[redacted]

#449

OSO-8

MERGED X-RAY DATA

75-057A-03A

DSC#449

OSO 8

MERGED X-RAY DATA

75-057A-03A

This data set has been restored. There was originally 270 Binary 9-Track, 1600 BPI tapes. There are 92 restored tapes. The DR and DS tapes are 9-track, 6250 BPI. The tapes were created on an IBM 360 computer. The DR and DS numbers along with the corresponding D numbers and the time spans are as follows:

DR#	DS#	D#	FILES	TIME SPAN
DR01081	DS01081	D31774	1-5	06/30/75 - 07/03/75
		D31775	6-9	07/03/75 - 07/06/75
		D31833	10-12	07/09/75 - 07/12/75
		D31776	13-15	07/15/75 - 07/19/75
DR01082	DS01082	D31777	1-3	07/19/75 - 07/21/75
		D31778	4-6	07/21/75 - 07/24/75
		D31192	7-9	07/24/75 - 07/27/75
		D31193	10-12	07/27/75 - 07/29/75
DR01083	DS01083	D31836	1-3	07/30/75 - 08/02/75
		D32019	4-6	08/01/75 - 08/03/75
		D31251	7-9	08/05/75 - 08/08/75
		D32017	10-12	08/07/75 - 08/10/75
DR01084	DS01084	D31194	1	08/11/75 - 08/12/75
		D31195	2-4	08/12/75 - 08/15/75
		D31252	5-6	08/15/75 - 08/16/75
		D31196	7-10	08/17/75 - 08/20/75
DR01085	DS01085	D31197	1-3	08/20/75 - 08/23/75
		D31198	4-6	08/24/75 - 08/26/75
		D31199	7-9	08/26/75 - 08/29/75
		D31253	10-12	08/29/75 - 09/01/75
DR01086	DS01086	D31779	1-3	09/01/75 - 09/04/75
		D31254	4-6	09/04/75 - 09/06/75
		D31780	7-9	09/07/75 - 09/10/75
		D31255	10-12	09/10/75 - 09/13/75
DR01087	DS01087	D31834	1-3	09/13/75 - 09/16/75
		D31256	4-6	09/16/75 - 09/19/75
		D31257	7-9	09/19/75 - 09/22/75
		D31258	10-12	09/28/75 - 10/01/75
DR01088	DS01088	D31200	1-3	10/01/75 - 10/04/75
		D31201	4-6	10/04/75 - 10/07/75
		D31804	7-9	10/07/75 - 10/10/75
		D31259	10-12	10/11/75 - 10/13/75

DR#	DS#	D#	FILES	TIME SPAN
DR01089	DS01089	D31260	1-3	10/13/75 - 10/16/75
		D31202	4-6	10/19/75 - 10/21/75
		D31203	7-9	10/22/75 - 10/25/75
		D31204	10-12	10/25/75 - 10/28/75
DR01090	DS01090	D31205	1-3	10/28/75 - 10/30/75
		D31261	4-6	10/31/75 - 11/03/75
		D31262	7-9	11/03/75 - 11/06/75
		D31781	10-12	11/06/75 - 11/09/75
DR01091	DS01091	D31782	1-3	11/09/75 - 11/12/75
		D31783	4-6	11/12/75 - 11/15/75
		D31784	7-9	11/15/75 - 11/18/75
		D31805	10-12	11/18/75 - 11/21/75
DR01092	DS01092	D31806	1-3	11/21/75 - 11/24/75
		D31807	4-6	11/24/75 - 11/27/75
		D31785	7-9	11/27/75 - 11/30/75
		D31835	10-12	11/30/75 - 12/03/75
DR01093	DS01093	D31786	1-3	12/03/75 - 12/06/75
		D31787	4-6	12/06/75 - 12/09/75
		D31788	7-9	12/09/75 - 12/12/75
		D31789	10-12	12/12/75 - 12/15/75
DR01094	DS01094	D32020	1-3	12/14/75 - 12/17/75
		D31808	4-6	12/18/75 - 12/21/75
		D31809	7-9	12/21/75 - 12/24/75
		D31810	10-12	12/24/75 - 12/27/75
DR01095	DS01095	D31811	1-3	12/27/75 - 12/30/75
		D31790	4	12/30/75 - 12/31/75
		D32016	5-7	01/01/76 - 01/03/76
		D32013	8-10	01/06/76 - 01/09/76
DR01096	DS01096	D31998	1-3	01/10/76 - 01/12/76
		D32270	4-6	01/12/76 - 01/15/76
		D32000	7-9	01/15/76 - 01/18/76
		D31997	10-12	01/18/76 - 01/21/76
DR01097	DS01097	D32010	1-3	01/21/76 - 01/24/76
		D31999	4-6	01/24/76 - 01/27/76
		D32009	7-9	01/27/76 - 01/30/76
		D32018	10-12	01/31/76 - 02/02/76
DR01098	DS01098	D31996	1-3	02/03/76 - 02/05/76
		D32022	4-6	02/05/76 - 02/08/76
		D32012	7-9	02/08/76 - 02/11/76
		D32015	10-12	02/11/76 - 02/14/76
DR01099	DS01099	D32014	1-3	02/14/76 - 02/17/76
		D32021	4-6	02/17/76 - 02/20/76
		D32001	7-9	02/20/76 - 02/23/76
		D32271	10-12	02/24/76 - 02/26/76
DR01100	DS01100	D32006	1-3	02/26/76 - 02/29/76
		D32011	4-6	03/01/76 - 03/03/76
		D32003	7-9	03/03/76 - 03/07/76
		D32005	10-12	03/07/76 - 03/09/76

