25. PRINTS: LAOD/END?
	26. ENTER: L [CR]		27. PRINTS: LAOD/END?
	28. ENTER: L [CR]	PRT	29. PRINTS: TSKIDT 0174 <LIST OF DEF's>
			30. PRINTS: LOAD/END?
KB	31. ENTER: L [CR]		
	32. REPEAT THE ABOVE RESPONSE UNTIL P3IDT.	PRT	33. PRINTS: LOAD/END?
KB	34. ENTER: E [CR]	PRT	35. PRINTS: ENTRY = 0120 TERM/CONT?
KB	36. ENTER: T [CR]	PRT	37. PRINTS: . (MONITOR PROMPTING CHARACTER)
KB	38. UNLOAD BC1		
	39. PROCEED TO STEP: F		

```
 OV
LL
.LL

LD PT?
LD BI? 100
F/P LIST? F

LOAD/END? L

      TDRIVE   0100
LOAD/END? L
```

```
        TSKIDT  0174
              * NSAVE   0174
              * MSG1    023E
              * MSG2    0250
              * LANGUA  0268
              * DIGIT   027E
              * MSG3    0286
              * MSG4    02AA
              * MSG5    02D0
              * MSG6    02EC
              * MSG7    0306
              * YAZEEK  0308
              * NUMBER  0328
              * NUMS    032F
              * LNGTH   03E9
              * TRANS   041B
              * YES     0443
              * NO      0444
              * OUTPUT  0446
              * PRNBUF  044C
              * OUTCNT  0450
              * ENDPT   0456
              * INPUT   0458
              * INBUF   0468
              * BTD     046F
              * DEC     0471
              * INCNT   0462
LOAD/END? L

        OUTIDT  0478
              * OUTPRC  0498
              * CLEAR   04AE
LOAD/END? L

        P1IDT   04C0
              * INPRC1  04E0
LOAD/END? L

        P2IDT   055C
              * INPRC2  057C
LOAD/END? L

        P3IDT   060C
              * INPRC3  062C
LOAD/END? E
 ENTRY = 0120

TERM/CONT? T
```

```
.EX
ENTER NUMBER SIETE
NUMBER WAS WRITTEN IN SPANISH, ITS VALUE IS  00007
ENTER LANGUAGE DESIRED ENGLISH
ENGLISH NAME FOR THIS NUMBER IS SEVEN
DO YOU WISH TO CONTINUE? (Y OR N)Y
ENTER NUMBER ZWEI
NUMBER WAS WRITTEN IN GERMAN , ITS VALUE IS  00002
ENTER LANGUAGE DESIRED RUSSIAN
RUSSIAN NAME FOR THIS NUMBER IS DVA
DO YOU WISH TO CONTINUE? (Y OR N)Y
ENTER NUMBER MANY
NUMBER WAS NOT PROPERLY WRITTEN
DO YOU WISH TO CONTINUE? (Y OR N)Y
ENTER NUMBER FIVE
NUMBER WAS WRITTEN IN ENGLISH, ITS VALUE IS  00005
ENTER LANGUAGE DESIRED FRENCH
LANGUAGE NOT AVAILABLE
DO YOU WISH TO CONTINUE? (Y OR N)N
.
```

