

IMP-E

#302

82 SEC. AVGD VECTORS
67-070A-03C
(RESTORED)

5.12 ~~SEC~~ SEC. MAG. VECTORS
67-070A-03D
(RESTORED)

20000000/

62 SEC AVGD VECTORS REMOVED

Table of Contents

1. Introduction
2. Errata/Change Log
3. LINKS TO RELEVANT INFORMATION IN THE ONLINE NSSDC
INFORMATION SYSTEM
4. Catalog Materials
 - a. Associated Documents
 - b. Core Catalog Materials

1. INTRODUCTION:

The documentation for this data set was originally on paper, kept in NSSDC's Data Set Catalogs (DSCs). The paper documentation in the Data Set Catalogs have been made into digital images, and then collected into a single PDF file for each Data Set Catalog. The inventory information in these DSCs is current as of July 1, 2004. This inventory information is now no longer maintained in the DSCs, but is now managed in the inventory part of the NSSDC information system. The information existing in the DSCs is now not needed for locating the data files, but we did not remove that inventory information.

The offline tape datasets have now been migrated from the original magnetic tape to Archival Information Packages (AIP's).

A prior restoration may have been done on data sets, if a requestor of this data set has questions; they should send an inquiry to the request office to see if additional information exists.

2. ERRATA/CHANGE LOG:

NOTE: Changes are made in a text box, and will show up that way when displayed on screen with a PDF reader.

When printing, special settings may be required to make the text box appear on the printed output.

Version	Date	Person	Page	Description of Change
01				
02				

3 LINKS TO RELEVANT INFORMATION IN THE ONLINE NSSDC
INFORMATION SYSTEM:

<http://nssdc.gsfc.nasa.gov/nmc/>

[NOTE: This link will take you to the main page of the NSSDC Master Catalog. There you will be able to perform searches to find additional information]

4. CATALOG MATERIALS:

- a. Associated Documents To find associated documents you will need to know the document ID number and then click here.
<http://nssdcftp.gsfc.nasa.gov/miscellaneous/documents/>

- b. Core Catalog Materials

IMP-E

82 SEC. AVERAGED VECTORS

3C-SPHE-00384

5.12 SEC. MAG. VECTORS

3D-SPHE-00616

MERGED NEBB/SONNET

3F-SPHE-00343

67-070A-03C, D, F

THIS CATALOG CONTAINS THREE DATA SETS. THESE DATA SETS HAVE BEEN RESTORED. ORIGINALLY DATA SET 67-070A-03C CONTAINED 20 7-TRACK, 556 BPI TAPES WRITTEN IN BINARY, THERE ARE NOW FIVE RESTORED TAPES WRITTEN IN BINARY. ORIGINALLY DATA SET 67-070A-03D CONTAINED 210 7-TRACK, 556 BPI TAPES WRITTEN IN BCD, THERE ARE NOW 22 RESTORED TAPES WRITTEN IN ASCII. ORIGINALLY 67-070A-03F CONTAINED 20 9-TRACK, 1600 BPI TAPES WRITTEN IN BINARY, THERE IS NOW ONE RESTORED TAPE WRITTEN IN BINARY. 67-070A-03C AND 03D WERE PACKED FROM 6-BIT BYTES INTO 8-BIT BYTES. 67-070A-03C AND 03D WERE CREATED ON AN IBM 7094 AND 67-070A-03F WAS CREATED ON AN IBM 360. IN DETERMINING THE START AND STOP TIMES CONSIDER JANUARY 1 TO BE DAY ZERO. FOR 67-070A-03C AND 03D THE DR AND DS TAPES ARE 9-TRACK, 6250 BPI. FOR 67-070A-03F THE DR TAPE IS ON A 3480 CARTRIDGE AND THE DS TAPE IS 9-TRACK, 6250 BPI. THE DR AND DS NUMBERS ALONG WITH THE CORRESPONDING D NUMBERS AND THE TIME SPANS ARE AS FOLLOWS:

Imp E

82 Second Averaged Vectors

67-070A-03C

This data set has been restored. There were originally 20 7-track, 556 BPI tapes written in binary on an IMB 7094 computer. There are now 5 restored tapes maintaining the binary format, packing 6-bit bytes into 8-bit bytes. The DR and DS tapes are 9-track, 6250 BPI tapes. When determining start and stop times, consider January 1 to be day zero. The DR and DS numbers along with the corresponding D numbers and time spans follow.

DR	DS	D	Files	Time Span
DR000850	DS000850	41		07/19/67 - 04/25/68
		D-017537	1-10	07/19/67 - 09/27/67
		D-017540	11-20	09/28/67 - 12/07/67
		D-017532	21-30	12/06/67 - 02/14/68
		D-017533	31-41	02/14/68 - 04/25/68
DR000851	DS000851	44		04/25/68 - 01/30/69
		D-017525	1-11	04/25/68 - 07/04/68
		D-017534	12-21	07/04/68 - 09/12/68
		D-017535	22-33	09/12/68 - 11/21/68
		D-017538	34-44	11/21/68 - 01/30/69
DR000852	DS000852	44		01/30/69 - 11/05/69
		D-017536	1-11	01/30/69 - 04/09/69
		D-017539	12-22	04/10/69 - 06/19/69
		D-017531	23-33	06/19/69 - 08/28/69
		D-017541	34-44	08/28/69 - 11/05/69
DR000853	DS000853	44		11/06/69 - 08/13/70
		D-017530	1-12	11/06/69 - 01/15/70
		D-017529	13-24	01/15/70 - 03/26/70
		D-017528	25-34	03/26/70 - 06/04/70
		D-017542	35-44	06/04/70 - 08/13/70
DR000854	DS000854	42		08/13/70 - 12/30/71
		D-017527	1-10	08/13/70 - 10/22/70
		D-017526	11-19	10/22/70 - 12/24/70
		D-017543	20-33	07/29/71 - 10/28/71
		D-018983	34-42	10/28/71 - 12/30/71

67-070A-03C

DR#	DS#	D#	FILES	TIME SPAN
DR00850	DS00850	D17527	1-10	07/19/67 - 09/27/67
		D17540	11-20	09/29/67 - 12/07/67
		D17532	21-30	12/06/67 - 02/14/68
		D17533	31-41	02/14/68 - 04/25/68
DR00851	DS00851	D17525	1-11	04/25/68 - 07/04/68
		D17534	12-21	07/04/68 - 09/12/68
		D17535	22-33	09/12/68 - 11/21/68
		D17538	34-44	11/21/68 - 01/30/69
DR00852	DS00852	D17536	1-11	01/30/69 - 04/09/69
		D17539	12-22	04/10/69 - 06/19/69
		D17531	23-33	06/19/69 - 08/28/69
		D17541	34-44	08/28/69 - 11/05/69
DR00853	DS00853	D17530	1-12	11/06/69 - 01/15/70
		D17529	13-24	01/15/70 - 03/26/70
		D17528	25-34	03/26/70 - 06/04/70
		D17542	35-44	06/04/70 - 08/13/70
DR00854	DS00854	D17527	1-10	08/13/70 - 10/22/70
		D17526	11-19	10/22/70 - 12/24/70
		D17543	20-33	07/29/71 - 10/28/71
		D18983	34-42	10/29/71 - 12/30/71

67-070A-03D

DR01016	DS01016	D17774	1	07/19/67 - 07/27/67
		D17782	2	07/27/67 - 08/03/67
		D17783	3	08/03/67 - 08/10/67
		D17772	4	08/10/67 - 08/17/67
		D17784	5	08/17/67 - 08/24/67
		D17785	6	08/24/67 - 08/30/67
		D17786	7	08/31/67 - 09/07/67
		D17787	8	09/07/67 - 09/14/67
		D17819	9	09/14/67 - 09/21/67
DR01017	DS01017	D17788	1	09/21/67 - 09/27/67
		D17773	2	09/28/67 - 10/04/67
		D17789	3	10/05/67 - 10/12/67
		D17777	4	10/12/67 - 10/19/67
		D17776	5	10/19/67 - 10/25/67
		D17815	6	10/26/67 - 11/02/67
		D17775	7	11/02/67 - 11/09/67
		D17778	8	11/09/67 - 11/15/67
		D17779	9	11/16/67 - 11/23/67

67-070A-000

DR#	DS#	D#	FILES	TIME SPAN
DR01018	DS01018	D17780	1	11/23/67 - 11/29/67
		D17781	2	11/30/67 - 12/07/67
		D17818	3	12/07/67 - 12/14/67
		D17817	4	12/14/67 - 12/20/67
		D17816	5	12/21/67 - 12/28/67
		D17795	6	01/04/68 - 01/11/68
		D17794	7	01/11/68 - 01/17/68
		D17792	8	01/18/68 - 01/25/68
		D17793	9	01/25/68 - 01/31/68
DR01019	DS01019	D17823	1	02/01/68 - 02/08/68
		D17822	2	02/08/68 - 02/15/68
		D17821	3	02/15/68 - 02/22/68
		D17801	4	02/29/68 - 03/07/68
		D17812	5	03/07/68 - 03/13/68
		D17808	6	03/14/68 - 03/21/68
		D17800	7	03/21/68 - 03/28/68
		D17799	8	03/28/68 - 04/04/68
		D17807	9	04/04/68 - 04/11/68
DR01020	DS01020	D17798	1	04/11/68 - 04/18/68
		D17797	2	04/18/68 - 04/25/68
		D17820	3	04/25/68 - 05/02/68
		D19218	4	04/25/68 - 05/02/68
		D19219	5	05/02/68 - 05/09/68
		D19220	6	05/09/68 - 05/16/68
		D19221	7	05/16/68 - 05/23/68
		D19222	8	05/23/68 - 05/30/68
		D19223	9	05/30/68 - 06/06/68
		D19224	10	06/06/68 - 06/13/68
DR01021	DS01021	D19225	1	06/13/68 - 06/20/68
		D19226	2	06/20/68 - 06/27/68
		D19227	3	06/27/68 - 07/04/68
		D19228	4	07/04/68 - 07/11/68
		D19229	5	07/11/68 - 07/17/68
		D19230	6	07/18/68 - 07/25/68
		D19231	7	07/25/68 - 08/01/68
		D19232	8	08/01/68 - 08/08/68
		D19233	9	08/08/68 - 08/15/68
DR01022	DS01022	D19234	1	08/15/68 - 08/22/68
		D19235	2	08/22/68 - 08/29/68
		D19236	3	08/29/68 - 09/05/68
		D19237	4	09/05/68 - 09/12/68
		D18589	5	09/12/68 - 09/19/68
		D18590	6	09/19/68 - 09/26/68
		D18591	7	09/26/68 - 10/03/68
		D18592	8	10/03/68 - 10/06/68
		D18593	9	10/06/68 - 10/07/68

67-070A-000

DR#	DS#	D#	FILES	TIME SPAN
DR01023	DS01023	D18594	1	10/07/68 - 10/10/68
		D18595	2	10/10/68 - 10/17/68
		D18596	3	10/17/68 - 10/24/68
		D18597	4	10/24/68 - 10/31/68
		D18598	5	10/31/68 - 11/07/68
		D18599	6	11/07/68 - 11/14/68
		D18570	7	11/14/68 - 11/21/68
		D19010	8	11/21/68 - 11/28/68
		D18571	9	11/28/68 - 12/05/68
DR01024	DS01024	D18572	1	12/05/68 - 12/12/68
		D18573	2	12/12/68 - 12/19/68
		D18574	3	12/26/68 - 12/31/68
		D18575	4	01/01/69 - 01/01/69
		D18569	5	01/02/69 - 01/09/69
		D19238	6	01/09/69 - 01/16/69
		D19011	7	01/16/69 - 01/23/69
		D18576	8	01/23/69 - 01/30/69
		D18577	9	01/30/69 - 02/06/69
		D19541	10	02/06/69 - 02/13/69
DR01025	DS01025	D19012	1	02/13/69 - 02/20/69
		D18578	2	02/20/69 - 02/27/69
		D18600	3	02/27/69 - 03/03/69
		D18601	4	03/03/69 - 03/06/69
		D18602	5	03/06/69 - 03/13/69
		D18603	6	03/13/69 - 03/20/69
		D18604	7	03/20/69 - 03/27/69
		D18605	8	03/27/69 - 04/03/69
		D18606	9	04/03/69 - 04/09/69
		D18607	10	04/10/69 - 04/17/69
DR01026	DS01026	D18608	1	04/17/69 - 04/24/69
		D19239	2	04/24/69 - 05/01/69
		D18609	3	05/01/69 - 05/08/69
		D18619	4	05/08/69 - 05/15/69
		D18618	5	05/15/69 - 05/22/69
		D18617	6	05/22/69 - 05/29/69
		D18616	7	05/29/69 - 06/05/69
		D19013	8	06/05/69 - 06/11/69
		D18615	9	06/12/69 - 06/15/69
		D18614	10	06/15/69 - 06/19/69
DR01027	DS01027	D19014	1	06/19/69 - 06/26/69
		D18613	2	06/26/69 - 07/03/69
		D18612	3	07/03/69 - 07/10/69
		D19542	4	07/17/69 - 07/24/69
		D18611	5	07/24/69 - 07/31/69
		D18610	6	07/31/69 - 08/07/69
		D18588	7	08/07/69 - 08/10/69
		D18587	8	08/10/69 - 08/14/69
		D18586	9	08/14/69 - 08/21/69
		D18585	10	08/21/69 - 08/28/69

67-070A-CED

DR#	DS#	D#	FILES	TIME SPAN
DR01028	DS01028	D18584	1	08/28/69 - 09/04/69
		D18583	2	09/04/69 - 09/11/69
		D18582	3	09/11/69 - 09/17/69
		D18581	4	09/18/69 - 09/25/69
		D18580	5	09/25/69 - 10/01/69
		D18579	6	10/01/69 - 10/08/69
		D18984	7	10/08/69 - 10/09/69
		D18985	8	10/09/69 - 10/16/69
		D18986	9	10/16/69 - 10/23/69
		D18987	10	10/23/69 - 10/30/69
DR01029	DS01029	D18988	1	10/30/69 - 11/05/69
		D18989	2	11/06/69 - 11/13/69
		D18990	3	11/13/69 - 11/20/69
		D18991	4	11/20/69 - 11/26/69
		D18992	5	11/27/69 - 12/01/69
		D18993	6	11/30/69 - 12/01/69
		D18994	7	12/01/69 - 12/04/69
		D19240	8	12/04/69 - 12/11/69
		D19543	9	12/11/69 - 12/18/69
		D19025	10	12/18/69 - 12/25/69
DR01030	DS01030	D19026	1	12/25/69 - 01/01/70
		D19027	2	01/01/70 - 01/08/70
		D19028	3	01/08/70 - 01/15/70
		D19029	4	01/15/70 - 01/22/70
		D19030	5	01/22/70 - 01/29/70
		D19031	6	01/28/70 - 01/29/70
		D19032	7	01/29/70 - 02/05/70
		D19034	8	02/05/70 - 02/06/70
		D19033	9	02/06/70 - 02/11/70
		D19024	10	02/11/70 - 02/19/70
DR01031	DS01031	D19023	1	02/19/70 - 02/25/70
		D19022	2	02/26/70 - 03/05/70
		D19021	3	03/05/70 - 03/11/70
		D19020	4	03/12/70 - 03/19/70
		D19019	5	03/19/70 - 03/26/70
		D19018	6	03/26/70 - 04/02/70
		D19017	7	04/02/70 - 04/09/70
		D19035	8	04/09/70 - 04/16/70
		D19036	9	04/16/70 - 04/23/70
		D19015	10	04/23/70 - 04/30/70
DR01032	DS01032	D19016	1	04/30/70 - 05/07/70
		D19037	2	05/07/70 - 05/14/70
		D19038	3	05/14/70 - 05/21/70
		D19039	4	05/21/70 - 05/28/70
		D19040	5	05/28/70 - 06/04/70
		D19041	6	06/04/70 - 06/11/70
		D19042	7	06/11/70 - 06/18/70
		D19043	8	06/18/70 - 06/25/70
		D19044	9	06/25/70 - 07/01/70
		D19045	10	07/02/70 - 07/09/70

67-070A-033

DR#	DS#	D#	FILES	TIME SPAN
DR01033	DS01033	D19046	1	07/09/70 - 07/15/70
		D19047	2	07/16/70 - 07/23/70
		D19048	3	07/23/70 - 07/29/70
		D19049	4	07/29/70 - 08/06/70
		D19050	5	08/06/70 - 08/13/70
		D19051	6	08/13/70 - 08/20/70
		D19052	7	08/20/70 - 08/27/70
		D19053	8	08/27/70 - 09/03/70
		D19054	9	09/03/70 - 09/10/70
		D18995	10	09/10/70 - 09/17/70
DR01034	DS01034	D18996	1	09/17/70 - 09/24/70
		D18997	2	09/24/70 - 10/01/70
		D18998	3	10/01/70 - 10/08/70
		D18999	4	10/08/70 - 10/15/70
		D19000	5	10/15/70 - 10/22/70
		D19001	6	10/22/70 - 10/29/70
		D19002	7	10/29/70 - 11/05/70
		D19003	8	11/05/70 - 11/12/70
		D19004	9	11/12/70 - 11/19/70
		D19005	10	11/19/70 - 11/26/70
DR01035	DS01035	D19006	1	11/26/70 - 12/03/70
		D19007	2	12/03/70 - 12/10/70
		D19008	3	12/10/70 - 12/17/70
		D19009	4	12/17/70 - 12/24/70
		D17796	5	07/29/71 - 08/05/71
		D17806	6	08/05/71 - 08/11/71
		D17810	7	08/12/71 - 08/19/71
		D17809	8	08/19/71 - 08/25/71
		D17805	9	08/26/71 - 08/28/71
		D17790	10	08/29/71 - 09/02/71
DR01036	DS01036	D17804	1	09/02/71 - 09/08/71
		D19241	2	09/09/71 - 09/15/71
		D17791	3	09/16/71 - 09/23/71
		D17803	4	09/23/71 - 09/29/71
		D17802	5	09/30/71 - 10/07/71
		D17811	6	10/07/71 - 10/14/71
		D17813	7	10/14/71 - 10/21/71
		D17814	8	10/21/71 - 10/28/71
		D19242	9	10/28/71 - 11/04/71
		D19243	10	11/04/71 - 11/11/71
DR01037	DS01037	D19244	1	11/11/71 - 11/17/71
		D19245	2	11/18/71 - 11/26/71
		D19246	3	11/25/71 - 12/01/71
		D19247	4	12/02/71 - 12/09/71
		D19248	5	12/09/71 - 12/16/71
		D19249	6	12/16/71 - 12/23/71
		D19250	7	12/23/71 - 12/30/71