DR#	DS#	D#	FILES	TIME SPAN
DR01101	DS01101	D32007	1-3	03/09/76 - 03/11/76
		D32002	4-6	03/14/76 - 03/16/76
		D32008	7-9	03/15/76 - 03/18/76
		D32004	10-12	03/18/76 - 03/21/76
DR01102	DS01102	D32703	1-3	03/30/76 - 04/01/76
		D32704	4-6	04/02/76 - 04/05/76
		D32705	7-9	04/05/76 - 04/08/76
		D32706	10-12	04/08/76 - 04/11/76
DR01103	DS01103	D32707	1-3	04/11/76 - 04/14/76
		D32708	4-6	04/14/76 - 04/17/76
		D32729	7-9	04/17/76 - 04/20/76
		D32709	10-12	04/20/76 - 04/23/76
DR01104	DS01104	D32710	1-3	04/23/76 - 04/26/76
		D32711	4-6	04/26/76 - 04/29/76
		D32712	7-9	04/29/76 - 05/02/76
		D32713	10-12	05/03/76 - 05/05/76
DR01105	DS01105	D32714	1-3	05/05/76 - 05/08/76
		D32715	4-6	05/08/76 - 05/11/76
		D32716	7-9	05/11/76 - 05/14/76
		D32717	10-12	05/14/76 - 05/17/76
DR01106	DS01106	D32718	1-3	05/17/76 - 05/20/76
		D32730	4-6	05/20/76 - 05/23/76
		D32719	7-9	05/23/76 - 05/26/76
		D32720	10-12	06/04/76 - 06/07/76
DR01107	DS01107	D32721	1-3	06/07/76 - 06/10/76
		D32722	4-6	06/10/76 - 06/13/76
		D32723	7-9	06/13/76 - 06/16/76
		D32724	10-12	06/16/76 - 06/19/76
DR01108	DS01108	D32725	1-3	06/19/76 - 06/22/76
		D32726	4-6	06/22/76 - 06/24/76
		D32727	7-9	06/25/76 - 06/28/76
		D32728	10-12	06/28/76 - 07/01/76
DR01109	DS01109	D32756	1-3	07/01/76 - 07/04/76
		D32757	4-6	07/04/76 - 07/07/76
		D32758	7-9	07/07/76 - 07/10/76
		D32759	10-12	07/10/76 - 07/13/76
DR01110	DS01110	D32760	1-3	07/13/76 - 07/16/76
		D32761	4-6	07/16/76 - 07/19/76
		D32762	7-9	07/19/76 - 07/22/76
		D32763	10-12	07/22/76 - 07/25/76
DR01111	DS01111	D32764	1-3	07/25/76 - 07/28/76
		D32765	4-6	07/28/76 - 07/31/76
		D32766	7-9	07/31/76 - 08/03/76
		D32767	10-12	08/03/76 - 08/06/76
DR01112	DS01112	D32768	1-3	08/06/76 - 08/09/76
		D32769	4-6	08/09/76 - 08/12/76
		D32770	7-9	08/12/76 - 08/15/76
		D32771	10-12	08/15/76 - 08/18/76

DR#	DS#	D#	FILES	TIME SPAN
DR01496	DS01496	D32772	1-3	08/18/76 - 08/21/76
		D32773	4-6	08/21/76 - 08/24/76
		D32774	7-9	08/25/76 - 08/27/76
		D32731	10-12	08/27/76 - 08/30/76
DR01497	DS01497	D32732	1-3	08/30/76 - 09/02/76
		D32733	4-6	09/02/76 - 09/05/76
		D32734	7-9	09/05/76 - 09/08/76
		D32735	10-12	09/09/76 - 09/11/76
DR01498	DS01498	D32736	1-3	09/11/76 - 09/14/76
		D32737	4-6	09/14/76 - 09/17/76
		D32738	7-9	09/17/76 - 09/21/76
		D32739	10-12	09/21/76 - 09/23/76
DR01499	DS01499	D32740	1-3	09/23/76 - 09/26/76
		D32741	4-6	09/26/76 - 09/29/76
		D32742	7-9	09/29/76 - 10/02/76
		D32743	10-12	10/01/76 - 01/05/76
DR01500	DS01500	D32744	1-3	10/05/76 - 10/08/76
		D32775	4-6	10/08/76 - 10/11/76
		D32745	7-9	10/11/76 - 10/14/76
		D32746	10-12	10/14/76 - 10/17/76
DR01501	DS01501	D32747	1-3	10/17/76 - 10/20/76
		D32748	4-6	10/20/76 - 10/23/76
		D32749	7-9	10/23/76 - 10/26/76
		D32750	10-12	10/26/76 - 10/29/76
DR01502	DS01502	D33321	1-3	10/30/76 - 11/01/76
		D33322	4-6	11/01/76 - 11/04/76
		D33323	7-9	11/04/76 - 11/07/76
		D33324	10-12	11/07/76 - 11/10/76
DR01503	DS01503	D33325	1-3	11/10/76 - 11/13/76
		D33326	4-6	11/13/76 - 11/16/76
		D33327	7-9	11/16/76 - 11/19/76
		D33328	10-12	11/19/76 - 11/22/76
DR01504	DS01504	D33329	1-3	11/22/76 - 11/25/76
		D33330	4-6	11/25/76 - 11/28/76
		D33331	7-9	11/28/76 - 12/01/76
		D33332	10-12	12/01/76 - 12/04/76
DR01505	DS01505	D33333	1-3	12/04/76 - 12/07/76
		D33334	4-6	12/07/76 - 12/10/76
		D33335	7-9	12/10/76 - 12/13/76
		D33336	10-12	12/13/76 - 12/16/76
DR01506	DS01506	D33337	1-3	12/16/76 - 12/19/76
		D33338	4-6	12/19/76 - 12/22/76
		D33339	7-9	12/22/76 - 12/25/76
		D33340	10-12	12/25/76 - 12/28/76
DR01507	DS01507	D33341	1-3	12/28/76 - 12/31/76
		D33343	4-6	01/01/77 - 01/03/77
		D33343	7-9	01/03/77 - 01/06/77
		D33344	10-12	01/06/77 - 01/09/77