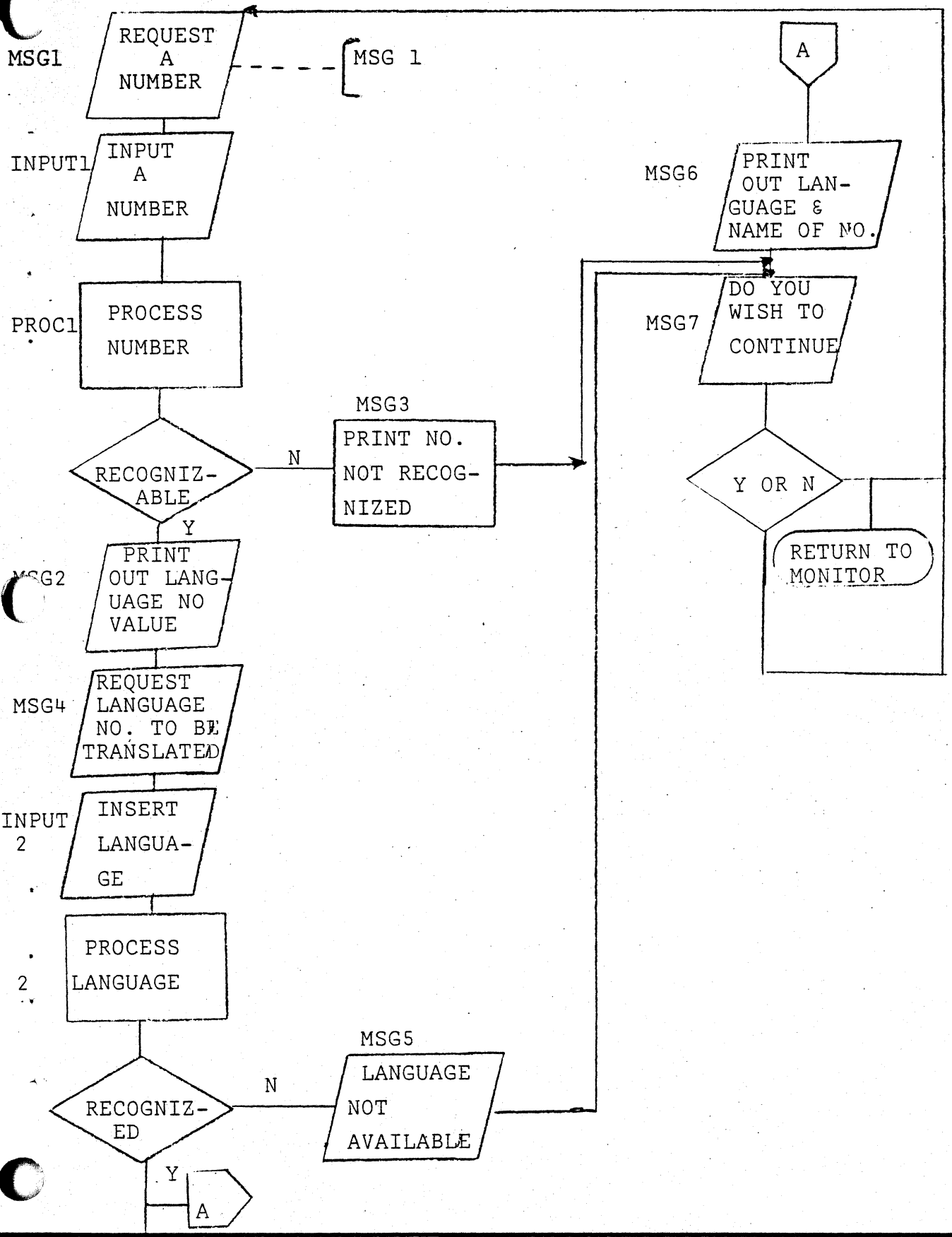
F. EXECUTION

WITH THE PROGRAM LOADED AN EXECUTION STATEMENT (EX) WILL HAND OVER CONTROL TO THE SAMPLE PROGRAM. THE PROGRAM IS INTERACTIVE AND WILL BE TESTED TO SEE IF IT IS FUNCTIONING CORRECTLY.

DEVICE		
KB	1.	ENTER: EX [CR]
PRT	2.	PRINTS: ENTER NUMBER
KB	3.	ENTER: SIETE [CR]
PRT	4.	RESPONSE: NUMBER WAS WRITTEN IN SPANISH, ITS VALUE IS +00007 ENTER LANGUAGE DESIRED
KB	5.	ENTER:
	6.	RESPONSE: ENGLISH NAME FOR THIS NUMBER IS SEVEN. DO YOU WISH TO CONTINUE? (Y OR N)
	7.	ENTER: Y
	8.	RESPONSE: ENTER NUMBER
	9.	ENTER: ZWEI [CR]
	10.	RESPONSE: NUMBER WAS WRITTEN IN GERMAN, ITS VALUE IS +00002 ENTER LANGUAGE DESIRED
	11.	ENTER: RUSSIAN
	12.	RESPONSE: RUSSIAN NAME FOR THIS NUMBER IS DVA. DO YOU WISH TO CONTINUE? (Y OR N)
	14.	ENTER: Y
	15.	RESPONSE: ENTER NUMBER
	16.	ENTER: MANY
	17.	RESPONSE: NUMBER WAS NOT PROPERLY WRITTEN DO YOU WISH TO CONTINUE? (Y OR N)
	18.	ENTER: Y [CR]
	20.	RESPONSE: ENTER NUMBER
	21.	ENTER: FIVE [CR]
	22.	RESPONSE: NUMBER WAS WRITTEN IN ENGLISH, ITS VALUE IS +00005 ENTER LANGUAGE DESIRED
	23.	ENTER: FRENCH [CR]
	24.	RESPONSE: LANGUAGE NOT AVAILABLE DO YOU WISH TO CONTINUE? (Y OR N)
	25.	ENTER: N [CR]
	26.	RESPONSE: .(MONITOR PROMPTING CHAR)
	27.	PROGRAM RELINQUISHED CONTROL TO THE MONITOR PROGRAM