67-070A-03F

DR#	DS#	ID#	FILES	TIME SPAN
DS03007	DS03007	D26560	1	07/19/67 - 09/27/67
		D26561	2	09/28/67 - 12/07/67
		D26562	3	12/07/67 - 02/15/68
		D26563	4	02/15/68 - 04/25/68
		D26564	5	04/25/68 - 07/04/68
		D26565	6	07/04/68 - 09/12/68
		D26566	7	09/12/68 - 11/21/68
		D26567	8	11/21/68 - 01/30/69
		D26568	9	01/30/69 - 04/09/69
		D26569	10	04/10/69 - 06/19/69
		D26570	11	06/19/69 - 08/28/69
		D26571	12	08/28/69 - 11/05/69
		D26572	13	11/06/69 - 01/15/70
		D26573	14	01/15/70 - 03/26/70
		D26574	15	03/26/70 - 06/04/70
		D26575	16	06/04/70 - 08/13/70
		D26576	17	08/13/70 - 10/22/70
		D26577	18	10/22/70 - 12/31/70
		D26578	19	07/29/71 - 10/28/71
		D26579	20	10/28/71 - 12/30/71

REQ. AGENT
CMT
VJP

RAND NO.
RC2827
RC3358

ACQ. AGENT
JHK

IMP-E

82 SEC. AVGD VECTORS

5.12 SEC. MAG. VECTORS

67-070A-03C

67-070A-03D

This data set catalog contains 20 82 Second Averaged Vectors and 210 5.12 Second Magnetic Field Vectors data tapes. The tapes are 556 BPI, Binary, 7 track. The 82 Sec. data are multifiled and the 5.12 Sec. data are 1 file each. These tapes were written on an IBM 7094 computer. In determining the start and stop times consider Jan. 1, = day 0.

Time spans are as follows:

82 Second Averaged Vectors

<u>D#</u>	<u>C#</u>	<u>FILES</u>	<u>TIME SPAN</u>
D-17537	C-14120	10	7/19/67 - 9/27/67
D-17540	C-14121	10	9/28/67 - 12/07/67
D-17532	C-14115	10	12/06/67 - 2/14/68
D-17533	C-14116	11	2/14/68 - 4/25/68
D-17525	C-14108	11	4/25/68 - 7/04/68
D-17534	C-14117	10	7/04/68 - 9/12/68
D-17535	C-14118	12	9/12/68 - 11/21/68
D-17538	C-15123	11	11/21/68 - 1/30/69
D-17536	C-14119	11	1/30/69 - 4/09/69
D-17539	C-15124	11	4/10/69 - 6/19/69
D-17531	C-14114	11	6/19/69 - 8/28/69
D-17541	C-14122	11	8/28/69 - 11/05/69
D-17530	C-14113	12	11/06/69 - 1/15/70
D-17529	C-14112	12	1/15/70 - 3/26/70

<u>D#</u>	<u>C#</u>	<u>FILES</u>	<u>TIME SPAN</u>
D-17528	C-14111	10	3/26/70 - 6/04/70
D-17542	C-14123	10	6/04/70 - 8/13/70
D-17527	C-14110	10	8/13/70 - 10/22/70
D-17526	C-14109	9	10/22/70 - 12/24/70
D-17543	C-14124	14	7/29/71 - 10/28/71
D-18983	C-14134	9	10/28/71 - 12/30/71

5.12 Second Magnetic Field Vectors

<u>D#</u>	<u>C#</u>	<u>WEEK</u>	<u>TIME SPAN</u>
D-17774	C-14825	01	7/19/67 - 7/27/67
D-17782	C-14833	02	7/27/67 - 8/03/67
D-17783	C-14834	03	8/03/67 - 8/10/67
D-17772	C-14823	04	8/10/67 - 8/17/67
D-17784	C-14835	05	8/17/67 - 8/24/67
D-17785	C-14836	06	8/24/67 - 8/30/67
D-17786	C-14837	07	8/31/67 - 9/07/67
D-17787	C-14838	08	9/07/67 - 9/14/67
D-17819	C-14870	09	9/14/67 - 9/21/67
D-17788	C-14839	10	9/21/67 - 9/27/67
D-17773	C-14824	11	9/28/67 - 10/04/67
D-17789	C-14840	12	10/05/67 - 10/12/67
D-17777	C-14828	13	10/12/67 - 10/19/67
D-17776	C-14827	14	10/19/67 - 10/25/67
D-17815	C-14866	15	10/26/67 - 11/02/67
D-17775	C-14826	16	11/02/67 - 11/09/67
D-17778	C-14829	17	11/09/67 - 11/15/67
D-17779	C-14830	18	11/16/67 - 11/23/67
D-17780	C-14831	19	11/23/67 - 11/29/67

<u>D#</u>	<u>C#</u>	<u>WEEK</u>	<u>TIME SPAN</u>
D-17781	C-14832	20	11/30/67 - 12/07/67
D-17818	C-14869	21	12/07/67 - 12/14/67
D-17817	C-14868	22	12/14/67 - 12/20/67
D-17816	C-14867	23	12/21/67 - 12/28/67
D-17795	C-14846	25	1/04/68 - 1/11/68
D-17794	C-14845	26	1/11/68 - 1/17/68
D-17792	C-14843	27	1/18/68 - 1/25/68
D-17793	C-14844	28	1/25/68 - 1/31/68
D-17823	C-14874	29	2/01/68 - 2/08/68
D-17822	C-14873	30	2/08/68 - 2/15/68
D-17821	C-14872	31	2/15/68 - 2/22/68
D-17801	C-14852	33	2/29/68 - 3/07/68
D-17812	C-14863	34	3/07/68 - 3/13/68
D-17808	C-14859	35	3/14/68 - 3/21/68
D-17800	C-14851	36	3/21/68 - 3/28/68
D-17799	C-14850	37	3/28/68 - 4/04/68
D-17807	C-14858	38	4/04/68 - 4/11/68
D-17798	C-14849	39A	4/11/68 - 4/13/68
D-17797	C-14848	39B	4/13/68 - 4/18/68
D-17820	C-14871	40	4/18/68 - 4/25/68
D-19218	C-15619	41	4/25/68 - 5/02/68
D-19219	C-15620	42	5/02/68 - 5/09/68
D-19220	C-15621	43	5/09/68 - 5/16/68
D-19221	C-15622	44	5/16/68 - 5/23/68
D-19222	C-15623	45	5/23/68 - 5/30/68
D-19223	C-15624	46	5/30/68 - 6/06/68
D-19224	C-15625	47	6/06/68 - 6/13/68
D-19225	C-15626	48	6/13/68 - 6/20/68
D-19226	C-15627	49	6/20/68 - 6/27/68

<u>D#</u>	<u>C#</u>	<u>WEEK</u>	<u>TIME SPAN</u>
D-19227	C-15628	50	6/27/68 - 7/04/68
D-19228	C-15629	51	7/04/68 - 7/11/68
D-19229	C-15630	52	7/11/68 - 7/17/68
D-19230	C-15631	53	7/18/68 - 7/25/68
D-19231	C-15632	54	7/25/68 - 8/01/68
D-19232	C-15633	55	8/01/68 - 8/08/68
D-19233	C-15634	56	8/08/68 - 8/15/68
D-19234	C-15635	57	8/15/68 - 8/22/68
D-19235	C-15636	58	8/22/68 - 8/29/68
D-19236	C-15637	59	8/29/68 - 9/05/68
D-19237	C-15638	60	9/05/68 - 9/12/68
D-18589	C-14950	61	9/12/68 - 9/19/68
D-18590	C-14951	62	9/19/68 - 9/26/68
D-18591	C-14952	63	9/26/68 - 10/03/68
D-18592	C-14953	64A	10/03/68 - 10/06/68
D-18593	C-14954	64B	10/06/68 - 10/07/68
D-18594	C-14955	64C	10/07/68 - 10/10/68
D-18595	C-14956	65	10/10/68 - 10/17/68
D-18596	C-14957	66	10/17/68 - 10/24/68
D-18597	C-14958	67	10/24/68 - 10/31/68
D-18598	C-14959	68	10/31/68 - 11/07/68
D-18599	C-14960	69	11/07/68 - 11/14/68
D-18570	C-14931	70	11/14/68 - 11/21/68
D-18571	C-15321	71	11/21/68 - 11/28/68
D-18572	C-14932	72	11/28/68 - 12/05/68
D-18573	C-14933	73	12/05/68 - 12/12/68
D-18574	C-14934	74	12/12/68 - 12/19/68
D-18575	C-14935	76	12/26/68 - 12/31/68

<u>D#</u>	<u>C#</u>	<u>WEEK</u>	<u>TIME SPAN</u>
D-18575	C-14936	76	1/01/69 - 1/01/69
D-18569	C-14930	77	1/02/69 - 1/09/69
D-19238	C- 19239 15639	78	1/09/69 - 1/16/69
D-19011	C-15322	79	1/16/69 - 1/23/69
D-18576	C-14937	80	1/23/69 - 1/30/69
D-18577	C-14938	81	1/30/69 - 2/06/69
D-19541	C-15813	82	2/06/69 - 2/13/69
D-19012	C-15323	83	2/13/69 - 2/20/69
D-18578	C-14939	84	2/20/69 - 2/27/69
D-18600	C-14961	85	2/27/69 - 3/03/69
D-18601	C-14962	85	3/03/69 - 3/06/69
D-18602	C-14963	86	3/06/69 - 3/13/69
D-18603	C-14964	87	3/13/69 - 3/20/69
D-18604	C-14965	88	3/20/69 - 3/27/69
D-18605	C-14966	89	3/27/69 - 4/03/69
D-18606	C-14967	90	4/03/69 - 4/09/69
D-18607	C-14968	91	4/10/69 - 4/17/69
D-18608	C-14969	92	4/17/69 - 4/24/69
D-19239	C- 19240 15640	93	4/24/69 - 5/01/69
D-18609	C-14970	94	5/01/69 - 5/08/69
D-18619	C-14980	95	5/08/69 - 5/15/69
D-18618	C-14979	96	5/15/69 - 5/22/69
D-18617	C-14978	97	5/22/69 - 5/29/69
D-18616	C-14977	98	5/29/69 - 6/05/69
D-19013	C-15324	99	6/05/69 - 6/11/69
D-18615	C-14976	100	6/12/69 - 6/15/69
D-18614	C-14975	100	6/15/69 - 6/19/69
D-19014	C-15325	101	6/19/69 - 6/26/69

<u>D#</u>	<u>C#</u>	<u>WEEK</u>	<u>TIME SPAN</u>
D-18613	C-14974	102	6/26/69 - 7/03/69
D-18612	C-14973	103	7/03/69 - 7/10/69
D-19542	C-15814	105	7/17/69 - 7/24/69
D-18611	C-14972	106	7/24/69 - 7/31/69
D-18610	C-14971	107	7/31/69 - 8/07/69
D-18588	C-14949	108	8/10/69 - 8/10/69
D-18587	C-14948	108	8/10/69 - 8/14/69
D-18586	C-14947	109	8/14/69 - 8/21/69
D-18585	C-14946	110	8/21/69 - 8/28/69
D-18584	C-14945	111	8/28/69 - 9/04/69
D-18583	C-14944	112	9/04/69 - 9/11/69
D-18582	C-14943	113	9/11/69 - 9/17/69
D-18581	C-14942	114	9/18/69 - 9/25/69
D-18580	C-14941	115	9/25/69 - 10/01/69
D-18579	C-14940	116	10/01/69 - 10/08/69
D-18984	C-15136	116	10/08/69 - 10/09/69
D-18985	C-15137	117	10/09/69 - 10/16/69
D-18986	C-15138	118	10/16/69 - 10/23/69
D-18987	C-15139	119	10/23/69 - 10/30/69
D-18988	C-15140	120	10/30/69 - 11/05/69
D-18989	C-15141	121	11/06/69 - 11/13/69
D-18990	C-15142	121	11/13/69 - 11/20/69
D-18991	C-15143	123	11/20/69 - 11/26/69
D-18992	C-15144	124	11/27/69 - 12/01/69
D-18993	C-15145	124	11/30/69 - 12/01/69
D-18994	C-15305	124	12/01/69 - 12/04/69
D-19240	C-15641	125	12/04/69 - 12/11/69
D-19543	C-15815	126	12/11/69 - 12/18/69

<u>D#</u>	<u>C#</u>	<u>WEEK</u>	<u>TIME SPAN</u>
D-19025	C-15409	127	12/18/69 - 12/25/69
D-19026	C-15410	128	12/25/69 - 1/01/70
D-19027	C-15411	129	1/01/70 - 1/08/70
D-19028	C-15412	130	1/08/70 - 1/15/70
D-19029	C-15413	131	1/15/70 - 1/22/70
D-19030	C-15414	132A	1/22/70 - 1/29/70
D-19031	C_15415	132B	1/28/70 - 1/29/70
D-19032	C-15416	133	1/29/70 - 2/05/70
D-19034	C-15418	134B	2/05/70 - 2/06/70
D-19033	C-15417	134E	2/06/70 - 2/11/70
D-19024	C-15408	135	2/11/70 - 2/19/70
D-19023	C-15334	136	2/19/70 - 2/25/70
D-19022	C-15333	137	2/26/70 - 3/05/70
D-19021	C-15332	138	3/05/70 - 3/11/70
D-19020	C-15331	139	3/12/70 - 3/19/70
D-19019	C-15330	140	3/19/70 - 3/26/70
D-19018	C-15329	141	3/26/70 - 4/02/70
D-19017	C-15328	142	4/02/70 - 4/09/70
D-19035	C-15419	143	4/09/70 - 4/16/70
D-19036	C-15420	144	4/16/70 - 4/23/70
D-19015	C-15326	145	4/23/70 - 4/30/70
D-19016	C-15327	146	4/30/70 - 5/07/70
D-19037	C-15421	147	5/07/70 - 5/14/70
D-19038	C-15422	148	5/14/70 - 5/21/70
D-19039	C-15423	149	5/21/70 - 5/28/70
D-19040	C-15424	150	5/28/70 - 6/04/70
D-19041	C-15425	151	6/04/70 - 6/11/70
D-19042	C-15426	152	6/11/70 - 6/18/70
D-19043	C-15427	153	6/18/70 - 6/25/70

<u>D#</u>	<u>C#</u>	<u>WEEK</u>	<u>TIME SPAN</u>
D-19044	C-15428	154	6/25/70 - 7/01/70
D-19045	C-15429	155	7/01/70 - 7/09/70
D-19046	C-15430	156	7/09/70 - 7/15/70
D-19047	C-15431	157	7/16/70 - 7/23/70
D-19048	C-15432	158	7/23/70 - 7/29/70
D-19049	C-15433	159	7/29/70 - 8/06/70
D-19050	C-15434	160	8/06/70 - 8/13/70
D-19051	C-15435	161	8/13/70 - 8/20/70
D-19052	C-15436	162	8/20/70 - 8/27/70
D-19053	C-15437	163	8/27/70 - 9/03/70
D-19054	C-15438	164	9/03/70 - 9/10/70
D-18995	C-15306	165	9/10/70 - 9/17/70
D-18996	C-15307	166	9/17/70 - 9/24/70
D-18997	C-15308	167	9/24/70 - 10/01/70
D-18998	C-15309	168	10/01/70 - 10/08/70
D-18999	C-15310	169	10/08/70 - 10/15/70
D-19000	C-15311	170	10/15/70 - 10/22/70
D-19001	C-15312	171	10/22/70 - 10/29/70
D-19002	C-15313	172	10/29/70 - 11/05/70
D-19003	C-15314	173	11/05/70 - 11/12/70
D-19004	C-15315	174	11/12/70 - 11/19/70
D-19005	C-15316	175	11/19/70 - 11/26/70
D-19006	C-15317	176	11/26/70 - 12/03/70
D-19007	C-15318	177	12/03/70 - 12/10/70
D-19008	C-15319	178	12/10/70 - 12/17/70
D-19009	C-15320	179	12/17/70 - 12/24/70
D-17796	C-14847	211	7/29/71 - 8/05/71
D-17806	C-14857	212	8/05/71 - 8/11/71

<u>D#</u>	<u>C#</u>	<u>WEEK</u>	<u>TIME SPAN</u>
D-17810	C-14861	213	8/12/71 - 8/19/71
D-17809	C-14860	214	8/19/71 - 8/25/71
D-17805	C-14856	215	8/26/71 - 8/28/71
D-17790	C-14841	215	8/29/71 - 9/02/71
D-17804	C-14855	216	9/02/71 - 9/08/71
D-19241	C-15642	217	9/09/71 - 9/15/71
D-17791	C-14842	218	9/16/71 - 9/23/71
D-17803	C-14854	219	9/23/71 - 9/29/71
D-17802	C-14853	220	9/30/71 - 10/07/71
D-17811	C-14862	221	10/07/71 - 10/14/71
D-17813	C-14864	222	10/14/71 - 10/21/71
D-17814	C-14865	223	10/21/71 - 10/28/71
D-19242	C-15643	224	10/28/71 - 10/04/71
D-19243	C-15644	225	11/04/71 - 11/11/71
D-19244	C-15645	226	11/11/71 - 11/17/71
D-19245	C-15646	227	11/18/71 - 11/26/71
D-19246	C-15647	228	11/25/71 - 12/01/71
D-19247	C-15648	229	12/02/71 - 12/09/71
D-19248	C-15649	230	12/09/71 - 12/16/71
D-19249	C-15650	231	12/16/71 - 12/23/71
D-19250	C-15651	232	12/23/71 - 12/30/71