DR#	DS#	D#	FILES	TIME SPAN
DR01508	DS01508	D33345	1-3	01/09/77 - 01/12/77
		D33346	4-6	01/12/77 - 01/15/77
		D33347	7-9	01/15/77 - 01/18/77
		D33348	10-12	01/18/77 - 01/21/77
DR01509	DS01509	D33461	1-3	01/21/77 - 01/24/77
		D33438	4-6	01/24/77 - 01/27/77
		D33349	7-9	01/27/77 - 01/30/77
		D33350	10-12	01/30/77 - 02/02/77
DR01510	DS01510	D33437	1-3	02/02/77 - 02/05/77
		D33351	4-6	02/05/77 - 02/08/77
		D33459	7-9	02/09/77 - 02/11/77
		D33460	10-12	02/11/77 - 02/14/77
DR01511	DS01511	D33462	1-3	02/15/77 - 02/17/77
		D33463	4-6	02/17/77 - 02/20/77
		D33464	7-9	02/20/77 - 02/23/77
		D33465	10-12	02/23/77 - 02/26/77
DR01512	DS01512	D33447	1-3	02/28/77 - 03/04/77
		D33448	4-6	03/04/77 - 03/07/77
		D33439	7-9	03/07/77 - 03/10/77
		D33454	10-12	03/10/77 - 03/13/77
DR01513	DS01513	D33449	1-3	03/13/77 - 03/16/77
		D33450	4-6	03/16/77 - 03/19/77
		D33451	7-9	03/19/77 - 03/22/77
		D33452	10-12	03/22/77 - 03/28/77
DR01514	DS01514	D33453	1-3	03/26/77 - 03/28/77
		D33457	4-6	03/28/77 - 03/31/77
		D33458	7-9	03/31/77 - 04/03/77
		D33456	10-12	04/04/77 - 04/06/77
DR01515	DS01515	D33455	1-3	04/06/77 - 04/09/77
		D33441	4-6	04/09/77 - 04/12/77
		D33440	7-9	04/12/77 - 04/15/77
		D33445	10-12	04/15/77 - 04/18/77
DR01516	DS01516	D33446	1-3	04/18/77 - 04/30/77
		D33442	4-6	04/18/77 - 04/21/77
		D33443	7-9	04/21/77 - 04/24/77
		D33444	10-12	04/24/77 - 04/27/77
DR01517	DS01517	D33561	1-3	04/30/77 - 05/03/77
		D33562	4-6	05/03/77 - 05/06/77
		D33563	7-9	05/06/77 - 05/09/77
		D33564	10-12	05/09/77 - 05/12/77
DR01518	DS01518	D33565	1-3	05/12/77 - 05/15/77
		D33566	4-6	05/15/77 - 05/18/77
		D33578	7-9	05/19/77 - 05/21/77
		D33569	10-12	05/21/77 - 05/24/77
DR01519	DS01519	D33567	1-3	05/24/77 - 05/27/77
		D33568	4-6	05/27/77 - 05/30/77
		D33570	7-9	05/31/77 - 06/02/77
		D33572	10-12	06/02/77 - 06/05/77