THIS COMPLETES OUR TEST OF THE MODEL 990 OPERATING PROCEDURES

SAMPLE PROGRAM
SYSTEM FLOWCHART



APPLICATION		REVISIONS			
NEXT ASSY	USED ON	LTR	DESCRIPTION	DATE	APPROVED
	7506				

REV																														
SHEET																														
REV STATUS OF SHEETS	REV																													
	SHEET																													

UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES
TOLERANCES
ANGLES ±1°
3 PLACE DECIMAL ± 010
2 PLACE DECIMAL ± 02

IDENTIFYING NUMBERS SHOWN IN PARENTHESES [illegible] REFERENCE ONLY

[illegible]PRET DWG IN [illegible]ORDANCE WITH MIL-STD-100

DWN DATE
CHK
ENGR
QA
APVD
CONTR NO
DESIGN ACTIVITY RELEASE

TEXAS INSTRUMENTS
INCORPORATED
Equipment Group Dallas, Texas

[illegible]PD, SAMPLE PROGRAM 2-PX990

SIZE	CODE IDENT NO	DRAWING NO
A	96214	945329-9901

SCALE	REV.	SHEET 1 of 21

TI-9983.C

1.0 INTRODUCTION

In this example, the main functions of the 990-733 ASR System Software are illustrated including assembling, linking and loading, debugging and editing. It is assumed that the user has read the manual and has been introduced to these functions. Included with each step in this example is a brief explanation of procedure, a listing of the actual procedure followed, and a reference to the section in the manual where more information can be found.

2.0 CONCORDANCE PROGRAM

This program creates concordance of all the symbols used in a program. The user may specify labels, operators, and/or operands to be included in the concordance printout. To run the program, follow these steps:

1. Mount and ready the source tape in cassette drive 1 (the left-hand drive).
2. Execute the program using the EX command or using the RU command for debugging.

In the concordance program, an error is included in the print routine (PRTBM) so that the user may be exposed to the process involved in creating and debugging a working program. The steps in the process are:

1. Assemble the source programs and create object modules using PX9ASM.
2. Link and load the object modules into user memory using PX9LAL.
3. Using the monitor, debug the program.
4. Using PX9EDT the source edit source module which contains the

2.0 (continued)

error.

5. Reassemble the source module.
6. Link and load all of the modules into memory again.
7. Execute the final program to see that the error has been corrected and the program executes correctly.

3.0 ASSEMBLING MODULES WITH PX9ASM

The first part of the program is the assembly of modules using PX9ASM. For a description of how to use the assembler, refer to Section VII. The assembler must first be loaded using the LU command (Section V). Predefined register definitions are not required; therefore, in answer to the question:

PREDEFINED REGISTERS?

Enter "N"

```
.LU 8

.EX


PX9ASM   VERSION     DATE
ADD 4K MEM BLOCKS CONFIGURED? 0
PREDEFINED REGISTERS? N

ASM/TERM? A
```

4.0 LOADING MODULES WITH PX9LAL

The second part of the program is the loading of modules. Using PX9LAL, link and load the object modules into memory. (Refer to the software loading procedures in Section IV.). Before PX9LAL can be used, it must be loaded into the monitor transient area using the OV command (Section V). When PX9LAL asks for

LD PT?

LD BI?

enter a carriage return after each to specify the default values of 0 and $A0_{16}$, respectively. In answer to

LIST OPT?

enter either F (full) or P (partial). The object modules may be loaded from either cassette drive. When PX9LAL prints

LOAD/END?

enter either L or L7 to load from cassette 1 or L8 to load from cassette 2.