Project 1064-3 - Technical Note 122

Originator: Sharone Lathrop

Date: March 8, 1972

Subject: Documentation of a FORTRAN Package to Read DCS
Blocked Explorer Individual Vector Tapes

TABLE OF CONTENTS

		<u>Page</u>
Section	1 PROGRAM DESCRIPTION	1-1
	1.1 Purpose	1-1
	1.2 Input/Output	1-1
	1.3 Method	
Section	2 SUBROUTINE DESCRIPTION	2-1
	2.1 Subroutine IVREAD	2-1
	2.1.1 Purpose	2-1
	2.1.2 Calling Sequence	2-1
	2.1.3 Key Variables	2-2
	2.1.4 Method	2-3
	2.1.5 Comments	2-3
	2.2 Subroutine CNVTCP	2-8
	2.2.1 Purpose	2-8
	2.2.2 Calling Sequence	
	2.2.3 Key Variables	2-8
	2.2.4 Method	2-10
	2.2.5 Comments	2-10
	2.3 Function ALPHA	2-12
	2.3.1 Purpose	2-12
	2.3.2 Calling Sequence	2-12
	2.3.3 Key Variables	2-12
	2.3.4 Method	2-12
	2.3.5 Comments	2-12
Section	3, USAGE INSTRUCTIONS	3-1
	3.1 Description	3-1

3.2	System and Equipment	3-1
3.3	Control C	3-1
3.4	Calling the Routine	3-1
3.5	Restrictions	3-3
Appendix A	DESCRIPTION OF THE EXPLORER INDIVIDUAL VECTOR TAPE	A-1
Appendix B	AMES, IBM/360 RE-READ CAPABILITY	B-1
Appendix C	PROGRAM LISTING	C-1

Section 1

PROGRAM DESCRIPTION

1.1 PURPOSE

This set of three FORTRAN subprograms is used to read the Explorer 33 and Explorer 35 Magnetometer Individual Vector tapes on the IBM/360. The tapes are blocked for the IBM 7040/7094 DCS which makes them directly unreadable by other systems.

1.2 INPUT/OUTPUT

Input is an Explorer Magnetometer Individual Vector tape. A complete tape description is included as Appendix A. Output is one logical record, that is one individual vector for each call to the subroutine IVREAD. The exact amount of data returned depends on whether the vector is the first in a sequence as described under method.

1.3 METHOD

The program is written in FORTRAN IV for use on the IBM 360/67. The program makes use of a re-read capability which may not be available on other machines. The Ames write-up for this feature is included as Appendix B. Essentially, an array is defined to be also a buffer area. Each time a physical record is read it is placed in this array, starting at any desired place in the array. The array is then treated as an external device which may be read many times. The procedure is to read one physical record into the array in A format, then locate and read successive logical records in the appropriate integer and floating point formats.

The re-read approach is used because there are two vector formats. The first vector of each sequence contains time information, data and a count of the number of vectors in the sequence. The rest of the vectors in a sequence have blanks instead of time information and are six characters shorter because they do not have the vector count. The number of vectors in a sequence ranges from eight to fourteen. Each physical record contains a fixed number of characters, 2760, including control words. There are about twenty-one vectors then in a physical record.

Section 2

SUBROUTINE DESCRIPTION

2.1 SUBROUTINE IVREAD

2.1.1 Purpose

This subroutine reads standard Explorer Individual Vector tapes and returns one vector for each subroutine call.

2.1.2 Calling Sequence

CALL IVREAD (ITIME, L, VECTOR, NVEC)

All four quantities are output; NVEC is also input and should not be changed by the calling program.

ITIME is an integer array of dimension five containing sequence number, day, hour, minute, second.

L is vector position.

VECTOR is a real array containing the magnitude, angles and vector components in three coordinate systems. The dimension is sixteen.

The dimension is sixteen.

NVEC is the number of vectors in the sequence.

L and VECTOR are set on every call, but ITIME and NVEC are changed only when the first vector of a sequence is read.

2.1.3 Key Variables

<u>Variable</u>	<u>Description</u>
A	The array into which the data is read, and which serves as a buffer to be re-read, 721 words
BLANK	An integer containing four blank characters for resetting formats.
FIRST	A logical flag set to true before the first entry to IVREAD and false after
FMT	The format for reading all vectors except the first in a sequence (integer).
FMT1	The format for reading the first vector in a sequence (integer)
FMT3, FMT4	The integer words for initializing the third and fourth words of FMT and FMT1.
INDEX	The character position inside the word array A where the previous vector ended. The position of the last character of the six character control word
IREAD	A logical word set to true if a physical record will be read on the next subroutine call, i. e. if there is less than a complete vector left in the buffer
IVEC	A counter of the number of vectors read in this sequence
J	A counter of the number of words moved from the end to the start of the buffer
N1	The number of multiples of 255 characters to be skipped with format FMT or FMT1
N2	The number of remaining characters to skip, $0 \leq N2 \leq 255$.

2.1.4 Method

We start by declaring that A is a buffer area of 2884 characters. This size is one physical record, 2748 characters, (the twelve character control word is skipped) plus the longest vector, 132 characters, plus four characters.

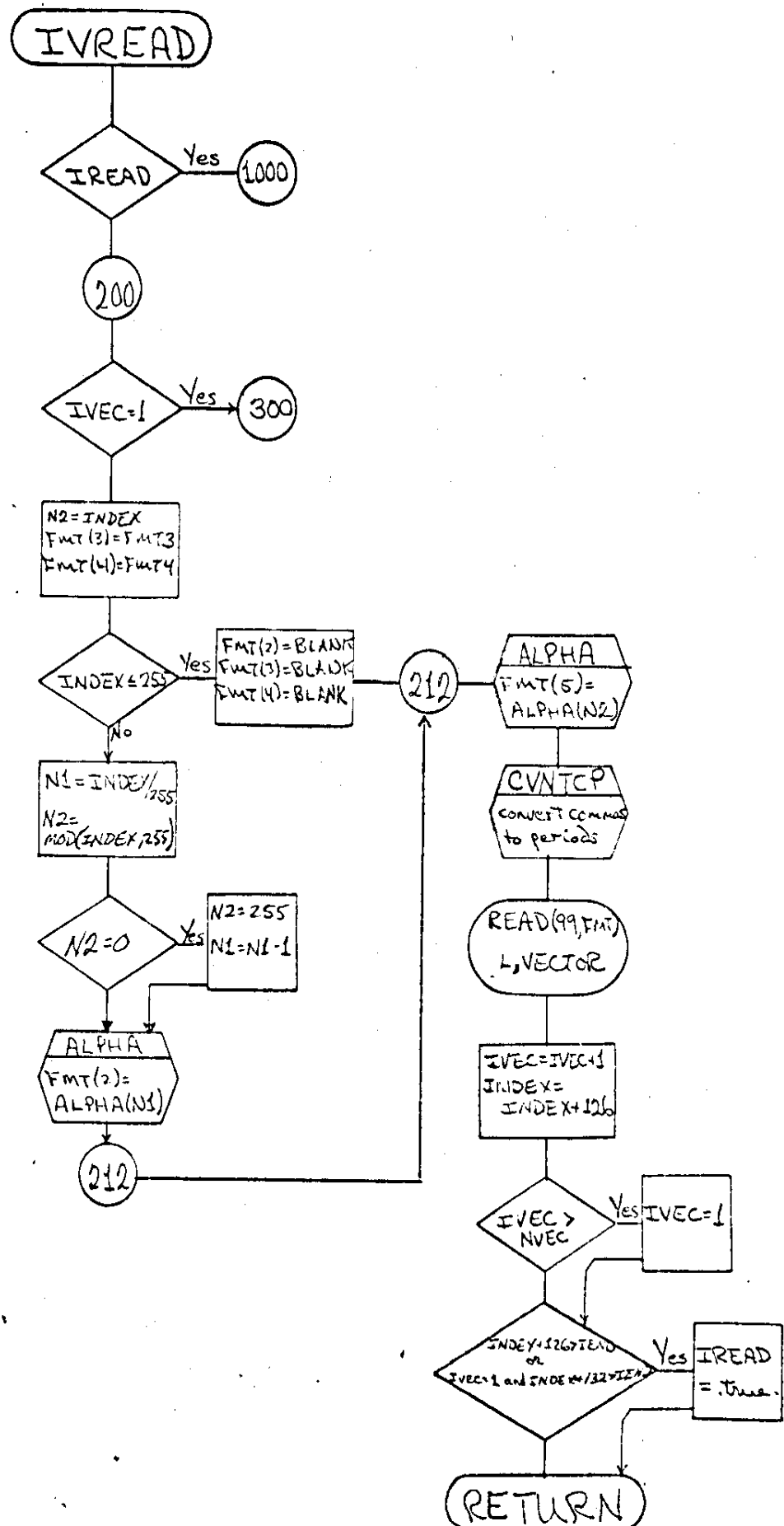
Next, one physical record is read in A format skipping the twelve character control word. On the first read the data is placed at the beginning of the buffer. Before subsequent reads any partial vector at the end of the buffer is moved to the beginning of the buffer. This transfer is on a word basis. To avoid trying to move zero words an extra four characters (one word) are moved.

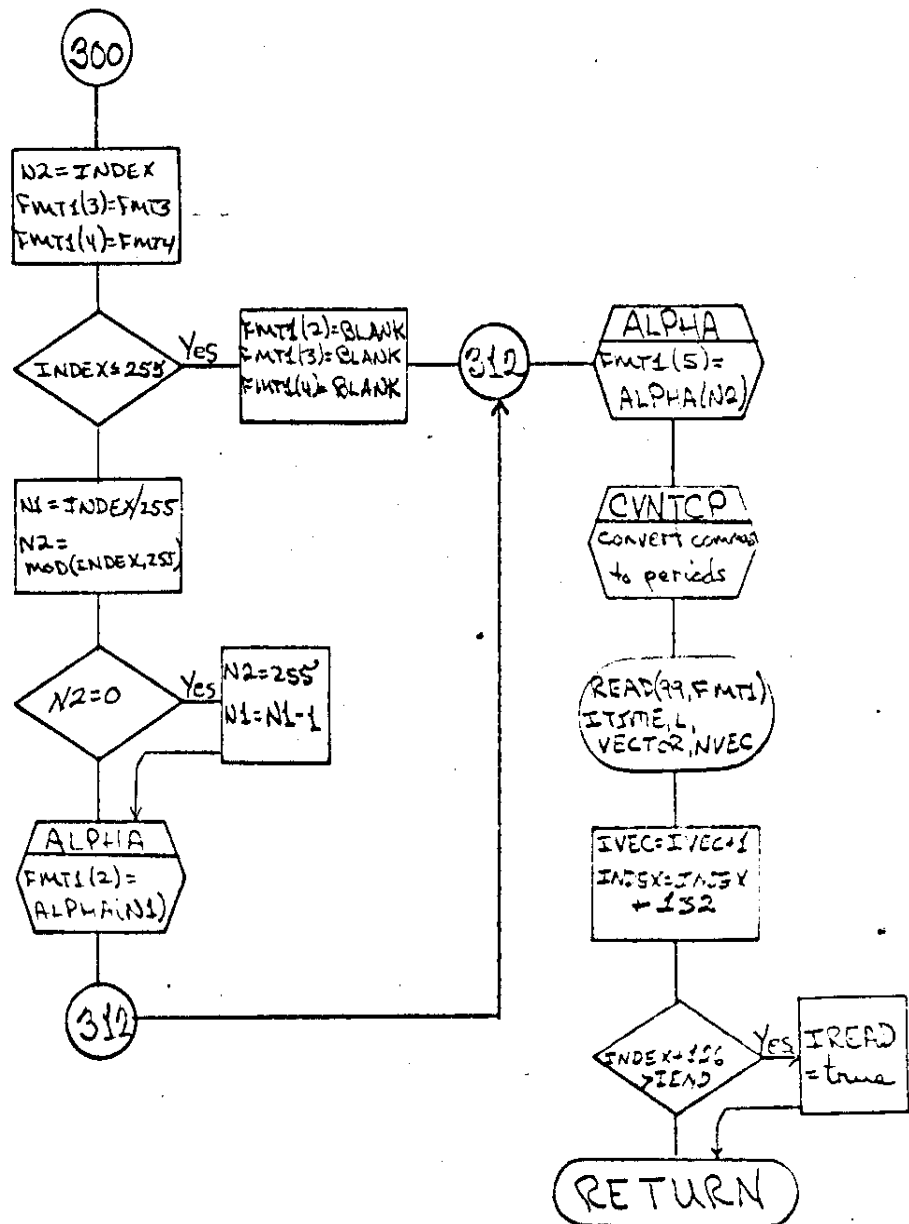
A vector is re-read from the buffer area using an execution time format. The format is referenced to the beginning of the buffer area, so a variable number of characters must be skipped. On the very first read the control word of six characters must be skipped. The second read skips 138 characters, the first vector and another control word. Soon more than 255 characters must be skipped. However Fortran limits you to 255X. Thus we code 2558x as 10(255X), 8X. The ten and eight are N1 and N2 listed under Key Variables. Likewise, 2550x would be coded 9(255x), 255x to avoid the error of reading 0x. The execution time formats are setup by determining N1 and N2 then using function ALPHA to obtain their character representation in the appropriate place in the format.

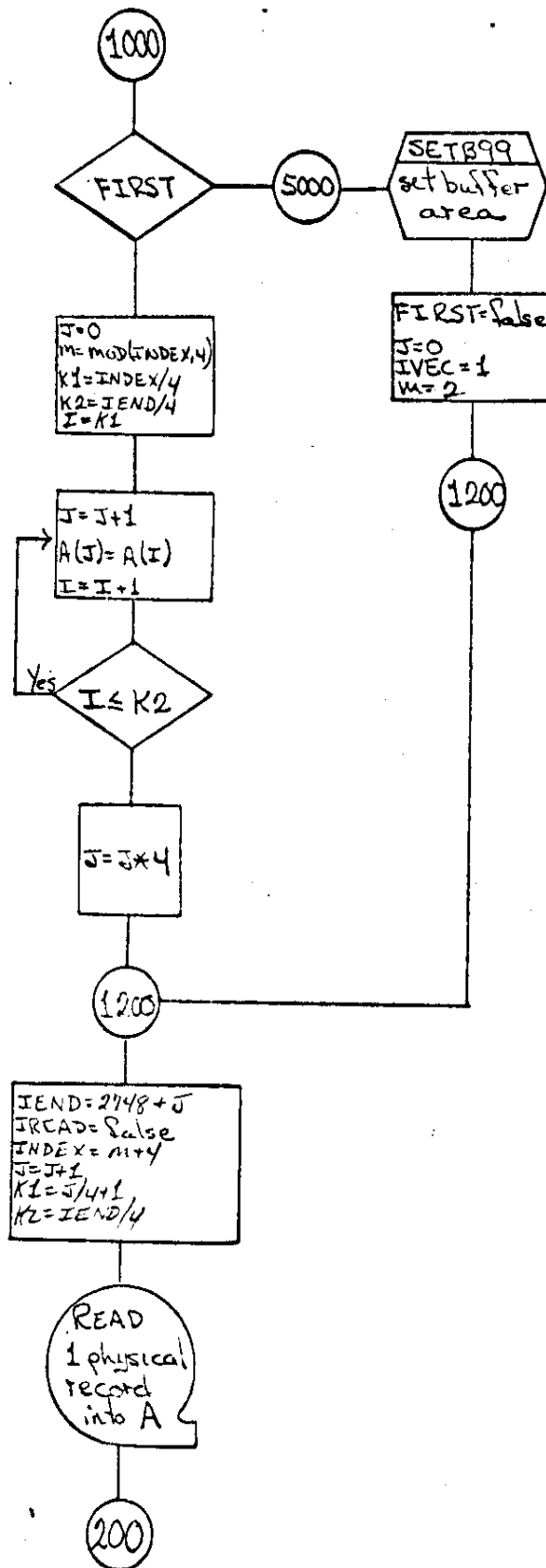
2.1.5 Comments

There is extensive use of the number four in this routine. There are four characters per word on the IBM/360. A machine such as a CDC 6600 which has ten characters per word would need to change these

fours to tens. Also, a CDC 6600 would probably not have the Read (99) feature, but the CDC DECODE routine is similar.







2.2 SUBROUTINE CNVTCP

2.2.1 Purpose

When the IBM 7040/7094 DCS writes a blocked tape it does not translate characters from internal Storage format to tape format (Table 2-1). Numbers have the same code, except for zero which seems to read satisfactorily. Blanks show up as ampersands, which read correctly, but periods show up as commas. This subroutine converts the commas to periods.

2.2.2 Calling Sequence

CALL CNVTCP (A, INDEX)

where A is the array containing the data, INDEX is the byte of A where the last vector ended.