DR#	DS#	D#	FILES	TIME SPAN
DR01520	DS01520	D33573	1-3	06/05/77 - 06/08/77
		D33574	4-6	06/09/77 - 06/11/77
		D33571	7-9	06/11/77 - 06/14/77
		D33575	10-12	06/14/77 - 06/17/77
DR01521	DS01521	D33576	1-3	06/17/77 - 06/20/77
		D33577	4-6	06/21/77 - 06/23/77
		D33579	7-9	06/23/77 - 06/26/77
		D33583	10-12	06/26/77 - 06/29/77
DR01522	DS01522	D33580	1-3	06/29/77 - 07/02/77
		D33584	4-6	07/03/77 - 07/05/77
		D33581	7-9	07/05/77 - 07/08/77
		D33586	10-12	07/08/77 - 07/11/77
DR01523	DS01523	D33582	1-3	07/11/77 - 07/14/77
		D33585	4-6	07/14/77 - 07/17/77
		D33631	7-9	07/17/77 - 07/20/77
		D33591	10-12	07/20/77 - 07/23/77
DR01524	DS01524	D33589	1-3	07/23/77 - 07/26/77
		D33587	4	07/27/77 - 07/29/77
		D33592	5-7	07/29/77 - 08/01/77
		D33593	8-9	08/02/77 - 08/04/77
DR01525	DS01525	D33588	1-3	08/05/77 - 08/07/77
		D33621	4-6	08/07/77 - 08/10/77
		D33590	7-9	08/11/77 - 08/13/77
		D33622	10-12	08/13/77 - 08/16/77
DR01526	DS01526	D33623	1-3	08/16/77 - 08/09/77
		D33632	4-6	08/19/77 - 08/22/77
		D33633	7-9	08/22/77 - 08/25/77
		D33639	10-12	08/25/77 - 08/28/77
DR01527	DS01527	D33640	1-3	08/28/77 - 08/31/77
		D33624	4-6	08/31/77 - 09/03/77
		D33625	7-8	09/03/77 - 09/05/77
		D33626	9-11	09/06/77 - 09/09/77
DR01528	DS01528	D33627	1-3	09/10/77 - 09/13/77
		D33638	4-6	09/12/77 - 09/15/77
		D33628	7-9	09/13/77 - 09/15/77
		D33645	10-12	09/15/77 - 09/18/77
DR01539	DS01539	D33641	1-3	09/18/77 - 09/20/77
		D33642	4-6	09/22/77 - 09/24/77
		D33629	7-9	09/25/77 - 09/27/77
		D33630	10-12	09/28/77 - 09/30/77
DR01540	DS01540	D33634	1-3	09/30/77 - 10/03/77
		D33635	4-6	10/04/77 - 10/06/77
		D33636	7-9	10/07/77 - 10/09/77
		D33637	10-12	10/09/77 - 10/12/77
DR01541	DS01541	D33648	1-3	10/15/77 - 10/18/77
		D33658	4-6	10/17/77 - 10/20/77
		D33646	7-9	10/22/77 - 10/23/77
		D33649	10-12	10/24/77 - 10/27/77

DR#	DS#	D#	FILES	TIME SPAN
DR01542	DS01542	D33650	1-3	10/27/77 - 10/30/77
		D33644	4-6	11/04/77 - 11/05/77
		D33651	7-9	11/05/77 - 11/08/77
		D33655	10-12	11/08/77 - 11/10/77
DR01543	DS01543	D33643	1-3	11/11/77 - 11/13/77
		D33652	4-6	11/14/77 - 11/17/77
		D33657	7-9	11/16/77 - 11/19/77
		D33647	10-12	11/20/77 - 11/23/77
DR01544	DS01544	D33653	1-3	11/23/77 - 11/25/77
		D33656	4-6	11/25/77 - 11/28/77
		D33752	7-9	11/28/77 - 12/03/77
		D33659	10-12	11/28/77 - 12/01/77
DR01545	DS01545	D33654	1-3	12/04/77 - 12/06/77
		D33753	4-6	12/09/77 - 12/12/77
		D33756	7-9	12/14/77 - 12/17/77
		D33754	10-12	12/16/77 - 12/17/77
DR01546	DS01546	D33755	1-3	12/21/77 - 12/23/77
		D33758	4-6	12/23/77 - 12/26/77
		D33757	7-9	12/26/77 - 12/31/77
		D33759	10-11	12/26/77 - 12/28/77
DR01547	DS01547	D33774	1-3	01/01/78 - 01/03/78
		D33767	4-5	01/04/78 - 01/06/78
		D33775	6-8	01/06/78 - 01/09/78
		D33768	9-11	01/09/78 - 01/12/78
DR01548	DS01548	D33780	1-3	01/13/78 - 01/15/78
		D33778	4-6	01/15/78 - 01/18/78
		D33781	7-9	01/18/78 - 01/21/78
		D33789	10-12	01/21/78 - 01/24/78
DR01551	DS01551	D33762	1-3	01/24/78 - 01/27/78
		D33761	4-6	01/27/78 - 01/30/78
		D33769	7-9	01/30/78 - 02/02/78
		D33782	10-11	02/02/78 - 02/04/78
DR01552	DS01552	D33760	1-3	02/05/78 - 02/08/78
		D33765	4-6	02/08/78 - 02/11/78
		D33763	7-9	02/11/78 - 02/14/78
		D33766	10-12	02/14/78 - 02/18/78
DR01553	DS01553	D33790	1-3	02/18/78 - 02/20/78
		D33791	4-6	02/20/78 - 02/22/78
		D33792	7-9	02/23/78 - 02/26/78
		D33770	10-12	02/27/78 - 03/01/78
DR01554	DS01554	D33764	1-3	03/01/78 - 03/04/78
		D33771	4-6	03/04/78 - 03/06/78
		D33772	7-9	03/07/78 - 03/10/78
		D33776	10-12	03/11/78 - 03/13/78
DR01555	DS01555	D33773	1-3	03/13/78 - 03/16/78
		D33783	4-6	03/16/78 - 03/19/78
		D33784	7-9	03/20/78 - 03/22/78
		D33793	10-11	03/22/78 - 03/25/78