When all of the modules have been loaded, the program entry point is printed. This value is placed in the user's PC register.

```
.LL

LD PT?
LD BI?
F/P LIST? F

LOAD/END? L

       XREF    00A0
             * PRINTC  0136
             * GETCHR  0144
             * TERM    0168
LOAD/END? L

       PARSEM  024A
             * PARSE   02AC
             * DEFPR   0406
             * OPNDPR  040A
             * OPERPR  0408
             * STMT    03FC
LOAD/END? L

       CTYPM   046C
             * CTYP    04B4
LOAD/END? L

       PRTBM   04FE
             * PRTB    04FE
LOAD/END? L

       CSYMM   0648
             * CSYM    0648
             * ISYM    06AA
             * NXTLOC  06BC
             * ENDSYM  1DCC
             * FSTSYM  06BE
LOAD/END? L

       SYMREFM 1DD8
             * SYMREF  1DD8
             * OVFL    1E0E
LOAD/END? L

       SYMDEFM 1E2E
             * SYMDEF  1E2E
LOAD/END? E
 ENTRY = 00A0

TERM/CONT? T
```

5.0 DEBUGGING THE PROGRAM

The third part of the program is the debugging. Execute the program using the EX command (Section V). Use the source module named SYMDFM as the input to the concordance program. (This is the shortest source module).

The source tape may be positioned to the beginning of SYMDFM by setting the PLAYBACK switch on the data terminal to the LOCAL position and the PRINTER switch of the OFF position. By pressing the CONT START switch in the Playback Control area on the data terminal upper switch panel, the tape will be read to an end-of-file marker and positioned at the beginning of the next file. Repeat this process until the tape is correctly psoitioned to SYMDFM Set the PLAYBACK and PRINTER switches to the LINE position.

Follow the debugging procedure outlined in the computer printout in this paragraph. For descriptions of the individual monitor keyboard commands, refer to Section V.

[MOUNT SOURCE FOR SYMDFM ON CASSETTE DRIVE. EXECUTE PROGRAM.]

```
IR
PC=00A0  WP=0000  ST=0000
.EX
CROSS REFERENCE - DEC 31,1975
PROCESS LABELS? Y
PROCESS OPERATORS(INSTRUCTIONS)? Y
PROCESS OPERANDS? Y



 ** ** **  CROSS REFERENCE  ** ** 990 ** ** **

SYMBOL  DEF      REFS
AI              0038
BB              0040
TBL             0037
CSYM            0036    0037
ODEF            0024
HEND            0041
!EQU            0026    0027    0028    0029    0031    0033
LIDT            0003
 INSE           0002
LMOV            0034    0035    0039
>PAGE           0001    0030
<R11    0029    0034
>R3     0026    0035    0039
>R4     0027    0034    0040
DR9     0028    0035    0038    0039



REF             0036
,SMDE   0031    0038


SYMD  0033    0024
\SYMD           0003
@TITL           0002
```

[INCORRECT OUTPUT. INSPECT OUTPUT BUFFER BEFORE EACH RECORD WRITTEN.
SET BREAKPOINT AT INSTRUCTION BEFORE SUPERVISOR CALL TO WRITE RECORD.]

```
HA 4FE AC
SUM=05AA   01450  DIFF=0452   01106
.SB 1,5AA
.MR

PC=00A0
WP=0000
ST=0000  2
.
```

[REPOSITION TAPE TO SYMDEF.]

```
RU
CROSS REFERENCE - DEC 31,1975
PROCESS LABELS? Y
PROCESS OPERATORS(INSTRUCTIONS)? Y
PROCESS OPERANDS? Y
```

```
 ** ** **  CROSS REFERENCE  ** ** 990 ** ** **

SYMBOL  DEF      REFS
BKPT#1
PC=05AA  WP=016A  ST=D402
```

[LOOK AT OUTPUT BUFFER.]

```
HA 4FE E8
SUM=05E8   01512  DIFF=0414   01044
.IM 5E8 620
05E8=081C  4149  2020  2020  >2020  2020  2020  2020
05F8=3030  3338  2020  2020  >2020  2020  2020  2020
0608=2020  2020  2020  2020  >2020  2020  2020  2020
0618=2020  2020  2020  2020  >2020
.
```

FIRST TWO CHARACTERS OF BUFFER INVALID. POINTER TO SYMBOL TABLE NEEDS TO BE INCREMENTED BY TWO. INSERT THE STMT AI R1,2 AFTER STMT 57. SINCE THERE IS NO ROOM TO INSERT THIS TWO WORD INSTRUCTION WE MUST MAKE A PATCH TO AN UNUSED PORTION OF MEMORY, INSERT THE INSTRUCTIONS (STMT 58 WHICH WE MUST OVERLAY WITH A BRANCH, AND STMT THAT WE ARE INSERTING), AND BRANCH BACK TO THE CODE WE CAME FROM.