2.2.3 Key Variables

CHAR	Array of masks to select one character from a word
COMMA	Array of masks to check for comma in any of the four character positions
I	Index of word in A being checked or changed
J	Position of character in word of A being checked or changed
K1	Index of first word of A to be checked
PERIOD	Array of masks to change comma to period in any of four character positions

APPENDIX B — TABLE OF SOURCE PROGRAM CHARACTERS

CHARACTER	CARD	BCD TAPE	STORAGE	CHARACTER	CARD	BCD TAPE	STORAGE	CHARACTER	CARD	BCD TAPE	STORAGE	CHARACTER	CARD	BCD TAPE	STORAGE
1	1	01	01	A	12 1	61	21	J	11 1	41	41	/	0 1	21	61
2	2	02	02	B	12 2	62	22	K	11 2	42	42	S	0 2	22	62
3	3	03	03	C	12 3	63	23	L	11 3	43	43	T	0 3	23	63
4	4	04	04	D	12 4	64	24	M	11 4	44	44	U	0 4	24	64
5	5	05	05	E	12 5	65	25	N	11 5	45	45	V	0 5	25	65
6	6	06	06	F	12 6	66	26	O	11 6	46	46	W	0 6	26	66
7	7	07	07	G	12 7	67	27	P	11 7	47	47	X	0 7	27	67
8	8	10	10	H	12 8	70	30	Q	11 8	50	50	Y	0 8	30	70
9	9	11	11	I	12 9	71	31	R	11 9	51	51	Z	0 9	31	71
blank	blank	20	60	+	12 60	20	-	11 40	40	40	- 0	0 0	12	00	-
=	8-3	13	13	.	12 8-3	73	33	\$	11 8-3	53	53	,	0 8-3	33	73
-	8-4	14	14	)	12 8-4	74	34	*	11 8-4	54	54	(	0 8-4	34	74

NOTE: There are two - signs. Only the 11-punch minus sign can be used in FORTRAN source program cards. Either minus sign may be used in input data to the object program; object program output uses the 11-punch minus sign.

The character \$ can be used in FORTRAN only as Hollerith text in a FORMAT statement.

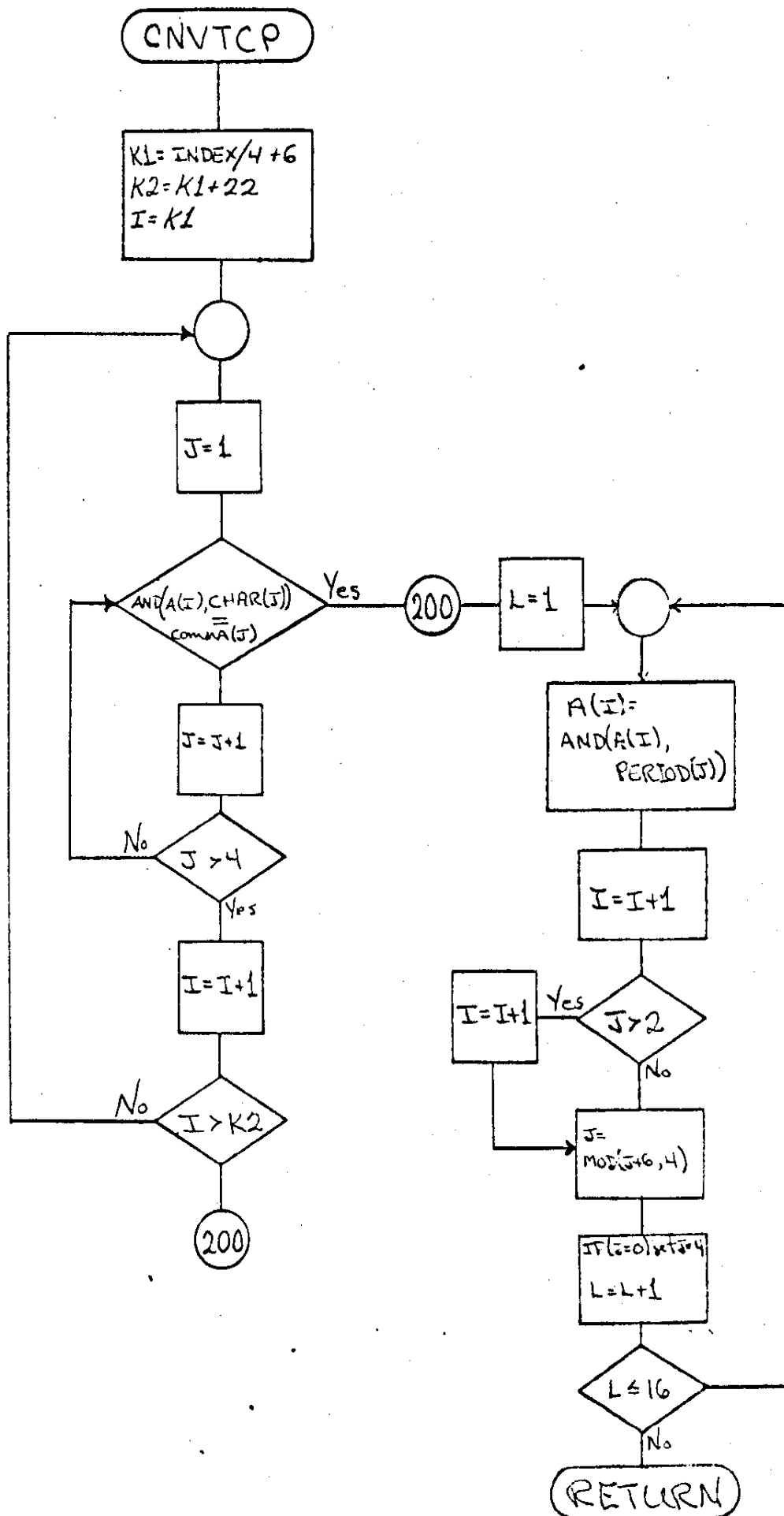
2.2.4 Method

The first word to be checked is the word containing the vector position (time words or blanks are skipped). The four characters in each word are checked until a comma is found. That character is changed to a period and every sixth character for fifteen more times is changed to a period.

The tests are performed on integers because the numbers are not really floating point and test as zero despite their bit configuration when floating point tests are used.

2.2.5 Comments

This method of course pre-supposes that the data is in core in EBCDIC. If the in core format is different, the masks at least must be different.



2.3 FUNCTION ALPHA

2.3.1 Purpose

This function generates the EBCDIC representation of an integer with four or fewer decimal digits.

2.3.2 Calling Sequence

KI = ALPHA(I)

where I is the integer to be converted and KI is the resulting characters right adjusted in the word.

2.3.3 Key Variables

J1	The left most digit
J2	The next digit
J3	The third digit from the left
ZONES	A mask to set the zone bits

2.3.4 Method

The digits are isolated by integer division. Then the characters are constructed by moving the digits to the appropriate positions (by multiplying by powers of two) and inserting the zone bits.

2.3.5 Comments

There is a faster way to accomplish the purpose of ALPHA. This alternate method is to make ALPHA an array of dimension 255 which is initialized with the character representations of the indices.

For example, ALPHA(10) would contain ' 10'.

Both methods are of course machine dependent.

Section 3

USAGE INSTRUCTIONS

3.1 DESCRIPTION

This set of three subprograms can be used to read logical records from a standard Explorer 33 or Explorer 35 Individual Vector tape.

3.2 SYSTEM AND EQUIPMENT

The program is written in FORTRAN IV for use on the IBM/360. The cards are punched in EBCDIC. As discussed elsewhere in this document, this program may not be usable on other machines. One seven-track tape drive is required.

3.3 CONTROL CARDS

The program expects the input tape to be mounted on FORTRAN logical unit 7. Thus for the IBM/360, data definition cards should have the form:

Col 1	Col 16	Col 72
//GJ.FT07F001	DD UNIT=TAPE, DISP=(OLD, KEEP), LABEL=(1, BLP),	X
//	DCB=(TRTCH=T, RECFM=F, DEN=1, LRECL=2760, BLKSIZE	X
	=2760),	
//	VOLUME=SER=EXPV	

3.4 CALLING THE ROUTINE

The calling sequence is:

CALL IVREAD (ITIME, L, VECTOR, NVEC)

all of these variables are output from IVREAD.

ITIME is an integer array of size five which will contain the sequence number and time when return is made to the calling program after reading the first vector of a sequence. Reading other vectors does not reset ITIME.

L is the position of the vector in the original sequence (1-14). The first vector is not necessarily position 1, but positions are in ascending order.

VECTOR is a real array of size sixteen containing the data.

NVEC is the number of vectors in the sequence and is returned only by the first vector of a sequence. However it must have the correct value on subsequent calls since it is used internally by IVREAD. Figure 3-1 is the listing of a sample program which reads and prints thirty sequences.

13/43/12

DATE = 72063

MAIN

FORTRAN IV LEVEL 20

```

0001 DIMENSION ITIME(5), VECTOR(16)
0002 DO 100 I=1,30
0003 CALL IVREAD(ITIME,L,VECTOR,NVEC)
0004 WRITE(6,10) ITIME,L,VECTOR,NVEC
0005 10 FORMAT(1X,I6,I4,3I3,I3,16F6.1,I6)
0006 N = NVEC
0007 DO 50 J=2,N
0008 CALL IVREAD(ITIME,L,VECTOR,NVEC)
0009 WRITE(6,11) L,VECTOR
0010 11 FORMAT(20X,I3,16F6.1)
0011 50 CONTINUE
0012 100 CONTINUE
0013 STOP
0014 END

```

Figure 3-1
Sample Program to Read and Print 30 Sequences

3.5 RESTRICTIONS

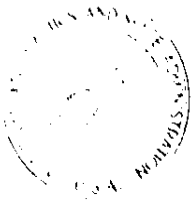
A complete tape descriptions is given in Appendix A.

If the tape should develop a bad record (parity error, redundant record, etc.) the program will not detect it or compensate for it in any way. The exact treatment will depend on the computer operating system.

No test for end of data (end of file) is made in the reading routine.

Appendix A

DESCRIPTION OF THE EXPLORER INDIVIDUAL VECTOR TAPE



NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
AMES RESEARCH CENTER
MOFFETT FIELD, CALIFORNIA 94035



REPLY TO
ATTN OF:

SST:245-3

January 8, 1975

Dr. Joseph H. King
Code 601.1
National Space Science Data Center
Goddard Space Flight Center
Greenbelt, Maryland 20771

Dear Dr. King:

Continuing our transmission of Explorer 35 Ames magnetometer data, we enclose 52 reels of magnetic tape comprising 52 weeks of individual vectors at the 6 second sampling rate. We plan to send such tapes for all weeks covered by the plots which you now have. In November we sent you tapes for 82 second averages of these data.

Details of the time period covered are contained in the enclosed packing list. We also enclose the format and documentation for a program to read the tapes on an IBM machine.

For data after week 211 there is a source of noise and possible uncertainty in the direction of the spin plane component. This shows up as noise above frequencies of 0.01 Hz, so that rapid resolution data for these weeks has principal value in identification of transients rather than Fourier transform at high frequencies.

Yours Sincerely,

David S. Colburn
Research Scientist
Space Science Division

Deck for TECHNICAL NOTE 122

IS INCLUDED IN BOX 6

Appendix B

AMES, IBM/360 RE-READ CAPABILITY

360/50 FORMAT CONVERSION AND RE-READ CAPABILITY

TAPE 99 FORMAT CONVERSION

Format conversion without actual input or output is now available for FORTRAN programs under the regular operating system on the 360/50.

A formatted write statement referring to unit 99 will place the generated string of output characters in a special buffer instead of transmitting it to an output device. The contents of this buffer may then be read (usually under a different format) by a formatted read statement referring to unit 99. For example, the program statements

```
      I = 641
      WRITE(99,100) I,I
      READ(99,101) A,B
100   FORMAT(2I4)
101   FORMAT(A4,F4.2)
```

will place a blank followed by the three characters 641 in A , and the floating-point number 6.41 in B . READ(K,) or WRITE(K,) , where K has the value 99 , is also permitted.

A WRITE(99,) statement fills the entire buffer with blanks before placing the generated output in it. READ(99,) statements do not alter the contents of the buffer, and it may be re-read as often as is wanted. The buffer is 133 characters long, and an attempt to write or read a single record longer than 133 characters will terminate the job with an error message. A READ(99,) statement that calls for more than one record simply re-reads the buffer for each record. If the read statements in the sample above were

```
      READ(99,101) A,B,C,D
```

then C would receive the same contents as B , and D would receive the same contents as B . If a WRITE(99,) statement produces more than one record, later records simply overlap earlier ones. The buffer is not blanked out between records.

Re-Read Capability

Executing the statement

```
      CALL SETB99(A,NCHAR)
```

causes the first NCHAR characters of the array A to be used instead of the standard buffer. (Note that NCHAR is the number of characters (bytes) in A , not the number of words.) For example, if input card images are to be scanned several times with different formats, one may write

```
DIMENSION A(20)
CALL SETB99(A,80)
READ(5,103) A
103  FORMAT(20A4)
```

The contents of the input card may then be scanned immediately by READ(99,) statements; no WRITE(99,) statement is necessary. The array and buffer length specified in a call of SETB99 will continue in use until another call of SETB99 or the statement

```
CALL RESB99
```

is executed. The latter ("reset buffer 99") reinstates the standard 133-character buffer.

The buffer length NCHAR in a call of SETB99 must be at least 2 . It may be as large as is wanted, provided the array used for the buffer is sufficiently large. If NCHAR is greater than 257 , however, a WRITE(99,) statement will not blank out the buffer properly before writing, and the user must make his write statement insert trailing blanks if they are necessary for correct operation of his program.

Note:

The Ames library subroutine AL CVRT uses the READ(99,), WRITE(99) feature. This subroutine is recommended for use, when a program does not have large amounts of format conversion.

Appendix C

PROGRAM LISTING

C-2

0001

```

C SUBROUTINE IVREAD(ITIME,L,VECTOR,NVEC)
C SUBROUTINE TO READ EXPLORER 33 AND 35 MAGNETOMETER INDIVIDUAL VECTOR
C TAPES GENERATED ON THE IBM 7040/7094 DCS
C THE TAPES ARE BLOCKED IN A MANNER WHICH MAKES THEM UNREADABLE
C DIRECTLY ON ANY OTHER COMPUTER
C THIS ROUTINE MAKES USE OF THE RE-READ CAPABILITY OF UNIT 99,
C THIS FEATURE MAY NOT BE AVAILABLE ON MACHINES OTHER THAN THE IBM/360
C CALLING SEQUENCE...
C ITIME AN INTEGER ARRAY OF SIZE 5 CONTAINING THE INFORMATION
C FROM THE FIRST 5 WORDS OF THE FIRST VECTOR OF THE SEQUENCE,
C NOT CHANGED WHEN OTHER VECTORS ARE READ
C L VECTOR NUMBER, A NUMBER FROM 1 TO 14
C VECTOR A REAL ARRAY OF SIZE 16 CONTAINING THE DATA
C NVEC THE NUMBER OF VECTORS IN THE SEQUENCE

```

0002

```

C LOGICAL FIRST, IREAD
C DATA FIRST, IREAD/.TRUE.,.TRUE./
C DIMENSION VECTOR(16),ITIME(5),A(721)
C INTEGER FMT(9),FMT1(11),BLANK,FMT3,FMT4
C DATA FMT/ (,,,'',(255,,X),,,,'X,19','X,13','',16F',
C *'6.1)'/
C DATA FMT1/ (,,,'',(255,,X),,,,'X,16','',14,,',413',
C *'16F6',,1,1,,6) '/
C DATA FMT3/ (255',,FMT4/,X), '/
C DATA BLANK /, '/
C INTEGER ALPHA
C DATA M/2/

```

0007

0008

0009

0010

0011

0012

0013

0014

0015

0016

0017

0018

0019

0020

0021

0022

0023

```

C 100 CONTINUE
C DO WE NEED TO READ A NEW RECORD
C IF(IREAD) GO TO 1000
C IS NEXT VECTOR THE FIRST OF A SEQUENCE
200 IF(IVECTOR.EQ.1) GO TO 300
C READ A NON-FIRST VECTOR
N2 = INDEX
FMT(3) = FMT3
FMT(4) = FMT4
IF(INDEX.LE.255) GO TO 210
N1 = INDEX/255
N2 = MOD(INDEX,255)
IF(N2.EQ.0) N1 = N1-1
IF(N2.EQ.0) N2 = 255
FMT(2) = ALPHA(N1)

```

```

IVRD 01
IVRD 02
IVRD 03
IVRD 04
IVRD 05
IVRD 06
IVRD 07
IVRD 08
IVRD 09
IVRD 10
IVRD 11
IVRD 12
IVRD 13
IVRD 14
IVRD 15
IVRD 16
IVRD 17
IVRD 18
IVRD 19
IVRD 20
IVRD 21
IVRD 22
IVRD 23
IVRD 24
IVRD 25
IVRD 26
IVRD 27
IVRD 28
IVRD 29
IVRD 30
IVRD 31
IVRD 32
IVRD 33
IVRD 34
IVRD 35
IVRD 36
IVRD 37
IVRD 38
IVRD 39
IVRD 40
IVRD 41
IVRD 42
IVRD 43