DR#	DS#	D#	FILES	TIME SPAN
DR01556	DS01556	D33777	1-3	03/26/78 - 04/02/78
		D33785	4-6	03/28/78 - 03/31/78
		D33822	7-9	03/31/78 - 04/03/78
		D33823	10-12	04/04/78 - 04/06/78
DR01557	DS01557	D33786	1-3	04/07/78 - 04/08/78
		D33787	4-6	04/09/78 - 04/12/78
		D33829	7-9	04/12/78 - 04/15/78
		D33830	10-12	04/15/78 - 04/18/78
DR01558	DS01558	D33824	1-3	04/18/78 - 04/21/78
		D33825	4-6	04/21/78 - 04/24/78
		D33826	7-9	04/25/78 - 04/27/78
		D33788	10-11	04/28/78 - 04/30/78
DR01559	DS01559	D33779	1-3	05/01/78 - 05/03/78
		D33831	4-6	05/04/78 - 05/06/78
		D33834	7-9	05/06/78 - 05/09/78
		D33827	10-12	05/15/78 - 05/18/78
DR01563	DS01563	D33832	1-3	05/18/78 - 05/21/78
		D33828	4-6	05/24/78 - 05/27/78
		D33833	7-9	05/27/78 - 05/30/78
		D33843	10-12	06/02/78 - 06/05/78
DR01564	DS01564	D33836	1-3	06/05/78 - 06/08/78
		D33844	4-6	06/08/78 - 06/11/78
		D33837	7-9	06/12/78 - 06/14/78
		D33838	10-12	06/15/78 - 06/17/78
DR01565	DS01565	D33839	1-3	06/18/78 - 06/20/78
		D33850	4-6	06/26/78 - 06/29/78
		D33855	7-9	06/30/78 - 07/02/78
		D33845	10-12	07/02/78 - 07/05/78
DR01566	DS01566	D33840	1-3	07/05/78 - 07/08/78
		D33835	4-6	07/08/78 - 07/11/78
		D33841	7-9	07/12/78 - 07/14/78
		D33842	10-12	07/14/78 - 07/17/78
DR01567	DS01567	D33846	1-3	07/17/78 - 07/20/78
		D33847	4-6	07/21/78 - 07/23/78
		D33848	7-9	07/23/78 - 07/26/78
		D33859	10	07/27/78 - 07/29/78
DR01568	DS01568	D33849	1-3	08/04/78 - 08/07/78
		D33851	4-6	08/07/78 - 08/10/78
		D33852	7-9	08/10/78 - 08/13/78
		D33856	10-12	08/14/78 - 08/15/78
DR01569	DS01569	D33857	1-3	08/17/78 - 08/19/78
		D33858	4-6	08/19/78 - 08/22/78
		D33853	7-9	08/22/78 - 08/25/78
		D33854	10-12	08/25/78 - 08/28/78
DR01570	DS01570	D33860	1-3	08/31/78 - 09/03/78
		D33861	4-5	09/06/78 - 09/08/78
		D33862	6-8	09/13/78 - 09/15/78

REQ. AGENT	RAND NO.	ACQ. AGENT
CMP	RD0048	RWP

OSO-8

MERGED X-RAY DATA

75-057A-03A

This data set catalog contains 370 Magnetic tapes. They are 9 track, Binary, 1600 BPI with multiple files. These tapes were created on an IBM 360 computer. The tapes have 3 files of data on each tape unless it is otherwise noted in the following listing. The dates are not necessarily in sequential order on the tape but the dates do reflect the total time span covered by that tape.

D#	C#	# of FILES	TIME SPAN
D-31776	C-19911		07/15/75 - 07/19/75
D-31777	C-19912		07/19/75 - 07/21/75
D-31778	C-19913		07/21/75 - 07/24/75
D-31779	C-19914		09/01/75 - 09/04/75
D-31780	C-19915		09/07/75 - 09/10/75
D-31781	C-19916		11/06/75 - 11/09/75
D-31782	C-19917		11/09/75 - 11/12/75
D-31783	C-19918		11/12/75 - 11/15/75
D-31784	C-19919		11/15/75 - 11/18/75
D-31785	C-19920		11/27/75 - 11/30/75
D-31786	C-19921		12/03/75 - 12/06/75
D-31787	C-19922		12/06/75 - 12/09/75
D-31788	C-19923		12/09/75 - 12/12/75
D-31789	C-19924		12/12/75 - 12/15/75
D-31790	C-19925	1	12/30/75 - 12/31/75
D-31804	C-19940		10/07/75 - 10/10/75
D-31805	C-19941		11/18/75 - 11/21/75
D-31806	C-19942		11/21/75 - 11/24/75
D-31807	C-19943		11/24/75 - 11/27/75
D-31808	C-19944		12/18/75 - 12/21/75
D-31809	C-19945		12/21/75 - 12/24/75
D-31810	C-19946		12/24/75 - 12/27/75
D-31811	C-19947		12/27/75 - 12/30/75
D-31833	C-19949		07/09/75 - 07/12/75
D-31834	C-19950		09/13/75 - 09/16/75
D-31835	C-19951		11/30/75 - 12/03/75
D-31836	C-19952		07/30/75 - 08/02/75
D-31996	C-20155		02/03/76 - 02/05/76
D-31997	C-20156		[illegible]