TO DO THIS PATCH AT STMT 58:

```
        B    @PATCH          WHERE PATCH = 1F00
```

AT 1F00 INSERT:

```
        AI   R1,2
        LI   R0,OUTBUF
        B    @STMT_59        WHERE STMT_59 = 4FE+18=516
```

THE CODE FOR THE PATCHES IS:

```
        0460
        1F00

        0221
        0002
        0200
        05E8
        0460
        0516
```

```
MM 512
0512=0200  0460
0514=05E8  1F00
.MM 1F00
1F00=0000  0221
1F02=0000  2
1F04=0000  200
1F06=1E0E  5E8
1F08=4F56  460
1F0A=464C  516
.
```

BY ENTERING THE RU COMMAND, THE PROGRAM WILL CONTINUE EXECUTING FROM THE BREAKPOINT. THE RECORD WITH THE ERROR IN IT WILL BE PRINTED AND THE NEXT RECCORD BUILT. THE PROGRAM WILL HALT AT THE BREAKPOINT BEFORE PRINTING THE NEXT RECORD. WE CAN INSPECT THE BUFFER TO DETERMINE IF OUR PATCH WAS CORRECT.

```
RU
AI              0038
BKPT#1
PC=05AA  WP=016A  ST=D002
.IM 5E8 620
05E8=4220  2020  2020  2020  >2020  2020  2020  2020
05F8=3030  3430  2020  2020  >2020  2020  2020  2020
0608=2020  2020  2020  2020  >2020  2020  2020  2020
0618=2020  2020  2020  2020  >2020
.
```

THE BUFFER APPEARS CORRECT. REMOVE THE BREAKPOINT AND PRINT THE REST OF THE SYMBOL TABLE.

```
CB.1
.RU
B               0040
BL              0037
CSYM            0036    0037
DEF             0024
END             0041
EQU             0026    0027    0028    0029    0031    0033
IDT             0003
INSERT          0002
MOV             0034    0035    0039
PAGE            0001    0030
R11     0029    0034
R3      0026    0035    0039
R4      0027    0034    0040
R9      0028    0035    0038    0039
REF             0036
SMDEF   0031    0038
SYMDEF  0033    0024
SYMDFM          0003
TITL            0002
.
```

NOW THAT WE HAVE DETERMINED THAT THE PATCH IS CORRECT, RELOAD THE EDITOR AND INSERT THE MISSING STATEMENT, RE-ASSEMBLE, AND VERIFY THAT EVERYTHING IS CORRECT.

6.0 EDITING WITH PX9EDT

The fourth part of the program is editing using PX9EDT. For a description of how to use the text editor, refer to Section VI. The text editor must be loaded using the LU command (Section V).

```
.LU
.EX

PX9EDT  945394 ** 15MAR76
ADD 4K MEM BLOCKS CONFIGURED? 0

POSITION TAPES, ENTER CR


?D63
?P-15
0048 PRTB    EQU  $
0049         MOV  R11,R6              SAVE RETURN
0050         MOV  @FSTSYM,@NXTSYM     POINTER TO FIRST SYMBOL ENTRY
0051 *
0052 PRTB01  EQU  $                   **PRINT A SYMBOL
0053         JEQ  PRTBXT              IF DONE
0054         BL   @BLNKLN             SET OUTBUF TO BLANKS
0055         MOV  @NXTSYM,R2          MOVE SYMBOL TO BUFFER
0056         MOV  R2,R1
0057 *       AI   R1,SMSYM       REMOVED TO CREATE ERROR
0058         LI   R0,OUTBUF
0059         MOV  *R1+,*R0+
0060         MOV  *R1+,*R0+
0061         MOV  *R1+,*R0+
0062 *
?C57-57
AI R1,SMSYM                          CORRECTED ERROR
CR,CR
?P55-59
0055         MOV  @NXTSYM,R2          MOVE SYMBOL TO BUFFER
0056         MOV  R2,R1
             AI   R1,SMSYM                CORRECTED ERROR
0058         LI   R0,OUTBUF
0059         MOV  *R1+,*R0+
?Q
END EDIT
TERMINATE/CONTINUE?T
```

7.0 REASSEMBLING, RELINKING AND LOADING MODULES AND EXECUTING THE PROGRAM

The fifth part of the program is reassembly of the edited module. The sixth part of the program is the relinking and loading of all modules. The seventh part of the program is execution of the final version of the program.