```

13/43/12

DATE = 72063

IVREAD

FORTRAN IV G LEVEL 20

```

0024      GO TO 212
0025      FMT (2) = BLANK
0026      FMT (3) = BLANK
0027      FMT (4) = BLANK
0028      FMT (5) = ALPHA(N2)
0029      C CONVERT COMMAS TO PERIODS
0030      CALL CNVTCP(A,INDEX)
0031      READ(99,FMT) L,VECTOR
0032      IVEC = IVEC + 1
0033      INDEX = INDEX + 126
0034      IF(IVEC.GT.NVEC) IVEC = 1
0035      IF(INDEX+126.GT.IEND) IREAD=.TRUE.
0036      IF(IVEC.EQ.1 .AND. INDEX+132.GT.IEND) IREAD=.TRUE.
0037      RETURN
0038      C READ THE FIRST VECTOR OF A SEQUENCE
0039      300 CONTINUE
0040      N2 = INDEX
0041      FMT1(3) = FMT3
0042      FMT1(4) = FMT4
0043      IF(INDEX.LE.255) GO TO 310
0044      N1 = INDEX/255
0045      N2 = MOD(INDEX,255)
0046      IF(N2.EQ.0) N1 = N1-1
0047      IF(N2.EQ.0) N2 = 255
0048      FMT1(2) = ALPHA(N1)
0049      GO TO 312
0050      FMT1(2) = BLANK
0051      FMT1(3) = BLANK
0052      FMT1(4) = BLANK
0053      FMT1(5) = ALPHA(N2)
0054      CALL CNVTCP(A,INDEX)
0055      C CONVERT COMMAS TO PERIODS
0056      READ(99,FMT1) ITIME,L,VECTOR,NVEC
0057      IVEC = IVEC + 1
0058      INDEX = INDEX + 132
0059      IF(INDEX+126.GT.IEND) IREAD=.TRUE.
0060      RETURN
0061      C TIME TO READ A NEW RECORD
0062      1000 CONTINUE
0063      C FIRST ENTRY REQUIRES INITIALIZATION
0064      IF(FIRST) GO TO 5000
0065      C FIRST MOVE PARTIAL RECORD AT END TO START OF BUFFER
0066      J = 0

```

IVRD 44
IVRD 45
IVRD 46
IVRD 47
IVRD 48
IVRD 49
IVRD 50
IVRD 51
IVRD 52
IVRD 53
IVRD 54
IVRD 55
IVRD 56
IVRD 57
IVRD 58
IVRD 59
IVRD 60
IVRD 61
IVRD 62
IVRD 63
IVRD 64
IVRD 65
IVRD 66
IVRD 67
IVRD 68
IVRD 69
IVRD 70
IVRD 71
IVRD 72
IVRD 73
IVRD 74
IVRD 75
IVRD 76
IVRD 77
IVRD 78
IVRD 79
IVRD 80
IVRD 81
IVRD 82
IVRD 83
IVRD 84
IVRD 85
IVRD 86

```

0061      M = MOD(INDEX,4)
0062      C  MOVE FOUR EXTRA CHARACTERS (1 WORD) TO PREVENT OX IN FORMAT
0063      K1 = INDEX/4
0064      K2 = IEND/4
0065      DO 1100 I=K1,K2
0066      J = J+1
0067      A(J) = A(I)
0068      J = J*4
0069      IEND = 2748 + J
0070      IREAD = .FALSE.
0071      INDEX = M + 4
0072      J = J+1
0073      K1 = J/4+1
0074      K2 = IEND/4
0075      C  READ THE DATA
0076      READ(7,1500) (A(I),I=K1,K2)
0077      1500  FORMAT(12X,100A4,100A4,100A4,100A4,100A4,100A4,87A4)
0078      GO TO 200
0079      C  ENTER HERE ONLY TO INITIALIZE THE BUFFER AREA, ARRAY A WITH 2884 CHAR
0080      5000  CALL SETB99(A,2884)
0081      FIRST = .FALSE.
0082      J=0
0083      IVEC = 1
0084      GO TO 1200
0085      END

```

```

IVRD 87
IVRD 88
IVRD 89
IVRD 90
IVRD 91
IVRD 92
IVRD 93
IVRD 94
IVRD 95
IVRD 96
IVRD 97
IVRD 98
IVRD 99
IVRD 100
IVRD 101
IVRD 102
IVRD 103
IVRD 104
IVRD 105
IVRD 106
IVRD 107
IVRD 108
IVRD 109
IVRD 110
IVRD 111

```

```

0001      SUBROUTINE CNVTCP(A,INDEX)
C      SURROUTINE TO CONVERT THE INTERNAL FORMULA PERIOD
C      (WHICH PRINTS AS A COMMA) TO THE EXTERNAL FORMULA PERIOD.
C      THE FACT THAT THIS IS AN 16F6.1 FORMAT IS USED.
C      SEARCHING STARTS WITH THE VECTOR POSITION.
      INTEGER AND
      INTEGER COMMA(4),PERIOD(4),CHAR(4)
      DATA COMMA/Z6B000000,Z006B0000,Z00006800,Z00000006B /
      DATA PERIOD/Z48FFFFF,ZFF4BFFFF,ZFFFFF4BFF,ZFFFFF4B /
      DATA CHAR/ZFF000000,Z00FF0000,Z0000FF00,Z000000FF/
      INTEGER A(721)
      K1 = INDEX/4 + 6
      K2 = K1 + 22
C
      DO 100 I=K1,K2
      DO 90 J=1,4
      IF(AND(A(I), CHAR(J)).EQ.COMMA(J)) GO TO 200
      90      CONTINUE
      100     CONTINUE
      200     CONTINUE
      DO 300 L=1,16
      A(I) = AND(PERIOD(J),A(I))
      I = I+1
      IF(J.GT.2) I = I+1
      J = MOD(J+6,4)
      300     IF(J.EQ.0) J = 4
      RETURN
      END

```

CNVTCPO1
 CNVTCPO2
 CNVTCPO3
 CNVTCPO4
 CNVTCPO5
 CNVTCPO6
 CNVTCPO7
 CNVTCPO8
 CNVTCPO9
 CNVTCPO10
 CNVTCPO11
 CNVTCPO12
 CNVTCPO13
 CNVTCPO14
 CNVTCPO15
 CNVTCPO16
 CNVTCPO17
 CNVTCPO18
 CNVTCPO19
 CNVTCPO20
 CNVTCPO21
 CNVTCPO22
 CNVTCPO23
 CNVTCPO24
 CNVTCPO25
 CNVTCPO26
 CNVTCPO27
 CNVTCPO28

C-5

```

0001      INTEGER FUNCTION ALPHA(I)
0002      INTEGER OR,ZONES
0003      C THIS FUNCTION CHANGES AN INTERNAL INTEGER OF 4 OR LESS DECIMAL DIGITS
0004      C TO EBCDIC (CHARACTER) REPRESENTATION
0005      C IF SPACE IS NO PROBLEM, A FASTER METHOD WOULD BE TO DECLARE ALPHA AN
0006      C ARRAY AND USE A DATA STATEMENT TO PUT THE EBCDIC CHARACTERS IN THE
0007      C ARRAY, ONLY NUMBERS 1 THROUGH 255 ARE NEEDED
0008      DATA ZONES/ZF0F0F0F0/
0009      K = I
0010      J1 = K/1000
0011      K = K-J1*1000
0012      J2 = K/100
0013      K = K-J2*100
0014      J3 = K/10
0015      K = K-J3*10
0016      K = J1*(2**24) + J2*(2**16) + J3*(2**8) + K
0017      ALPHA = OR(ZONES,K)
0018      RETURN
0019      END

```

ALPHA001
ALPHA002
ALPHA003
ALPHA004
ALPHA005
ALPHA006
ALPHA007
ALPHA008
ALPHA009
ALPHA010
ALPHA011
ALPHA012
ALPHA013
ALPHA014
ALPHA015
ALPHA016
ALPHA017
ALPHA018
ALPHA019

The Explorer 33 and Explorer 35 sequence **summary** tapes have the following physical characteristics:

one file

density: 556 bpi

7-track

parity: odd

physical record length: 2760 characters (nominal,
BCD characters

The first twelve characters of each physical record contain the physical record count and tape number of the original tape. These twelve characters are skipped by the SAREAD routine. The first six characters are binary, the second six are BCD.

Each logical record is a sequence summary. The format of a sequence summary is shown in Figure A-1. Preceding every logical record is a six character word which contains binary information describing the preceding logical record (if any) and the following logical record.

We could view the final tape format as having been created in the following manner. All logical records are stored in order each preceded by its six character control word. Then a group of 2748 characters is picked off, prefixed by a twelve character physical record control word and placed on tape followed by an inter-record gap. The second physical record would contain the second 2748 characters prefixed by twelve control characters and followed by an inter-record gap. At the end of the data, the last physical record will contain however many characters remain less than or equal to 2742, suffixed by a six character logical record control

Sequence Number	Dev. Y	Dev. Z	Pl. Sec	Magitude	SOLAR			EQUATORIAL			SOLAR		
					Theta	P ₁₁₂	Y	Z	Dev. Y	Dev. Z	Theta	P ₁₁₂	Y
1	4	3	3	5	6	6	5	5	5	5	6	6	5
I	I	I	I	I	I	I	I	I	I	I	I	I	I
7	11	17	20	25	31	31	42	47	52	57	63	69	74

length
representable
character

MAGNETOSPHERIC	Dev. X	Dev. Y	Dev. Z	SOLAR			ECLIPTIC			SOLAR		
				Theta	P ₁₁₂	Y	Z	Dev. Y	Dev. Z	Theta	P ₁₁₂	Y
5	5	5	5	6	6	5	5	5	5	1	1	1
I	I	I	I	I	I	I	I	I	I	I	I	I
81	94	99	105	111	116	121	126	131	136	141	146	151

length
representable
character

FIGURE A-1
SEQUENCE SUMMARY FORMAT

word and preceded by a twelve character control word then followed by a single end of file. A schematic representation of this concept is shown in Figure A-2.

Figure A-3 shows the twenty three formats which a physical record may have. These twenty three formats are also shown in the listing of the block data subprogram, but that form is hard to read.

[illegible]

Record 1

LC = Logical Control Word
PC = Physical Control Word

[illegible]

Record 2

FIGURE A-2
SCHEMATIC REPRESENTATION OF LOGICAL RECORDS
WITHIN PHYSICAL RECORDS

THE PHYSICAL RECORD FORMATS

[illegible]

FILE 1 RECORD 1 2760 BYTES

```
( 0) 002600000001 600106001101 0000000000026 606001040301 016001111160 010560041060 010360030403 016060600501
( 48) 016060010407 046001000600 600204040560 010104106060 01110606060 040601606001 031064001006 051160010401
( 96) 006002020602 606007041160 010100066060 030605606060 050310606010 050306606010 020260020503 056001010602
( 144) 606060101060 00026000026 606001040301 106001111160 106001111160 040560030202 116060600310 006060010407
( 192) 106001030410 60011071160 600101116060 021103606060 030305606001 040301400201 040360010600 106001070705
( 240) 606001001060 600105116060 020704606060 040007606001 050107600101 050360020031 116060601101 606003003360
( 288) 00026000026 606001040303 606001040303 010660010560 021160020511 066060600304 106060010411 076060101010
( 336) 600102021160 0501076060 040704606060 030103606001 040600400105 020460010002 066001010107 606010050260
( 384) 010006106060 050706606060 030705606001 050303606007 066060010300 106001010407 60600310660 00026000026
( 432) 606001040303 036001111160 010660011060 010260020306 066060600310 116040040301 606011060260 606011060260
( 480) 010401026060 110406606060 040010606001 110607400101 011060400303 056001000010 606006070360 010403076060
( 528) 110010606060 030603606002 000510604005 040160601101 046001031003 606011100760 000026000026 606001040304
( 576) 016001111160 010660021160 600760020405 116060060020 066001030502 046060111101 606007071160 606060606060
( 624) 050105606060 010701606001 050006400110 111060010006 106060060700 606004071060 60605076060 050111606060
( 672) 020205606001 050403606011 010260601007 016060606060 606005010660 000026000026 606001040304 06600111160
( 720) 010660030560 050660601006 006060600410 006060010104 106060030703 606004050660 600210106060 011103606060
( 768) 040211606001 010205604001 070360600401 056060040110 606005001060 600210016060 020004606060 050210606001
( 816) 010707606003 02106060410 116060021104 606001100360 000026000026 606001040304 07600111160 010660030760
( 864) 011060020207 07606060110 046060010407 116001010405 606007010760 606004056060 020102606001 041106600100
( 912) 040605400110 021060010200 116060060004 606060040160 606005056060 010006606060 020102606001 041106600100
( 960) 070260601002 036060600305 606001010460 000026000026 606001040304 10600111160 010660031060 031160002105
( 1008) 056060600204 026060010501 066060071105 606007050600 601102046060 020707606060 020104606001 041105400104
( 1056) 070060601006 036060607000 60606060560 6011000116060 030407606060 020607606001 050311606007 0200060401002
( 1104) 036060110400 606002011160 60606060026 606001040304 11600111160 010660040060 600160601104 01606060306
( 1152) 046060010401 106060040403 606005021060 600103026060 010311606060 030300606001 031100604005 060460600410
( 1200) 116060041006 606002031060 600101116060 010500606060 031106606001 04052606003 020360602021 106060010505
( 1248) 606001026060 00026000026 606001040305 00600111160 010660040160 600306606001 050505400110 076060010300
( 1296) 116060100401 606002001060 600207106060 06071606060 60077606060 600306606001 050505400110 076060010300
( 1344) 606002050260 600303106060 600303106060 060411606060 60071606060 600306606001 050505400110 076060010300
( 1392) 000026000026 606001040305 01600111160 010660040660 010660040260 030111606060 020360602021 106060010300
( 1440) 606004000260 600203046060 600203046060 020404606060 030111606060 020001504002 101060600411 076060030510
( 1488) 600201056060 020601606060 040100606001 040100606001 020403606004 0506060010407 060002020660 060002020660
( 1536) 606001040305 04600111160 010660040305 010660040305 05060011005 020260600404 046060020501 606002020660
( 1584) 600410046060 030107606060 030107606060 030107606060 040600400103 026060600111 066060041104 046060100600
( 1632) 030501606060 020205606001 020205606001 041102606010 000260600604 116060050007 606002071160 000026000026
( 1680) 056001111160 010660041060 010660041060 050602606001 070403606060 0506060010605 056060600607 606004020760
( 1728) 051007606001 070403606060 070403606060 050602606001 070403606060 0506060010605 056060600607 606004020760
( 1776) 010660041160 030460601003 030460601003 046060600501 046060600501 046060600501 046060600501 606001040305
( 1824) 050004606001 060610606003 060610606003 030660606007 030660606007 106060040107 606004020260 606004020260
( 1872) 071106606060 600260606042 046060041101 606002020060 116060100402 606003040560 600403046060 021010606001
( 1920) 050660011003 00606060111 006060010500 076060050206 606060010560 60600116060 60600116060 010660050060
( 1968) 041106400105 010260601010 076060050206 60606001060 60600116060 60600116060 60600116060 010660050060
( 2016) 100160600607 036060600200 036060600200 036060600200 036060600200 036060600200 036060600200 010660050060
( 2064) 066060600301 036060010502 036060010502 036060010502 036060010502 036060010502 036060010502 010660050060
( 2112) 010660040311 016060040401 016060040401 016060040401 016060040401 016060040401 016060040401 05003604006
( 2160) 036060600301 606060600360 606060600360 606060600360 606060600360 606060600360 606060600360 00106600500
( 2208) 016060010502 116060030503 116060030503 116060030503 116060030503 116060030503 116060030503 046060600301
( 2256) 116060040307 606060600260 606060600260 606060600260 606060600260 606060600260 606060600260 01160600310
( 2304) 606060600360 00026000026 60606000026 60606000026 60606000026 60606000026 60606000026 116060600301
( 2352) 016060030560 606004051160 606004051160 606004051160 606004051160 606004051160 606004051160 046060600301
( 2400) 606060600260 606004051160 606004051160 606004051160 606004051160 606004051160 606004051160 066060010503
( 2448) 000026000026 606001040306 606001040306 606001040306 606001040306 606001040306 606001040306 066060600301
( 2496) 60600401160 606005606060 606005606060 606005606060 606005606060 606005606060 606005606060 066060600301
( 2544) 60600401160 606005606060 606005606060 606005606060 606005606060 606005606060 606005606060 066060600301
( 2592) 60600401160 606005606060 606005606060 606005606060 606005606060 606005606060 606005606060 066060600301
```

(1824) 600360600460 030460606003 056060400307 006060010201 066060600202 606040020060 606060066060 606011606040
(1872) 010501606040 010460606002 116060604005 116060604005 606060600560 606060016060 600101606040 030210606001
(1920) 011110606060 020460604001 106060606005 606060601160 00026000026 600300030103 606001001060 600360600032
(1968) 050660606003 116060400411 026060010110 016060600202 606040021160 606060036060 606004606040 60606060003
(2016) 001103606040 010260606003 046060400104 606060600360 606060026060 606004606040 606004606001 010507606060
(2064) 020560604002 076060606003 606060600460 60606000026 600300030103 606001001060 600360600760 011060606003
(2112) 066060400205 116060010211 106060604001 606060600460 606000030103 6060010560 606007606060 020407606040
(2160) 020660606002 106060604001 606060600260 00026000026 606060016060 60600056060 606002606001 020560604001
(2208) 036060606003 606060600660 076060600204 60606060026 606060016060 606003606040 60606060003 07606060003
(2256) 046060010207 076060600260 606060600260 60606001160 606060016060 606003606040 60606060003 07606060003
(2304) 016060604004 606060600260 606060600260 606060016060 606060016060 606003606040 60606060003 07606060003
(2352) 606060600260 00026000026 60606000026 60606000026 60606000026 60606000026 60606060003 07606060003
(2400) 016060600107 606060600107 606060600107 606060600107 606060600107 606060600107 606060600107 07606060003
(2448) 606060600260 606060600260 606060600260 606060600260 606060600260 606060600260 606060600260 07606060003
(2496) 00026000026 606060600260 606060600260 606060600260 606060600260 606060600260 606060600260 07606060003
(2544) 606040030060 606060600260 606060600260 606060600260 606060600260 606060600260 606060600260 07606060003
(2592) 606060026060 606060600260 606060600260 606060600260 606060600260 606060600260 606060600260 07606060003
(2640) 600300030104 006001001060 606060600260 606060600260 606060600260 606060600260 606060600260 07606060003
(2688) 606060056060 606060600260 606060600260 606060600260 606060600260 606060600260 606060600260 07606060003
(2736) 606003606040 040200606001 110007606060 400560604002 021060606060 066060400205 606060600360 606060600360