56

D#	C#	# of FILES	TIME SPAN
D-31192	C-19849		07/24/75 - 07/27/75
D-31193	C-19850		07/27/75 - 07/29/75
D-31194	C-19851	1	08/11/75 - 08/12/75
D-31195	C-19852		08/12/75 - 08/15/75
D-31196	C-19853	4	08/17/75 - 08/20/75
D-31197	C-19854		08/20/75 - 08/23/75
D-31198	C-19855		08/24/75 - 08/26/75
D-31199	C-19856		08/26/75 - 08/29/75
D-31200	C-19857		10/01/75 - 10/04/75
D-31201	C-19858		10/04/75 - 10/07/75
D-31202	C-19859		10/19/75 - 10/21/75
D-31203	C-19860		10/22/75 - 10/25/75
D-31204	C-19861		10/25/75 - 10/28/75
D-31205	C-19862		10/28/75 - 10/30/75
D-31251	C-19882		08/05/75 - 08/08/75
D-31252	C-19883	2	08/15/75 - 08/16/75
D-31253	C-19884		08/29/75 - 09/01/75
D-31254	C-19885		09/04/75 - 09/06/75
D-31255	C-19886		09/10/75 - 09/13/75
D-31256	C-19887		09/16/75 - 09/19/75
D-31257	C-19888		09/19/75 - 09/22/75
D-31258	C-19889		09/28/75 - 10/01/75
D-31259	C-19890		10/11/75 - 10/13/75
D-31260	C-19891		10/13/75 - 10/16/75
D-31261	C-19892		10/31/75 - 11/03/75
D-31262	C-19893		11/03/75 - 11/06/75
D-31774	C-19909	5	06/30/75 - 07/03/75
D-31775	C-19910	4	07/03/75 - 07/06/75

D#	C#	# of FILES	TIME SPAN
D-31998	C-20157		01/10/76 - 01/12/76
D-31999	C-20158		01/24/76 - 01/27/76
D-32000	C-20159		01/15/76 - 01/18/76
D-32001	C-20160		02/20/76 - 02/23/76
D-32002	C-20161		03/14/76 - 03/16/76
D-32003	C-20162		03/03/76 - 03/07/76
D-32004	C-20163		03/18/76 - 03/21/76
D-32005	C-20164		03/07/76 - 03/09/76
D-32006	C-20165		02/26/76 - 02/29/76
D-32007	C-20166		03/09/76 - 03/11/76
D-32008	C-20167		03/15/76 - 03/18/76
D-32009	C-20168		01/27/76 - 01/30/76
D-32010	C-20169		01/21/76 - 01/24/76
D-32011	C-20170		03/01/76 - 03/03/76
D-32012	C-20171		02/08/76 - 02/11/76
D-32013	C-20172		01/06/76 - 01/09/76
D-32014	C-20173		02/14/76 - 02/17/76
D-32015	C-20174		02/11/76 - 02/14/76
D-32016	C-20175		01/01/76 - 01/03/76
D-32017	C-20176		08/07/75 - 08/10/75
D-32018	C-20177		01/31/76 - 02/02/76
D-32019	C-20178		08/01/75 - 08/03/75
D-32020	C-20179		12/14/75 - 12/17/75
D-32021	C-20180		02/17/76 - 02/20/76
D-32022	C-20181		02/05/76 - 02/08/76
D-32270	C-20135		01/12/76 - 01/15/76
D-32271	C-20136		02/24/76 - 02/26/76
D-32703			03/30/76 - 04/01/76
D-32704			[illegible]

D#	C#	# of FILES	TIME SPAN
D-32705			04/05/76 - 04/08/76
D-32706			04/08/76 - 04/11/76
D-32707			04/11/76 - 07/14/76
D-32708			04/14/76 - 04/17/76
D-32709			04/20/76 - 04/23/76
D-32710			04/23/76 - 04/26/76
D-32711			04/26/76 - 04/29/76
D-32712			04/29/76 - 05/02/76
D-32713			05/03/76 - 05/05/76
D-32714			05/05/76 - 05/08/76
D-32715			05/08/76 - 05/11/76
D-32716			05/11/76 - 05/14/76
D-32717			05/14/76 - 05/17/76
D-32718			05/17/76 - 05/20/76
D-32719			05/23/76 - 05/26/76
D-32720			06/04/76 - 06/07/76
D-32721			06/07/76 - 06/10/76
D-32722			06/10/76 - 06/13/76
D-32723			06/13/76 - 06/16/76
D-32724			06/16/76 - 06/19/76
D-32725			06/19/76 - 06/22/76
D-32726			06/22/76 - 06/24/76
D-32727			06/25/76 - 06/28/76
D-32728			06/28/76 - 07/01/76
D-32729			04/17/76 - 04/20/76
D-32730			05/20/76 - 05/23/76
D-32731			08/27/76 - 08/30/76
D-32732			08/30/76 - 09/02/76

D#	C#	# of FILES	TIME SPAN
D-32734			09/05/76 - 09/08/76
D-32735			09/09/76 - 09/11/76
D-32736			09/11/76 - 09/14/76
D-32737			09/14/76 - 09/17/76
D-32738			09/17/76 - 09/21/76
D-32739			09/21/76 - 09/23/76
D-32740			09/23/76 - 09/26/76
D-32741			09/26/76 - 09/29/76
D-32742			09/29/76 - 10/02/76
D-32743			10/02/76 - 10/05/76
D-32744			10/05/76 - 10/08/76
D-32745			10/11/76 - 10/14/76
D-32746			10/14/76 - 10/17/76
D-32747			10/17/76 - 10/20/76
D-32748			10/20/76 - 10/23/76
D-32749			10/23/76 - 10/26/76
D-32750			10/26/76 - 10/29/76
D-32756			07/01/76 - 07/04/76
D-32757			07/04/76 - 07/07/76
D-32758			07/07/76 - 07/10/76
D-32759			07/10/76 - 07/13/76
D-32760			07/13/76 - 07/16/76
D-32761			07/16/76 - 07/19/76
D-32762			07/19/76 - 07/22/76
D-32763			07/22/76 - 07/25/76
D-32764			07/25/76 - 07/28/76
D-32765			07/28/76 - 07/31/76
D-32766			07/31/76 - 08/03/76
D-32767			[illegible]