.LU 7
.EX

PX9ASM VERSION DATE
ADD 4K MEM BLOCKS CONFIGURED? 0
PREDEFINED REGISTERS? N

ASM/TERM? A


```
                   IDT   'PRTBM'
          * TITLE:       PRTBM
          *              PRINT SYMBOL TABLE
          * REVISION:    03/15/76
          *              ORIGINAL
          * COMPUTER:    990,ASSEMBLY
          * ABSTRACT:
          *              PRTB WILL READ THE SYMBOL TABLE ONE SYMBOL
          *              AT A TIME AND PRINT THE SYMBOL NAME,
          *              THE STATEMENT NUMBER WHERE THE SYMBOL
          *              WAS DEFINED, AND THE LIST OF STATEMENT
          *              NUMBERS WHERE THE SYMBOL WAS REFERENCED.
          * CALLING SEQUENCE:
          *              NO INPUT PARMS
          *
          *              REGISTERS DESTROYED - R0-R6,R9,R10
          *
          *
                   DEF   PRTB
                   REF   FSTSYM
          *
                   DXOP  SVC,15
          *
     0000 R0       EQU   0
     0001 R1       EQU   1
     0002 R2       EQU   2
     0003 R3       EQU   3
     0004 R4       EQU   4
     0005 R5       EQU   5
     0006 R6       EQU   6
     0009 R9       EQU   9
     000A R10      EQU   10
     000B R11      EQU   11
     0002 SMSYM    EQU   2
     0008 SMDEF    EQU   8
     000A SMREF    EQU   >A
     0002 REFVAL   EQU   2
     0004 SYSFLG   EQU   4                 PRB FLAGS
     0006 BFADR    EQU   6
     0008 BFLTH    EQU   8
     000A CCOUNT   EQU   10
```

```
0045                    *
0046                    *
0047                    *
0048          0000' PRTB    EQU   $
0049    0000  C18B          MOV   R11,R6              SAVE RETURN
0050    0002  C820          MOV   @FSTSYM,@NXTSYM     POINTER TO FIRST SYMBOL ENTRY
        0004  0000
        0006  ----
0051                    *
0052          0008' PRTB01  EQU   $                   **PRINT A SYMBOL
0053    0008  13--          JEQ   PRTBXT              IF DONE
0054    000A  06A0          BL    @BLNKLN             SET OUTBUF TO BLANKS
        000C  ----
0055    000E  C0A0          MOV   @NXTSYM,R2          MOVE SYMBOL TO BUFFER
        0010  ----
0056    0012  C042          MOV   R2,R1
0057    0014  0221          AI    R1,SMSYM                CORRECTED ERROR
        0016  0002
0058    0018  0200          LI    R0,OUTBUF
        001A  ----
0059    001C  CCB1          MOV   *R1+,*R0+
0060    001E  CCB1          MOV   *R1+,*R0+
0061    0020  CCB1          MOV   *R1+,*R0+
0062                    *
0063    0022  C062          MOV   @SMDEF(R2),R1       MOVE SYMBOL DEF TO OUTBUF
        0024  0008
0064    0026  0281          CI    R1,>FFFF              (IF IT EXISTS
        0028  FFFF
0065    002A  13--          JEQ   PRTB02
0066    002C  C281          MOV   R1,R10
0067    002E  0209          LI    R9,OUTBUF+8         CONVERT BIN TO DECIMAL
        0030  ----
0068    0032  06A0          BL    @CONV
        0034  ----
0069                    *
0070          0036' PRTB02  EQU   $                   PROCESS REFERENCES
        002A** 1305
0071    0036  C0A0          MOV   @NXTSYM,R2
        0038  ----
0072    003A  C162          MOV   @SMREF(R2),R5
        003C  000A
0073    003E  0203          LI    R3,OUTBUF+16
        0040  ----
0074    0042  0204          LI    R4,7                7 REFERENCES PER LINE
        0044  0007
0075          0046' PRTB03  EQU   $
0076    0046  C145          MOV   R5,R5               IF END OF REF CHAIN
0077    0048  13--          JEQ   PRTB05
0078                    *
0079    004A  C2A5          MOV   @REFVAL(R5),R10     OUTPUT REF TO LINE
        004C  0002
0080    004E  C243          MOV   R3,R9
0081    0050  06A0          BL    @CONV
        0052  ----
```

```
0082  0054  0223          AI    R3,8                NEXT LINE POSITION
      0056  0008
0083  0058  0604          DEC   R4                  IF LINE FULL AND MORE REFS REM
0084  005A  15--          JGT   PRTB04
0085  005C  C555          MOV   *R5,*R5
0086  005E  13--          JEQ   PRTB04
0087                     *
0088  0060  06A0          BL    @PRNTLN             PRINT CURRENT LINE
      0062  ----
0089  0064  06A0          BL    @BLNKLN             RESET LINE POINTERS
      0066  ----
0090  0068  0203          LI    R3,OUTBUF+16
      006A  ----
0091  006C  0204          LI    R4,7
      006E  0007
0092                     *
0093          0070' PRTB04 EQU   $