REGULAR OCTAL DUMP OF DS000850

FILE 41 RECORD 314 2048 BYTES

(0) 005340000472 216262312745 600405606003 020105606040 020760606060 036060606002 606060600160 400026000026
(48) 600301000403 026001010560 60060050360 60060606004 036060606010 046060600205 106060600205 606060600660
(96) 606060016060 606001606040 60001606003 020407606060 036060604002 046060604011 606060600160 606060600660
(144) 606001606060 600501606003 020303606040 020560606060 036060606001 606060600160 606060600160 606060600660
(192) 036001010560 60060050660 020260606004 03606060606 0760606030204 066060400204 606060600560 606060600660
(240) 606004606040 010302606003 020603606060 030460604002 036060604011 606060600160 606060600560 606060600660
(288) 600304606003 020402606040 020560606060 026060606001 606060600460 600026000026 606060600160 606060600660
(336) 60060050760 04046060604 026060604003 036060604001 116060400106 606060600260 606060600160 606060600660
(384) 020302606003 021110606060 0360606060 036060604001 036060600260 606060600260 606060600160 606060600660
(432) 020301606040 020560606040 056060606001 056060600260 606060600260 606060600260 606060600160 606060600660
(480) 60066060604 036060606002 036060600203 036060600205 606060600160 606060600160 606060600160 606060600660
(528) 020703606060 030460604002 026060600103 606060600160 606060600160 606060600160 606060600160 606060600660
(576) 020560606040 006060606001 606060600160 606060600160 606060600160 606060600160 606060600160 606060600660
(624) 046060606005 026060600203 046060400101 606060600160 606060600160 606060600160 606060600160 606060600660
(672) 030560604002 016060600260 046060400101 606060600160 606060600160 606060600160 606060600160 606060600660
(720) 016060606001 606060600160 606060600160 606060600160 606060600160 606060600160 606060600160 606060600660
(768) 1160606030203 00026000026 606060600160 606060600160 606060600160 606060600160 606060600160 606060600660
(816) 026060400103 606060600160 606060600160 606060600160 606060600160 606060600160 606060600160 606060600660
(864) 606060606060 00026000026 606060600160 606060600160 606060600160 606060600160 606060600160 606060600660
(912) 016060400205 606060600260 606060600260 606060600260 606060600260 606060600260 606060600260 606060600660
(960) 606060600260 606060600160 606060600160 606060600160 606060600160 606060600160 606060600160 606060600660
(1008) 00026000026 600301000403 116001010560 606001606060 606060600160 606060600160 606060600160 606060600660
(1056) 606060600360 606060600160 606060600160 606060600160 606060600160 606060600160 606060600160 606060600660
(1104) 606060016060 606001606060 600100606060 020310606040 020560606060 020560606060 020560606060 606060600660

REQ. AGENT
CMT
VJP

RAND NO.
RC2827
RC3358

ACQ. AGENT
JHK

IMP-E

5.12 SEC. MAG. VECTORS

67-070A-03D
(RESTORED)

This data set catalogue contains 22 data tapes of 5.12 second magnetic vectors. The tapes represent a restoration of 210 original tapes which were written on an IBM 7094 DCS. The original tapes were 556 BPI, binary and written on 7 tracks. The restored tapes are 6250 BPI and 9 track. In restoring the tapes, the 6-bit bytes of the original tapes were expanded to 8-bit bytes. Also the DCS format of the original tapes was converted to ASCII during the restoration. Each file on the restored tapes is a restoration of one of the original tapes. In determining the start and stop times consider January 1 to be day zero.

The contents of the restored tapes are as follows:

<u>DR#</u>	<u>DS#</u>	<u>FILE</u>	<u>RESTORATION OF</u>	<u>TIME SPAN</u>
DR01016	DS01016	1	D-17774	07/19/67 - 07/27/67
		2	D-17782	07/27/67 - 08/03/67
		3	D-17783	08/03/67 - 08/10/67
		4	D-17772	08/10/67 - 08/17/67
		5	D-17784	08/17/67 - 08/24/67
		6	D-17785	08/24/67 - 08/30/67
		7	D-17786	08/31/67 - 09/07/67
		8	D-17787	09/07/67 - 09/14/67
		9	D-17819	09/14/67 - 09/21/67
DR01017	DS01017	1	D-17788	09/21/67 - 09/27/67
		2	D-17773	09/28/67 - 10/04/67
		3	D-17789	10/05/67 - 10/12/67
		4	D-17777	10/12/67 - 10/19/67
		5	D-17776	10/19/67 - 10/25/67
		6	D-17815	10/26/67 - 11/02/67
		7	D-17775	11/02/67 - 11/09/67
		8	D-17778	11/09/67 - 11/15/67
		9	D-17779	11/16/67 - 11/23/67
DR01018	DS01018	1	D-17780	11/23/67 - 11/29/67
		2	D-17781	11/30/67 - 12/07/67
		3	D-17818	12/07/67 - 12/14/67
		4	D-17817	12/14/67 - 12/20/67
		5	D-17816	12/21/67 - 12/28/67
		6	D-17795	01/04/68 - 01/11/68
		7	D-17794	01/11/68 - 01/17/68
		8	D-17792	01/18/68 - 01/25/68
		9	D-17793	01/25/68 - 01/31/68

<u>DR#</u>	<u>DS#</u>	<u>FILE</u>	<u>RESTORATION OF</u>	<u>TIME SPAN</u>
DR01019	DS01019	1	D-17823	02/01/68 - 02/08/68
		2	D-17822	02/08/68 - 02/15/68
		3	D-17821	02/15/68 - 02/22/68
		4	D-17801	02/29/68 - 03/07/68
		5	D-17812	03/07/68 - 03/13/68
		6	D-17808	03/14/68 - 03/21/68
		7	D-17800	03/21/68 - 03/28/68
		8	D-17799	03/28/68 - 04/04/68
		9	D-17807	04/04/68 - 04/11/68
DR01020	DS01020	1	D-17798	04/11/68 - 04/13/68
		2	D-17797	04/13/68 - 04/18/68
		3	D-17820	04/18/68 - 04/25/68
		4	D-19218	04/25/68 - 05/02/68
		5	D-19219	05/02/68 - 05/09/68
		6	D-19220	05/09/68 - 05/16/68
		7	D-19221	05/16/68 - 05/23/68
		8	D-19222	05/23/68 - 05/30/68
		9	D-19223	05/30/68 - 06/06/68
		10	D-19224	06/06/68 - 06/13/68
DR01021	DS01021	1	D-19225	06/13/68 - 06/20/68
		2	D-19226	06/20/68 - 06/27/68
		3	D-19227	06/27/68 - 07/04/68
		4	D-19228	07/04/68 - 07/11/68
		5	D-19229	07/11/68 - 07/17/68
		6	D-19230	07/18/68 - 07/25/68
		7	D-19231	07/25/68 - 08/01/68
		8	D-19232	08/01/68 - 08/08/68
		9	D-19233	08/08/68 - 08/15/68
DR01022	DS01022	1	D-19234	08/15/68 - 08/22/68
		2	D-19235	08/22/68 - 08/26/68
		3	D-19236	08/29/68 - 09/05/68
		4	D-19237	09/05/68 - 09/12/68
		5	D-18589	09/12/68 - 09/19/68
		6	D-18590	09/19/68 - 09/26/68
		7	D-18591	09/26/68 - 10/03/68
		8	D-18592	10/03/68 - 10/06/68
		9	D-18593	10/06/68 - 10/07/68
DR01023	DS01023	1	D-18594	10/07/68 - 10/10/68
		2	D-18595	10/10/68 - 10/17/68
		3	D-18596	10/17/68 - 10/24/68
		4	D-18597	10/24/68 - 10/31/68
		5	D-18598	10/31/68 - 11/07/68
		6	D-18599	11/07/68 - 11/14/68
		7	D-18570	11/14/68 - 11/21/68
		8	D-19010	11/21/68 - 11/28/68
		9	D-18571	11/28/68 - 12/05/68

<u>DR#</u>	<u>DS#</u>	<u>FILE</u>	<u>RESTORATION OF</u>	<u>TIME SPAN</u>
DR01024	DS01024	1	D-18572	12/05/68 - 12/12/68
		2	D-18573	12/12/68 - 12/19/68
		3	D-18574	12/26/68 - 12/31/68
		4	D-18575	01/01/69 - 01/01/69
		5	D-18569	01/02/69 - 01/09/69
		6	D-19238	01/09/69 - 01/16/69
		7	D-19011	01/16/69 - 01/23/69
		8	D-18576	01/23/69 - 01/30/69
		9	D-18577	01/30/69 - 02/06/69
		10	D-19541	02/06/69 - 02/13/69
DR01025	DS01025	1	D-19012	02/13/69 - 02/20/69
		2	D-18578	02/20/69 - 02/27/69
		3	D-18600	02/27/69 - 03/03/69
		4	D-18601	03/03/69 - 03/06/69
		5	D-18602	03/06/69 - 03/13/69
		6	D-18603	03/13/69 - 03/20/69
		7	D-18604	03/20/69 - 03/27/69
		8	D-18605	03/27/69 - 04/03/69
		9	D-18606	04/03/69 - 04/09/69
		10	D-18607	04/10/69 - 04/17/69
DR01026	DS01026	1	D-18608	04/17/69 - 04/24/69
		2	D-19239	04/24/69 - 05/01/69
		3	D-18609	05/01/69 - 05/08/69
		4	D-18619	05/08/69 - 05/15/69
		5	D-18618	05/15/69 - 05/22/69
		6	D-18617	05/22/69 - 05/29/69
		7	D-18616	05/29/69 - 06/05/69
		8	D-19013	06/05/69 - 06/11/69
		9	D-18615	06/12/69 - 06/15/69
		10	D-18614	06/15/69 - 06/19/69
DR01027	DS01027	1	D-19014	06/19/69 - 06/26/69
		2	D-18613	06/26/69 - 07/03/69
		3	D-18612	07/03/69 - 07/10/69
		4	D-19542	07/17/69 - 07/24/69
		5	D-18611	07/24/69 - 07/31/69
		6	D-18610	07/31/69 - 08/07/69
		7	D-18588	08/07/69 - 08/10/69
		8	D-18587	08/10/69 - 08/14/69
		9	D-18586	08/14/69 - 08/21/69
		10	D-18585	08/21/69 - 08/28/69
DR01028	DS01028	1	D-18584	08/28/69 - 09/04/69
		2	D-18583	09/04/69 - 09/11/69
		3	D-18582	09/11/69 - 09/17/69
		4	D-18581	09/18/69 - 09/25/69
		5	D-18580	09/25/69 - 10/01/69
		6	D-18579	10/01/69 - 10/08/69
		7	D-18984	10/08/69 - 10/09/69
		8	D-18985	10/09/69 - 10/16/69
		9	D-18986	10/16/69 - 10/23/69
		10	D-18987	10/23/69 - 10/30/69

<u>DR#</u>	<u>DS#</u>	<u>FILE</u>	<u>RESTORATION OF</u>	<u>TIME SPAN</u>
DR01029	DS01029	1	D-18988	10/30/69 - 11/05/69
		2	D-18989	11/06/69 - 11/13/69
		3	D-18990	11/13/69 - 11/20/69
		4	D-18991	11/20/69 - 11/26/69
		5	D-18992	11/27/69 - 12/01/69
		6	D-18993	11/30/69 - 12/01/69
		7	D-18994	12/01/69 - 12/04/69
		8	D-19240	12/04/69 - 12/11/69
		9	D-19543	12/11/69 - 12/18/69
		10	D-19025	12/18/69 - 12/25/69
DR01030	DS01030	1	D-19026	12/25/69 - 01/01/70
		2	D-19027	01/01/70 - 01/08/70
		3	D-19028	01/08/70 - 01/15/70
		4	D-19029	01/15/70 - 01/22/70
		5	D-19030	01/22/70 - 01/29/70
		6	D-19031	01/28/70 - 01/29/70
		7	D-19032	01/29/70 - 02/05/70
		8	D-19034	02/05/70 - 02/06/70
		9	D-19033	02/06/70 - 02/11/70
		10	D-19024	02/11/70 - 02/19/70
DR01031	DS01031	1	D-19023	02/19/70 - 02/25/70
		2	D-19022	02/26/70 - 03/05/70
		3	D-19021	03/05/70 - 03/11/70
		4	D-19020	03/12/70 - 03/19/70
		5	D-19019	03/19/70 - 03/26/70
		6	D-19018	03/26/70 - 04/02/70
		7	D-19017	04/02/70 - 04/09/70
		8	D-19035	04/09/70 - 04/16/70
		9	D-19036	04/16/70 - 04/23/70
		10	D-19015	04/23/70 - 04/23/70
DR01032	DS01032	1	D-19016	04/30/70 - 05/07/70
		2	D-19037	05/07/70 - 05/14/70
		3	D-19038	05/14/70 - 05/21/70
		4	D-19039	05/21/70 - 05/28/70
		5	D-19040	05/28/70 - 06/04/70
		6	D-19041	06/04/70 - 06/11/70
		7	D-19042	06/11/70 - 06/18/70
		8	D-19043	06/18/70 - 06/25/70
		9	D-19044	06/25/70 - 07/01/70
		10	D-19045	07/02/70 - 07/09/70
DR01033	DS01033	1	D-19046	07/09/70 - 07/15/70
		2	D-19047	07/16/70 - 07/23/70
		3	D-19048	07/23/70 - 07/29/70
		4	D-19049	07/29/70 - 08/06/70
		5	D-19050	08/06/70 - 08/13/70
		6	D-19051	08/13/70 - 08/20/70
		7	D-19052	08/20/70 - 08/27/70
		8	D-19053	08/27/70 - 09/03/70
		9	D-19054	09/03/70 - 09/10/70
		10	D-18995	09/10/70 - 09/17/70

<u>DR#</u>	<u>DS#</u>	<u>FILE</u>	<u>RESTORATION OF</u>	<u>TIME SPAN</u>
DR01034	DS01034	1	D-18996	09/17/70 - 09/24/70
		2	D-18997	09/24/70 - 10/01/70
		3	D-18998	10/01/70 - 10/08/70
		4	D-18999	10/08/70 - 10/15/70
		5	D-19000	10/15/70 - 10/22/70
		6	D-19001	10/22/70 - 10/26/70
		7	D-19002	10/29/70 - 11/05/70
		8	D-19003	11/05/70 - 11/12/70
		9	D-19004	11/12/70 - 11/19/70
		10	D-19005	11/19/70 - 11/26/70
DR01035	DS01035	1	D-19006	11/26/70 - 12/03/70
		2	D-19007	12/03/70 - 12/10/70
		3	D-19008	12/10/70 - 12/17/70
		4	D-19009	12/17/70 - 12/24/70
		5	D-17796	07/29/71 - 08/05/71
		6	D-17806	08/05/71 - 08/11/71
		7	D-17810	08/12/71 - 08/19/71
		8	D-17809	08/19/71 - 08/25/71
		9	D-17805	08/26/71 - 08/28/71
		10	D-17790	08/29/71 - 09/02/71
DR01036	DS01036	1	D-17804	09/02/71 - 09/08/71
		2	D-19241	09/09/71 - 09/15/71
		3	D-17791	09/16/71 - 09/23/71
		4	D-17803	09/23/71 - 09/29/71
		5	D-17802	09/30/71 - 10/07/71
		6	D-17811	10/07/71 - 10/14/71
		7	D-17813	10/14/71 - 10/21/71
		8	D-17814	10/21/71 - 10/28/71
		9	D-19242	10/28/71 - 11/04/71
		10	D-19243	11/04/71 - 11/11/71
DR01037	DS01037	1	D-19244	11/11/71 - 11/17/71
		2	D-19245	11/18/71 - 11/26/71
		3	D-19246	11/25/71 - 12/01/71
		4	D-19247	12/02/71 - 12/09/71
		5	D-19248	12/09/71 - 12/16/71
		6	D-19249	12/16/71 - 12/23/71
		7	D-19250	12/23/71 - 12/30/71

67-0701 030

The Explorer 33 and Explorer 35 individual vector tapes have the following physical characteristics

density: 556 bpi
7-track
parity: odd
physical record length: 2760 characters
BCD characters (essentially)

The first twelve characters of each physical record contain the physical record count and tape number of the original tape. These twelve characters are skipped by the IVREAD routine.

Each logical record is an individual vector. The formats of the two kinds of individual vectors are shown in Figure A-1 and A-2. The first format is associated with the first vector in each sequence. All of the other vectors have the second format. Preceding every logical record is a six character word which contains binary information describing the preceding logical record (if any) and the following logical record. Note that on the DCS control words always start on the whole word boundary and a DCS word has six characters. Thus both vector formats have two blank characters between the end of the vector and the following control word.