D#	C#	# of FILES	TIME SPAN
D-32768			08/06/76 - 08/09/76
D-32769			08/09/76 - 08/12/76
D-32770			08/12/76 - 08/15/76
D-32771			08/15/76 - 08/18/76
D-32772			08/18/76 - 08/21/76
D-32773			08/21/76 - 08/24/76
D-32774			08/25/76 - 08/27/76
D-32775			10/08/76 - 10/11/76
D-33321			10/30/76 - 11/01/76
D-33322			11/01/76 - 11/04/76
D-33323			11/04/76 - 11/07/76
D-33324			11/07/76 - 11/10/76
D-33325			11/10/76 - 11/13/76
D-33326			11/13/76 - 11/16/76
D-33327			11/16/76 - 11/19/76
D-33328			11/19/76 - 11/22/76
D-33329			11/22/76 - 11/25/76
D-33330			11/25/76 - 11/28/76
D-33331			11/28/76 - 12/01/76
D-33332			12/01/76 - 12/04/76
D-33333			12/04/76 - 12/07/76
D-33334			12/07/76 - 12/10/76
D-33335			12/10/76 - 12/13/76
D-33336			12/13/76 - 12/16/76
D-33337			12/16/76 - 12/19/76
D-33338			12/19/76 - 12/22/76
D-33339			12/22/76 - 12/25/76
D-33340			12/25/76 - 12/28/76

170

D#	# OF FILES	TIME SPAN
D-33341		12/28/76 - 12/31/76
D-33342		01/01/77 - 01/03/77
D-33343		01/03/77 - 01/06/77
D-33344		01/06/77 - 01/09/77
D-33345		01/09/77 - 01/12/77
D-33346		01/12/77 - 01/15/77
D-33347		01/15/77 - 01/18/77
D-33348		01/18/77 - 01/21/77
D-33349		01/27/77 - 01/30/77
D-33350		01/30/77 - 02/02/77
D-33351		02/05/77 - 02/08/77
D-33437		02/02/77 - 02/05/77
D-33438		01/24/77 - 01/27/77
D-33439		03/07/77 - 03/10/77
D-33440		04/12/77 - 04/15/77
D-33441		04/09/77 - 04/12/77
D-33442		04/18/77 - 04/21/77
D-33443		04/21/77 - 04/24/77
D-33444		04/24/77 - 04/27/77
D-33445		04/15/77 - 04/18/77
D-33446		04/28/77 - 04/30/77
D-33447		02/29/77 - 03/04/77
D-33448		03/04/77 - 03/07/77
D-33449		03/13/77 - 03/16/77
D-33450		03/16/77 - 03/19/77
D-33451		03/19/77 - 03/22/77
D-33452		03/22/77 - 03/25/77
D-33453		03/26/77 - 03/28/77

D#	# OF FILES	TIME SPAN
D-33454		03/10/77 - 03/13/77
D-33455		04/06/77 - 04/09/77
D-33456		04/04/77 - 04/06/77
D-33457		03/28/77 - 03/31/77
D-33458		03/31/77 - 40/03/77
D-33459		02/09/77 - 02/11/77
D-33460		02/11/77 - 02/14/77
D-33461		01/21/77 - 01/24/77
D-33462		02/15/77 - 02/17/77
D-33463		02/17/77 - 02/20/77
D-33464		02/20/77 - 02/23/77
D-33465		02/23/77 - 02/26/77
D-33561		04/30/77 - 05/03/77
D-33562		05/03/77 - 05/06/77
D-33563		05/06/77 - 05/09/77
D-33564		05/09/77 - 05/12/77
D-33565		05/12/77 - 05/15/77
D-33566		05/15/77 - 05/18/77
D-33567		05/24/77 - 05/27/77
D-33568		05/27/77 - 05/30/77
D-33569		05/21/77 - 05/24/77
D-33570		05/31/77 - 06/02/77
D-33571		06/11/77 - 06/14/77
D-33572		06/02/77 - 06/05/77
D-33573		06/05/77 - 06/08/77
D-33574		09/09/77 - 06/11/77
D-33575		06/14/77 - 06/17/77
D-33576		06/17/77 - 06/20/77
D-33577		06/21/77 - 06/23/77