      005A**150A
      005E**1308
0094  0070  C155          MOV   *R5,R5              CHAIN TO NEXT REF
0095  0072  10E9          JMP   PRTB03
0096                     *
0097          0074' PRTB05 EQU   $
      0048**1315
0098  0074  06A0          BL    @PRNTLN             PRINT LAST LINE
      0076  ----
0099                     *
0100          0078' PRTB06 EQU   $
0101  0078  C0A0          MOV   @NXTSYM,R2          CHAIN TO NEXT SYMBOL
      007A  ----
0102  007C  C812          MOV   *R2,@NXTSYM
      007E  ----
0103  0080  10C3          JMP   PRTB01
0104                     *
0105  0082  0456  PRTBXT  B     *R6                 RETURN
      0008**133C
0106                     *
0107                     *
0108                     *     BLANK LINE BUFFER
0109                     *
0110                     *     REGISTERS USED - R0,R1,R2
0111                     *
0112          0084' BLNKLN EQU   $
      000C**0084'
      0066**0084'
0113  0084  0202          LI    R2,40
      0086  0028
0114  0088  0201          LI    R1,' '
      008A  2020
0115  008C  0200          LI    R0,OUTBUF
      008E  ----
0116          0090' BLNK01 EQU   $
0117  0090  CC01          MOV   R1,*R0+
0118  0092  0602          DEC   R2
0119  0094  15FD          JGT   BLNK01
```

```
0120  0096  045B          RT
0121                *
0122                *     STRIP TRAILING BLANKS AND PRINT OUTPUT LINE
0123                *
0124        0098' PRNTLN  EQU  $
      0062**0098'
      0076**0098'
0125  0098  0200          LI   R0,OUTBUF+79      LAST BUFFER POSITION
      009A  ----
0126        009C' PR1     EQU  $
0127  009C  9810          CB   *R0,@BLANK
      009E  ----
0128  00A0  16--          JNE  PR2
0129  00A2  0600          DEC  R0                ASSUME AT LEAST ONE CHAR IN BU
0130  00A4  10FB          JMP  PR1
0131        00A6' PR2     EQU  $
      00A0**1602
0132  00A6  0201          LI   R1,OUTBUF-3
      00A8  ----
0133  00AA  6001          S    R1,R0
0134  00AC  C800          MOV  R0,@WTCC          OUTPUT CHAR COUNT
      00AE  ----
0135  00B0  2FE0          SVC  @WTPRB            PRINT LINE
      00B2  ----
0136  00B4  045B          RT
0137                *
0138                **    CONVERT BINARY TO DECIMAL
0139                *
0140                *
0141                *       REGISTERS USED - R0,R1,R2
0142                *         R10 - VALUE TO BE CONVERTED
0143                *         R9  - POINTER TO BUFFER FOR RESULT
0144                *
0145        00B6' CONV    EQU  $
      0034**00B6'
      0052**00B6'
0146  00B6  C08A          MOV  R10,R2            USE REPEATED DIVIDE
0147  00B8  0200          LI   R0,1000           AND LOOK UP QUOTIENT IN TABLE
      00BA  03E8
0148  00BC  04C1          CLR  R1
0149  00BE  3C40          DIV  R0,R1
0150  00C0  DE61          MOVB @CLIST(R1),*R9+
      00C2  ----
0151  00C4  0200          LI   R0,100
      00C6  0064
0152  00C8  04C1          CLR  R1
0153  00CA  3C40          DIV  R0,R1
0154  00CC  DE61          MOVB @CLIST(R1),*R9+
      00CE  ----
0155  00D0  04C1          CLR  R1
0156  00D2  0200          LI   R0,10
      00D4  000A
0157  00D6  3C40          DIV  R0,R1
0158  00D8  DE61          MOVB @CLIST(R1),*R9+
      00DA  ----
```

```
  0159  00DC  DE62          MOVB @CLIST(R2),*R9+   REMAINDER IS LAST DIGIT
        00DE  ----
  0160  00E0  045B          RT
  0161                    *
  0162                    *
  0163  00E2    30  CLIST   TEXT '0123456789'      CONVERT BIN TO DEC
        00C2**00E2'
        00CE**00E2'
        00DA**00E2'
        00DE**00E2'
  0164  00EC  0A0D          DATA >0A0D            CR,LF
  0165  00EE          OUTBUF BSS  80
        001A**00EE'
        0030**00F6'
        0040**00FE'
        006A**00FE'
        008E**00EE'
        009A**013D'
        00A8**00EB'
  0166  013E  0000  WTPRB   DATA 0,>B00,0,OUTBUF-2,80  OUTPUT PRB
        0140  0B00
        0142  0000
        0144  00EC'
        0146  0050
        00B2**013E'
  0167  0148  0000  WTCC    DATA 0                OUTPUT CHAR COUNT
        00AE**0148'
  0168                    *
  0169  014A    20  BLANK   BYTE ' ',0
        014B    00
        009E**014A'
  0170  014C  0000  NXTSYM  DATA 0
        0006**014C'
        0010**014C'
        0038**014C'
        007A**014C'
        007E**014C'
  0171                      END

0000 ERRORS

ASM/TERM? T
```