We could view the final tape format as having been created in the following manner. All logical records are stored in order, each preceded by its six character control word. Then a group of 2748 characters is picked off, prefixed by a twelve character physical record control word and placed on tape followed by an inter-record gap. The second physical record would contain the second 2748 characters prefixed by twelve control characters and followed by an inter-record gap. The

At the end of the data, the last physical record will contain however many characters remain less than or equal to 2742, suffixed by a six character logical record control word and preceded by a twelve character control word then followed by a single end of file.

char. length
representation
char. count

Sequence Number	Day of Year Jan 1 = 0	Hour of Day	Minutes	Seconds	Vector Position *	B magnitude of vector
6	4	3	3	3	3	6
I6	I4	I3	I3	I3	I3	FG.1
6	10	13	16	19	22	25

char. length
representation
char. count

SOLAR EQUATORIAL				
θ Elevation angle latitude	ϕ Deviation angle longitude	X-Component of vector	Y-Component of vector	Z-Component of vector
6	6	6	6	6
FG.1	FG.1	FG.1	FG.1	FG.1
34	42	46	52	57

char. length
representation
char. count

SOLAR MAGNETOSPHERIC				
θ Elevation angle latitude	ϕ Deviation angle longitude	X-Component of vector	Y-Component of vector	Z-Component of vector
6	6	6	6	6
FG.1	FG.1	FG.1	FG.1	FG.1
64	70	76	82	87

char. length
representation
char. count

SOLAR ECLIPTIC					Total number of vectors in this sequence
θ Elevation angle latitude	ϕ Deviation angle longitude	X-Component of vector	Y-Component of vector	Z-Component of vector	
6	6	6	6	6	6
FG.1	FG.1	FG.1	FG.1	FG.1	I6
94	100	102	112	118	124

θ & ϕ are field vector latitude and longitude, obtained from B_x , B_y , B_z

Figure A-1

First Vector of a Sequence

* indicates 1st, or 2nd, or 3rd etc vector of sequence

char. length
representation
char. count

Blank	Vector Position	E magnitude of vector
19	3	6
19X	13	FG.1
	19	22

char. length
representation
char. count

SOLAR EQUATORIAL				
θ Elevation angle latitude	ϕ Deviation angle longitude	X-Component of vector	Y-Component of vector	Z-Component of vector
6	6	6	6	6
FG.1	FG.1	FG.1	FG.1	FG.1
94	40	46	55	58

char. length
representation
char. count

SOLAR MAGNETOSPHERIC				
θ Elevation angle latitude	ϕ Deviation angle longitude	X-Component of vector	Y-Component of vector	Z-Component of vector
6	6	6	6	6
FG.1	FG.1	FG.1	FG.1	FG.1
64	70	76	82	89

char. length
representation
char. count

SOLAR ECLIPTIC				
θ Elevation angle latitude	ϕ Deviation angle longitude	X-Component of vector	Y-Component of vector	Z-Component of vector
6	6	6	6	6
FG.1	FG.1	FG.1	FG.1	FG.1
94	100	109	112	118

Figure A-2
Other Vector of a Sequence

67-070A-032
PR01016

\$NOP
\$NOF
\$NOP
\$NOP
\$EXE

INPUT PARAMETERS ARE: AS FL=1=1 S 1 1

Tape No. 1

TAPE NO.	1	FILE NO.	1
RECORD	1	LENGTH	2760

[illegible]

TAPE NO. 1

TAPE NO.	FILE NO.
1	1
RECCRC	LENGTH 1841

[illegible]

Tape No. 1

TAPE NO.	FILE NO.
1	2
RECORD 1	LENGTH 2763

3 45.1 300.4 1.4 -2.5 2.9 34.6 295.7 1.4 -3.0 2.3 000000 10 3.9
4.5 284.4 0.7 -2.7 2.7 55.1 285.2 0.7 -2.1 3.2 43.7 284.3 0.7 -2.7 2.7 000000
1.1 -2.9 2.3 000000 1.1 -2.8 2.3 47.7 294.9 1.1 -2.3 2.8 36.8 290.7
9 -2.8 2.5 29.2 286.3 0.9 -3.2 1.9 000000 13 3.5 26.0 301.9 1.6
-2.6 1.5 35.7 305.8 1.6 -2.3 2.2 25.5 301.8 1.6 -2.6 1.5 000000 44.2 245.0 4.28
1 3.6 18.2 306.0 2.0 -2.8 1.1 27.5 309.0 2.0 -2.5 1.7 18.1 305.9 2.0 -2.8 1.1
13 000000 2 3.5 18.3 296.6 1.5 -3.0 1.1 28.6 298.9 1.5 -2.7 1.1
7 18.1 296.6 1.5 -3.0 1.1 000000 3 3.5 29.8 296.1 1.5 -3.0 1.9
4.1 295.9 1.5 -2.6 2.5 29.6 294.1 1.5 -3.0 1.5 000000 4 3.9 25.4
3 296.1 1.5 -3.0 1.9 40.1 299.9 1.5 -2.6 2.5 29.6 296.0 1.5 -3.0 1.9 000000
1 -2.9 1.5 000000 6 3.9 29.8 296.1 1.5 -3.0 1.5 000000 1.1 -2.4 2.4 31.6 291.0 1.1
-2.6 2.5 29.6 296.0 1.5 -3.0 1.5 000000 7 3.5 25.5 296.1 1.5 -3.0 1.5
3 1.9 40.1 299.9 1.5 -2.6 2.5 29.6 296.0 1.5 -3.0 1.9 000000 8
3.9 29.4 286.4 0.9 -3.2 1.9 40.5 288.8 0.9 -2.8 2.5 29.2 286.3 0.9 -3.2 1.9 000
-00 9 4.1 27.8 291.3 1.3 -3.4 1.9 38.6 294.3 1.3 -2.9

TAPE NO. 1 FILE NO. 8
RECORD 3787 LENGTH 968

12-09-000000 9.8 56.0 357.2 6.8 -0.3 10.1 53.2 338.8 6.8 -2.6 9.8 -000000
7 12.0 43.3 330.1 7.5 -4.3 8.2 48.8 343.5 7.5 -2.2 9.0 43.6 330.7 7.5 -4.2
8.2 000000 8 11.5 32.7 326.3 8.1 -5.4 6.2 39.5 335.2 8.1 -3.7 7.7
3 33.0 326.6 8.1 -5.3 6.3 000000 9 11.6 23.6 325.2 8.7 -6.1 4.7
31.0 331.4 8.7 -4.8 6.0 24.0 325.5 8.7 -6.0 4.7 000000 10 11.5 21.5
3 325.3 8.8 -6.1 4.3 29.2 330.9 8.8 -4.9 5.6 22.1 325.5 8.8 -6.0 4.3 000000
3 -5.1 5.9 000000 12 11.6 50.5 338.0 6.9 -2.8 9.0 53.7 355.6 6.9
-0.5 9.4 50.7 338.8 6.9 -2.7 9.0 000000 13 12.0 57.1 337.9 6.1 -2.5
3 10.1 59.8 6.5 6.1 0.0 10.4 57.4 338.9 6.1 -2.3 10.1 000000

TAPE NO. 1 FILE NO. 5
RECORD 1 LENGTH 2763

18-0100000000000000 7.3840 256.0 33.4 1 12.7 41.3 326.3 8.0 -5.3 8.4 47.7 338.4 8.0 -3.1
2 9.4 41.6 326.8 8.0 -5.2 8.5 14 000000 2 12.7 41.5 328.7 8.1
-4.5 8.4 47.4 341.0 8.1 -2.8 8.5 41.8 329.2 8.1 -4.8 8.5 000000
3 12.6 44.0 328.2 7.7 -4.8 8.8 49.9 341.7 7.7 -2.6 9.7 44.4 328.8 7.7 -4.7 8.8
100000 4 12.3 45.2 327.8 7.4 -4.6 8.8 51.2 341.9 7.4 -2.4 9.6 45.4
5 328.4 7.4 -4.5 8.8 000000 5 12.5 42.1 323.4 7.4 -5.5 8.4 49.2 335.6 7.4 -3.4
35.6 7.4 -3.4 9.4 42.5 323.9 7.4 -5.4 8.4 000000 6 12.5 39.6 324.4
7 7.8 -5.7 8.0 46.6 335.2 7.8 -3.6 9.1 40.0 324.5 7.8 -5.6 8.1 000000
7 12.3 40.3 321.1 7.3 -5.9 8.0 48.0 332.3 7.3 -3.8 9.1 40.7 321.5 7.3 -5.4
5.4 42.5 323.9 7.4 -5.4 8.4 000000 9 12.5 42.1 323.4 7.4 -5.5 8.4
4 49.2 335.6 7.4 -3.4 9.4 42.5 323.9 7.4 -5.4 8.4 000000 10 12.8
11.0 324.0 7.8 -5.7 8.4 48.0 335.7 7.8 -3.5 9.5 41.4 324.4
7.8 -5.6 8.5 000000 12 13.0 40.4 329.0 8.5 -5.1 8.4 46.3 340.9 8.5
11 12.8 41.0 324.0 7.8 -5.7 8.4 48.0 335.7 7.8 -3.5 9.5 41.4 324.4
5 -2.9 9.4 40.7 329.5 8.5 -5.0 8.5 000000 13 13.2 41.8 329.0 8.5
-5.1 8.8 47.6 341.4 8.5 -2.8 9.8 42.1 329.5 8.5 -5.0 8.9 000000
13.2 41.5 331.3 8.6 -4.7 8.8 47.2 343.9 8.6 -2.5 9.7 42.3 331.8 8.6 -4.6 8.9
100000 73841 256.0 34.25 1 12.9 42.9 328.6 8.1 -4.9 8.8 48.7 341.6 8.1 -2.7 9.7 43.4
329.2 8.1 -4.8 8.9 13 000000 2 13.2 41.8 329.0 8.5 -5.1 8.8
17.6 341.4 8.5 -2.8 9.8 42.1 329.5 8.5 -5.0 8.9 000000 3 12.9 45.4
328.2 7.7 -4.8 9.2 51.2 342.3 7.7 -2.5 10.1 45.7 328.8 7.7 -4.7 9.2 000000
4 12.9 45.4 328.2 7.7 -4.8 9.2 51.2 342.3 7.7 -2.5 10.1 45.7 328.8 7.7
7 -4.7 9.2 000000 5 12.9 42.9 328.6 8.1 -4.9 8.8 48.7 341.6 8.1 -2.7 9.7 43.4
-2.7 9.7 43.2 329.2 8.1 -4.8 8.9 000000 6 13.2 41.8 329.0 8.5 -5.1 8.8
1 8.8 47.6 341.4 8.5 -2.8 9.8 42.1 329.5 8.5 -5.0 8.9 000000
13.0 46.4 329.0 8.5 -5.1 8.4 46.3 340.9 8.5 -2.9 9.4 40.7 329.5 8.5 -5.0 8.5 000
-00 8 12.9 43.0 331.0 8.2 -4.6 8.8 48.4 344.2 8.2 -2.3

TAPE NO. 1 FILE NO. 9
RECORD 3825 LENGTH 86

18-0990000064.0 224.3 -4.3 -4.2 -12.4 -71.7 183.9 -4.3 -0.3 -13.1 -000000

REQ. AGENT
VJP

RAND NO.
RC6568

ACQ. AGENT
JHK

IMP-E

82 SEC. AVGD. VECTORS, (REBLOCKED)

67-070A-03F

This data set consists of 20 IMP-E data tapes. The tapes are binary, 9 track, 1600 BPI and contains 1 file each. The tapes were created on the IBM/360 computer. These tapes were created from data set 67-070A-03C.

<u>D#</u>	<u>C#</u>	<u>TIME SPAN</u>
D-26560	C-18399	7/19/67 - 9/27/67
D-26561	C-18400	9/28/67 - 12/07/67
D-26562	C-18401	12/07/67 - 2/15/68
D-26563	C-18402	2/15/68-- 4/25/68
D-26564	C-18403	4/25/68 - 7/04/68
D-26565	C-18404	7/04/68 - 9/12/68
D-26566	C-18405	9/12/68 - 11/21/68
D-26567	C-18406	11/21/68 - 1/30/69
D-26568	C-18407	1/30/69 - 4/09/69
D-26569	C-18408	4/10/69 - 6/19/69
D-26570	C-18409	6/19/69 - 8/28/69
D-26571	C-18410	8/28/69 - 11/05/69
D-26572	C-18411	11/09/69 - 1/15/70
D-26573	C-18412	1/15/70 - 3/26/70
D-26574	C-18413	3/26/70 - 6/04/70
D-26575	C-18414	6/04/70 - 8/13/70
D-26576	C-18415	8/13/70 - 10/22/70

<u>D#</u>	<u>C#</u>	<u>TIME SPAN</u>
D-26577	C-18416	10/22/70 - 12/24/70
D-26578	C-18417	7/29/71 - 10/28/71
D-26579	C-18418	10/28/71 - 12/30/71

Imp E

82 Sec Averaged Vectors Reblock

67-070A-03F

This data set has been restored. There were originally 20 9-track, 1600 bpi tapes written in binary on an IBM 360 computer. There is now one restored tape, the DR is on a 3480 cartridge and the DS is on a 9-track tape written in EBCDIC. This data set was created from data set 67-070A-03C. Refer to 67-070A-03C for the documentation. The DR and DS number along with the time spans and corresponding D numbers follow:

DR#	DS#	D#	FILES	TIME SPAN
DR03007	DS03007	D26560	1	07/19/67 - 09/27/67
		D26561	2	09/28/67 - 12/07/67
		D26562	3	12/07/67 - 02/15/68
		D26563	4	02/15/68 - 04/25/68
		D26564	5	04/25/68 - 07/04/68
		D26565	6	07/04/68 - 09/12/68
		D26566	7	09/12/68 - 11/21/68
		D26567	8	11/21/68 - 01/30/69
		D26568	9	01/30/69 - 04/09/69
		D26569	10	04/10/69 - 06/19/69
		D26570	11	06/19/69 - 08/28/69
		D26571	12	08/28/69 - 11/05/69
		D26572	13	11/06/69 - 01/15/70
		D26573	14	01/15/70 - 03/26/70
		D26574	15	03/26/70 - 06/04/70
		D26575	16	06/04/70 - 08/13/70
		D26576	17	08/13/70 - 10/22/70
		D26577	18	10/22/70 - 12/24/70
		D26578	19	07/29/71 - 10/28/71
		D26579	20	10/28/71 - 12/30/71

FORMAT OF REBLOCKED IMP-E

67-070A-03F

The reblocked tape is an unlabelled 9-track, 1600 bpi, binary tape created on an IBM/360 computer. Each logical record consists of 6 control bytes and 132 bytes of data. There are 200 logical records in a physical record. DCB statement to create tape was

DCB=(DEN=3,RECFM=FB,LRECL=2760,BLKSIZE=27600)

FORMAT OF LOGICAL RECORD

<u>BYTES</u>	<u>LENGTH</u>	<u>TYPE</u>	<u>DESCRIPTION</u>	
1- 6	6	A	7094 Control characters	
7-13	7	I	Sequence Number	
14-17	4	I	Day	
18-20	3	I	Hour	
21-23	3	I	Minute	
24-26	3	I	Second	
27-31	5	I	Magnitude	
32-37	6	I	Theta	Solar Equatorial
38-43	6	I	PHI	
44-48	5	I	Y	
49-53	5	I	Z	
54-58	5	I	Dev Y	
59-63	5	I	Dev Z	
64-69	6	I	Theta	Solar Magnetospheric
70-75	6	I	PHI	
76-80	5	I	X	
81-85	5	I	Y	
86-90	5	I	Z	

<u>BYTES</u>	<u>LENGTH</u>	<u>TYPE</u>	<u>DESCRIPTION</u>	
91-95	5	I	Dev X	}
96-100	5	I	Dev Y	
101-105	5	I	Dev Z	
106-111	6	I	Theta	}
112-117	6	I	PHI	
118-122	5	I	Y	
123	5	I	Z	
128-132	5	I	Dev Y	
133-137	5	I	Dev Z	}
138	1	X	Blank	

Solar

Ecliptic

67-676A-07F
DS003007

FILE 1 RECORD 1 27600 BYTES

00000F	14311	199	15	48	13	3431	511	1474	1060	2445	1148	198	461	1396	1659	1410	2262	749	1106	365	538	1536	822	2535	1162
88	00F00F	14318	199	15	57	45	3229	380	1478	1348	1979	119	293	335	1431	2143	1608	1775	108	159	274	407	1517	1153	209
9	91	303	00F00F	14331	199	16	15	29	2596	348	1497	888	1229	1117	474	313	1460	1524	1026	1117	852	1068	576	375	1533
66	1308	1147	396	00F00F	14333	199	16	18	12	2366	386	2019	451	962	1412	946	408	1967	1118	335	1008	673	1437	908	
058	541	914	1383	987	00F00F	14341	199	16	29	7	2459	200	1524	991	779	660	515	171	1506	1898	1068	670	478	657	
225	1543	912	871	660	516	00F00F	14346	199	16	35	56	860	480	1148	373	456	288	193	429	1125	173	415	418	508	
204	528	1177	328	489	294	183	00F00F	14347	199	16	37	18	2277	184	1479	1145	717	45	110	154	1465	1828	1209	604	
55	106	212	1496	1072	823	35	114	00F00F	14348	199	16	38	39	2155	242	1516	795	750	924	277	214	1495	1470	863	
665	901	347	267	1539	720	823	940	219	00F00F	14349	199	16	40	1	941	364	1418	443	528	132	139	330	1390	564	
9	486	238	119	150	396	1452	391	568	145	126	00F00F	14350	199	16	41	23	2211	57	1559	841	208	278	677	36	
86	857	130	252	338	649	77	1565	817	287	212	700	00F00F	14351	199	16	42	45	865	136	1219	462	402	234	244	
201	288	497	358	439	215	261	410	1243	422	444	251	226	00F00F	14354	199	16	46	50	1852	196	1474	850	569	484	
169	1460	1347	906	494	531	461	351	225	1492	802	649	507	279	00F00F	14355	199	16	48	12	793	577	1655	67	427	
182	562	1583	261	103	420	369	460	207	587	1743	26	432	478	161	00F00F	14356	199	16	49	34	834	513	1727	42	
474	256	504	1668	336	78	417	345	454	288	516	1796	2	424	491	220	00F00F	14357	199	16	50	56	1830	190	1509	
596	19	9	167	1496	1512	887	526	15	19	10	216	1527	781	673	20	8	00F00F	14358	199	16	52	18	906	313	
7	354	471	2	5	291	1503	688	391	441	2	2	4	337	1558	308	503	1	5	00F00F	14359	199	16	53	39	
1	1529	353	467	2	5	289	1505	689	389	437	2	3	5	335	1560	306	499	2	5	00F00F	14360	199	16	55	
1	306	1531	350	459	2	5	285	1510	693	384	431	2	2	5	329	1562	305	490	1	5	00F00F	14361	199	16	
23	898	300	1535	347	449	1	4	280	1514	696	379	422	1	2	5	323	1565	302	480	1	5	00F00F	14362	199	
16	57	45	886	299	1543	332	442	30	1	279	1522	692	364	415	17	30	3	321	1574	288	471	30	1	00F00F	
3	199	16	59	7	891	289	1542	186	1525	694	508	11	5	167	1514	1336	727	457	11	11	1570	297	461	0	
14364	199	17	0	28	1588	186	1542	186	1525	694	508	11	5	167	1514	1336	727	457	11	11	1570	297	461	0	
00F00F	14365	199	17	1	50	879	285	1551	324	419	29	3	267	1532	701	353	396	16	28	5	307	1580	282	448	
3	00F00F	14366	199	17	3	12	884	276	1550	331	410	0	0	258	1532	701	353	396	16	28	5	307	1580	282	
9	0	00F00F	14369	199	17	7	17	838	321	1635	186	413	273	45	310	1613	629	213	399	177	270	63	335	1667	
46	428	276	20	00F00F	14370	199	17	8	39	1559	137	1797	5	259	801	640	137	1788	1057	22	258	569	804	636	
810	20	258	794	648	00F00F	14371	199	17	10	1	1425	223	1593	400	466	610	111	211	1580	1060	428	441	387	602	
242	1615	354	502	618	54	00F00F	14372	199	17	11	23	1420	184	1553	560	447	81	82	169	1543	1220	586	412	146	
87	206	1571	514	498	88	75	00F00F	14374	199	17	14	7	285	114	2718	193	39	104	150	148	2719	6	190	50	
112	145	59	2718	196	20	91	159	00F00F	14377	199	17	18	12	865	233	1575	303	342	1	3	221	1563	734	322	
1	3	253	1598	269	370	1	3	00F00F	14378	199	17	19	34	856	230	1584	290	334	27	28	218	1572	732	30	
8	318	18	27	7	249	1607	256	360	1	28	4	00F00F	14379	199	17	20	56	853	226	1587	286	327	28	215	
34	301	313	20	27	10	245	1610	253	353	28	6	00F00F	14380	199	17	22	17	1195	171	1559	466	352	6	160	
550	1042	484	329	11	6	7	193	1575	431	395	7	6	00F00F	14381	199	17	23	39	860	214	1586	292	314	0	
203	1576	745	307	299	0	0	233	1607	260	341	0	0	00F00F	14382	199	17	25	1	858	209	1588	289	307	2	
5	200	1579	747	303	294	2	2	5	229	1609	258	334	1	5	00F00F	14383	199	17	26	23	849	210	1596	276	
29	5	201	1586	742	290	292	18	29	0	0	229	1616	246	330	29	2	00F00F	14384	199	17	27	45	858	207	
303	0	197	1581	749	301	290	0	0	0	0	226	1610	257	330	0	0	00F00F	14385	199	17	29	6	1115	173	
6	438	332	20	13	163	1549	969	453	312	19	19	13	195	1573	405	373	20	11	00F00F	14389	199	17	34	34	
8	1742	96	309	510	244	176	1735	954	109	305	227	506	252	183	1760	66	317	517	229	00F00F	14390	199	17	35	
8	109	1516	362	146	535	330	99	1513	671	367	134	572	528	341	134	1526	347	180	551	303	00F00F	14391	199	17	
17	263	277	1571	65	88	126	85	269	1562	155	68	86	104	125	87	298	1600	56	94	130	79	00F00F	14393	199	
17	40	1	1271	-6	1641	304	-12	196	697	-11	1642	1072	304	-21	361	199	695	9	1641	304	17	198	696	00F00F	
4	199	17	41	23	987	155	1576	361	264	12	23	149	1572	879	369	253	17	12	23	176	1591	335	298	12	
14395	199	17	42	45	851	173	1608	267	253	2	5	167	1603	768	274	245	2	2	5	190	1625	242	278	1	
00F00F	14396	199	17	44	6	287	251	1574	99	122	10	1	245	1566	240	103	119	6	11	1	272	1599	87	131	
1	00F00F	14397	199	17	45	28	851	171	1609	266	250	0	0	166	1605	769	272	244	0	0	0	188	1626	241	
5	0	0	00F00F	14398	199	17	46	50	851	171	1609	266	250	0	0	166	1605	769	272	244	0	0	188	1626	
41	275	0	0	00F00F	14399	199	17	48	12	908	157	1595	305	246	0	0	152	1592	819	311	238	0	0	175	

Appendix B

PROGRAM LISTING


```

C THESE FORATS ARE 19 AVERAGES PLUS 24 QUANTITIES
1000 IF(IFOR.EQ.0) GO TO 2000
      RETOR=0
C TEST TO SEE IF FIRST WORD IS VALID, OR CH 9.10
C IF(RET.EQ.1) GO TO 250
C FIRST WORD IS VALID
      VECTOR(0-1) = ISIN(IAS(VECTOR(J-1))+ARRAY(1),VECTOR(J-1))
      IF(FLAG.EQ.1) VECTOR(J-1) = -ARRAY(1)
      FLAG = 0
C FLAG SETTING OF VECTOR TO BE RETURNED
      INDEX = 25-LAST+INDEX + 1
      DO 1050 I=2,INDEX
        VECTOR(J) = ARRAY(I)
1050 J = J+1
C ADJUST FINAL WORD IN THE ARRAY
1100 CONTINUE
      IPOR = JPO- (IFORR)
      IDPAR = 1
C WORDS ORC REQUIRE TRANSLATION, BRANCH TO 10
      IF((IFORR.GT.1) .AND. IFORR.LT.5) .OR. (IFORR.GT.5 .AND. IFORR.LT.15)
        * .OR. IFORR.EQ.15 .OR. IFORR.GT.17) GO TO 1200
      ICHAR = JCHAR(IFORR)
      CALL ITRN(ARRAY(LAST),ICHR)
1200 IF(ICHR.GT.0) ARRAY(LAST)=ARRAY(LAST)*10**IPOR
      IFLAG = 0
      IF(ICHR.EQ.0) IFLAG = 1
      INDEX = INDEX + 1
      IFORR = IFORR + 1
      IF(IFORR.GT.23) IFORR = 1
      RETOR = RETOR + 1
C
C THIS IS THE CLEARING POINT FOR THE HANDLING OF SPECIAL FORATS
2000 CONTINUE
      IF(IFORR.EQ.12) GO TO 2100
C TEST IS TABLE CORRECT
      VECTOR(J) = (-PI)
      J = 0
      INDEX = 0
      GO TO 250
2100 IF(IFORR.EQ.20) GO TO 2200
C FOR AT EATS IN 46, 41
      RETOR=0
      IF(RET.EQ.1) ARRAY,TEMP2

```

```

C PURSUIT 20 Starts in a double loop
C COUN 112,111,100,90,80,70,60,50,40,30,20,10,0,1,2,18
C 118 I=0 I=25-Last+1+1
C 212 J=1 J=25-Last+1+1
C 214 VECTOR(J)=ARRAY(1)
C 216 J=J+1
C 218 KCOUNT=1
C 220 CALL ITIM(TEMP2,KCOUNT)
C 222 ITEMP=JTEMP(IFTEMP)
C 224 ITEMP=JTEMP(IFTEMP)
C 226 CALL ITIM(ARRAY(LAST),ICOUNT)
C 228 IF(ICOUNT.GT.0) ARRAY(LAST)=ARRAY(LAST)*1000IPTEMP2*10
C 230 IF(ICOUNT.EQ.0) ARRAY(LAST)=-TEMP2
C 232 IF(KCOUNT.EQ.0) IFLAG=1
C 234 GO TO 1300
C 236 IF(IFTEMP.EQ.23) GO TO 2300
C 238 QUANTITIES PLUS 19 AVERAGES
C 240 READ(6,FMT) TEMP1,TEMP2,ARRAY
C 242 VECTOR(J-1)=ISIGN(ABS(VECTOR(J-1))+TEMP1,VECTOR(J-1))
C 244 IF(IFLAG.EQ.1) VECTOR(J-1)=-TEMP1
C 246 VECTOR(J)=TEMP2
C 248 J=J+1
C 250 ITEMP=INDEX-1
C 252 GO TO 250
C 254 CONTINUE
C 256 READ(8,FMT) (ARRAY(I),I=2,Last)
C 258 VECTOR(J-1)=ISIGN(ABS(VECTOR(J-1))+ARRAY(2),VECTOR(J-1))
C 260 IF(IFLAG.EQ.1) VECTOR(J-1)=-ARRAY(2)
C 262 IFLAG=0
C 264 ITEMP=25-Last+INDEX+2
C 266 DO 2320 I=2,INDEX
C 268 VECTOR(J)=ARRAY(I)
C 270 J=J+1
C 272 GO TO 1100
C 274 END
C 276
C 278 IF(SUCCESS) GO TO 280,OPT=02,LI=0,SIZE=00000,
C 280
C 282 SOURCE,CD,ULIST,VECK,LUNG,TEMP,UNIT,IP,TEMP
C 284 SOURCE STATEMENTS = 130 ,PROGRAM SIZE = 4456

```

11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

150 0002  COMPILER OPTIONS - NAME= MAIN,OPT=02,LINENCT=45,SIZE=0000K,
      SOURCE, CD, COLIST, INDEX, L041, L04AP, R00011, 10, INDEX
      SUBROUTINE ITRAP(CD, ICHARS)
      C THIS SUBROUTINE IS FOR USE WITH SAREAD
      C SAREAD READS A BCD BLOCKED BCD TAPE (SEQ. SUMMARY) AND RETURNS ONE
      C AVERAGE FOR EACH CALL
      C THIS ROUTINE TRANSLATES THE PARTIAL PUNCH IN THOSE COLUMNS FOR BITS
      C WHICH READ THE FIRST PART OF A POSSIBLE NEGATIVE NUMBER, THIS
      C ROUTINE IS HIGHLY MACHINE DEPENDENT
      C MASK TESTS BLANKS
      C MASK ISOLATES ONE CHARACTER AT A TIME
      C MASK TEST BCD SIGNS
      C MASK MASKS OUT THE ZONE BITS OF NUMBERS
      C MASK TEST FOR A PERSANDS (BCD PLUS)
      C INTEGER AND
      DIMENSION MASK(4), MASKN(4), MASKP(4), MASKF(4), MASKP(4)
      DATA MASK/Z40000000,Z00400000,Z00004000,Z00000040/
      DATA MASKF/ZFF000000,Z00FF0000,Z0000FF00,Z000000FF/
      DATA MASKN/Z00000000,Z00500000,Z00005000,Z00000050/
      DATA MASKP/Z0F000000,Z00F00000,Z0000F000,Z0000000F/
      DATA MASKZ/Z0F0F0F0F/
      INTEGER MD, NDT
      ICHAR = ICHARS
      J = 1
      NDT = 1
      C CHECK FOR LEADING BLANK
      50 K = AND( MASKF(J), MD)
      IF(K.EQ.MASKP(J) .OR. K.EQ.MASK (J)) GO TO 100
      C CHECK FOR LEADING MINUS
      IF(K.EQ.MASKN(J)) GO TO 150
      GO TO 200
      C
      C LEADING BLANK
      100 IF(ICHAR.EQ.1) GO TO 300
      ICHAR = ICHAR -1
      J = J+1
      GO TO 50
      C
      C LEADING PLUS
      150 NDT = -1
      ICHAR = ICHAR +1
      IF(ICHAR.EQ.6) GO TO 400

```

```

      ITRAP 1
      ITRAP 2
      ITRAP 3
      ITRAP 4
      ITRAP 5
      ITRAP 6
      ITRAP 7
      ITRAP 8
      ITRAP 9
      ITRAP 10
      ITRAP 11
      ITRAP 12
      ITRAP 13
      ITRAP 14
      ITRAP 15
      ITRAP 16
      ITRAP 17
      ITRAP 18
      ITRAP 19
      ITRAP 20
      ITRAP 21
      ITRAP 22
      ITRAP 23
      ITRAP 24
      ITRAP 25
      ITRAP 26
      ITRAP 27
      ITRAP 28
      ITRAP 29
      ITRAP 30
      ITRAP 31
      ITRAP 32
      ITRAP 33
      ITRAP 34
      ITRAP 35
      ITRAP 36
      ITRAP 37
      ITRAP 38
      ITRAP 39
      ITRAP 40
      ITRAP 41

```

```

150 0003
150 0004
150 0005
150 0006
150 0007
150 0008
150 0009
150 0010
150 0011
150 0012
150 0013
150 0014
150 0015
150 0016
150 0018
150 0020
150 0021
150 0023
150 0024
150 0025
150 0026
150 0027
150 0028

```

```

IS# 0030      C LEADING ZEROS
200  C CHITUNE
C MASK OUT THE ZONES
    MD = ADD(ED,MASKZ)
C RIGHT ADJUST THE WORD
    MD = ED/(2**(8*(4-ICHARS)))
    MD = ADD(MASK(1),MD)/2**24*10**3 + ADD(MASK(2), MD)/2**15*10**2
    * ADD(MASK(3),MD)/2**8*10 + ADD(MASK(4),MD)
    MD = MD *EDT
    ICHARS = 1
    RETURN

C BLACK WORD
300  MD=0
    ICHARS = 1
    RETURN

C THE NUMBER HAS ONLY A MINUS SIGN
400  MD = EDT
    ICHARS = 0
    RETURN
END

IS# 0037
IS# 0038
IS# 0039

IS# 0040
IS# 0041
IS# 0042
IS# 0043

```

OPTIONS IN EFFECT* NAME= MAIN,OPT=02,LINECIT=45,SIZE=00000,

OPTIONS IN EFFECT* SOURCE,BCD,NOLIST,NODECK,LOAD,NORAP,NODEIT,ID,POXREF

STATISTICS* SOURCE STATEMENTS = 42 ,PROGRAM SIZE = 882

STATISTICS* NO DIAGNOSTICS GENERATED

***** END OF COMPILE *****

37K BYTES OF CORE NOT USED

COMPILER OPTIONS - NAME= DATA, OPT=02, LIPFCT=45, SIZE=0000K,

SOURCE, CO, LIST, CHECK, LOAD, PUNCH, PUNCHIT, ID, DEXEFF

ISL 0002

BLOCK DATA

C THIS BLOCK DATA IS FOR USE WITH THE PACKAGE WHICH IS USED TO READ

TC SEQUENCES ON TAPE (LOCKED DCS) OF THE IGA/360

C THE FORMATS ARE INITIALIZED IN DOUBLE PRECISION TO MINIMIZE THE USE OF SPACE

ISL 0003

IMPLICIT REAL*8(F)

ISL 0004

COMMON/SECSUB/FMT1(13),FMT2(13),FMT3(13),FMT4(13),FMT5(13),FMT6(13),
FMT7(13),FMT8(13),FMT9(13),FMT10(13),FMT11(13),FMT12(13),FMT13(13),
FMT14(13),FMT15(13),FMT16(13),FMT17(13),FMT18(13),FMT19(13),
FMT20(13),FMT21(13),FMT22(13),FMT23(13)

ISL 0005

DATA FMT1/

'(12X,19(1,6X,17,14,1,313,15,1,216,415,1,216,615,1,216,415,1,1,
'1X),6X,1,1,7,14,313,1,15,216,1,415,216,1,615,216,1,15,14,1,
')',/

ISL 0006

DATA FMT2/

'(12X,11,1,215,1X,1,1,9(6X,17,1,14,313,1,15,216,41,1,5,216,61,1,
'5,216,41,1,5,1X),6X,1,17,14,3,13,15,21,1,6,415,21,1,6,615,16,1,
'1,13)',/

ISL 0007

DATA FMT3/

'(12X,13,1,415,1X,1,1,9(6X,17,1,14,313,1,15,216,4,1,15,216,6,1,
'15,216,4,1,15,1X),6,1X,17,14,1,313,15,2,1,6,415,2,1,6,515,1,1,
'12)',/

ISL 0008

DATA FMT4/

'(12X,13,1,216,415,1,1X,19(6X,1,17,14,3,13,15,21,1,6,415,21,1,
'6,615,21,1,6,415,1X,1,1,6X,17,1,14,313,1,1,5,216,41,1,5,216,31,1,
'15)',/

ISL 0009

DATA FMT5/

'(12X,315,1,216,415,1,1X,19(6,1X,17,14,1,313,15,2,1,6,415,2,1,
'16,415,2,1,16,415,1,1,1X),6X,17,1,14,313,1,1,14,313,1,15,216,4,1,
'15,216,4,1,3)',/

ISL 0010

DATA FMT6/

'(12X,12,1,515,216,1,415,1X,1,1,9(6X,17,1,14,313,1,1,5,216,41,1,
'15,216,61,1,5,216,41,1,5,1X),6X,1,17,14,3,13,15,21,1,6,415,43,1,
')',/

ISL 0011

DATA FMT7/

'(12X,13,1,16,615,2,1,16,415,1,1X,19(6X,1,17,14,31,1,3,15,216,1,
'1,415,216,1,615,216,1,415,1X),1,6X,17,1,1,6,613,15,1,216,216,1,
'1,1)',/

ISL 0012

DATA FMT8/

'(12X,14,1,15,216,6,1,15,216,4,1,19,1X,19,1,6X,17,1,1,4,313,18,1,
'1,216,415,1,216,615,1,216,415,1,1X),6X,1,17,14,31,1,3,15,16,1,
'15)',/

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37