OV
LL

NEXT LOAD AND READY SAMPLE 2 OBJECT IN CS1, AND LOAD AND READY CORRECTED OBJECT MODULE 'PRTBM' IN CS2.

NOTE, AFTER LOADING OBJECT MODULES XREF, PARSEM, AND CTYPM, FROM CS1, CORRECTED MODULE PRTBM MUST BE LOADED FROM CS2. NEXT, THE OBJECT CASSETTE (CS1) MUST BE POSITIONED PAST THE BAD OBJECT MODULE PRTBM FOLLOWING INSTRUCTIONS PAGE 7. THE REMAINDER OF OBBJECT MODULES WILL BE LOADED FROM CS1.

```
LL

LD PT?
LD BI?
F/P LIST? P

LOAD/END? L

      XREF     00A0
LOAD/END? L

      PARSEM   024A
LOAD/END? L

      CTYPM    046C
LOAD/END? L8

      PRTBM    04FE
LOAD/END? L

      CSYMM    064C
LOAD/END? L

      SYMRFM   1DDC
LOAD/END? L

      SYMDFM   1E32
LOAD/END? E
 ENTRY = 00A0

TERM/CONT? T
```

```
IR
PC=00A0  WP=0000  ST=0000
.EX
CROSS REFERENCE - DEC 31,1975
PROCESS LABELS? Y
PROCESS OPERATORS(INSTRUCTIONS)? Y
PROCESS OPERANDS? Y



   ** ** **  CROSS REFERENCE  ** ** 990 ** ** **


   SYMBOL  DEF      REFS
   AI              0038
   B               0040
   BL              0037
   CSYM            0036    0037
   DEF             0024
   END             0041
   EQU             0026    0027    0028    0029    0031    0033
   IDT             0003
   INSERT          0002
   MOV             0034    0035    0039
   PAGE            0001    0030
   R11     0029    0034
   R3      0026    0035    0039
   R4      0027    0034    0040
   R9      0028    0035    0038    0039
   REF             0036
   SMDEF   0031    0038
   SYMDEF  0033    0024
   SYMDFM          0003
   TITL            0002
```