D#	# OF FILES	TIME SPAN
D-33578		05/19/77 - 05/21/77
D-33579		06/23/77 - 06/26/77
D-33580		06/29/77 - 07/02/77
D-33581		07/05/77 - 07/08/77
D-33582		07/11/77 - 07/14/77
D-33583		06/26/77 - 06/29/77
D-33584		07/03/77 - 07/05/77
D-33585		07/14/77 - 07/17/77
D-33586		07/08/77 - 07/11/77
D-33587		07/27/77 - 07/29/77
D-33588		08/05/77 - 08/07/77
D-33589		07/23/77 - 07/26/77
D-33590		08/11/77 - 08/13/77
D-33591		07/20/77 - 07/23/77
D-33592		07/29/77 - 08/01/77
D-33593		08/02/77 - 08/04/77
D-33621		08/07/77 - 08/10/77
D-33622		08/13/77 - 08/16/77
D-33623		08/16/77 - 08/19/77
D-33624		08/31/77 - 09/03/77
D-33625		09/02/77 - 09/05/77
D-33626		09/06/77 - 09/09/77
D-33627		09/10/77 - 09/13/77
D-33628		09/13/77 - 09/15/77
D-33629		09/25/77 - 09/27/77
D-33630		09/28/77 - 09/30/77
D-33631		07/17/77 - 07/20/77
D-33632		08/19/77 - 08/22/77
D-33633		08/22/77 - 08/25/77

256

D#	# OF FILES	TIME SPAN
D-33634		09/30/77 - 10/03/77
D-33635		10/04/77 - 10/06/77
D-33636		10/07/77 - 10/09/77
D-33637		10/09/77 - 10/12/77
D-33638		09/12/77 - 09/15/77
D-33639		08/25/77 - 08/28/77
D-33640		08/28/77 - 08/31/77
D-33641		09/18/77 - 09/20/77
D-33642		09/22/77 - 09/24/77
D-33643		11/11/77 - 11/13/77
D-33644		11/04/77 - 11/05/77
D-33645		09/15/77 - 09/18/77
D-33646		10/22/77 - 10/23/77
D-33647		11/20/77 - 11/23/77
D-33648		10/15/77 - 10/18/77
D-33649		10/24/77 - 10/27/77
D-33650		10/27/77 - 10/30/77
D-33651		11/05/77 - 11/08/77
D-33652		11/14/77 - 11/17/77
D-33653		11/23/77 - 11/25/77
D-33654		12/04/77 - 12/06/77
D-33655		11/08/77 - 11/10/77
D-33656		11/25/77 - 12/28/77
D-33657		11/16/77 - 11/19/77
D-33658		10/17/77 - 10/20/77
D-33659		11/28/77 - 12/01/77
D-33752		11/30/77 - 12/03/77
D-33753		12/09/77 - 12/12/77
D-33754		12/16/77 - 12/17/77

x5

D#	# OF FILES	TIME SPAN
D-33755		12/21/77 - 12/23/77
D-33756		12/14/77 - 12/17/77
D-33757		12/26/77 - 12/31/77
D-33758		12/23/77 - 12/26/77
D-33759		12/26/77 - 12/28/77
D-33760		02/05/78 - 02/08/78
D-33761		01/27/78 - 01/30/78
D-33762		01/24/78 - 01/21/78
D-33763		02/11/78 - 02/14/78
D-33764	2	03/01/78 - 03/04/78
D-33765		02/08/78 - 02/11/78
D-33766		02/14/78 - 02/18/78
D-33767		01/04/78 - 01/06/78
D-33768		01/09/78 - 01/12/78
D-33769		01/30/78 - 02/02/78
D-33770		02/27/78 - 03/01/79
D-33771		03/04/78 - 03/06/78
D-33772		03/07/78 - 03/10/78
D-33773		03/13/78 - 03/16/78
D-33774		01/01/78 - 01/03/78
D-33775		01/06/78 - 01/09/78
D-33776		03/11/78 - 03/13/78
D-33777		03/26/78 - 04/02/78
D-33778		01/15/78 - 01/18/78
D-33779		05/01/78 - 05/03/78
D-33780		01/13/78 - 01/15/78
D-33781		01/18/78 - 01/21/78
D-33782	2	02/02/78 - 02/04/78
D-33783		03/16/78 - 03/19/78

D#	# OF FILES	TIME SPAN
D-33784		03/20/78 - 03/22/78
D-33785		03/28/77 - 03/31/78
D-33786		04/07/78 - 04/08/78
D-33787		04/09/78 - 04/12/78
D-33788	2	04/28/78 - 04/30/78
D-33789		01/21/78 - 01/24/78
D-33790		02/18/78 - 20/20/78
D-33791		02/20/78 - 02/22/78
D-33792		02/23/78 - 02/26/78
D-33793		03/22/78 - 03/25/78
D-33822		03/31/78 - 04/03/78
D-33823		04/04/78 - 04/06/78
D-33824		04/18/78 - 04/21/78
D-33825		04/21/78 - 04/24/78
D-33826		04/25/78 - 04/27/78
D-33827		05/15/78 - 05/18/78
D-33828		05/24/78 - 05/27/78
D-33829		04/12/78 - 04/15/78
D-33830		04/15/78 - 04/18/78
D-33831		05/04/78 - 05/06/78
D-33832		05/18/78 - 05/21/78
D-33833		05/27/78 - 05/30/78
D-33834		05/06/78 - 05/09/78
D-33835		07/08/78 - 07/11/78
D-33836		06/05/78 - 06/08/78
D-33837		06/12/78 - 06/14/78
D-33838		06/15/78 - 06/17/78
D-33839		06/18/78 - 06/20/78
D-33840		07/05/78 - 07/08/78

D#	# OF FILES	TIME SPAN
D-33841		07/12/78 - 07/14/78
D-33842		07/14/78 - 07/17/78
D-33843		06/02/78 - 06/05/78
D-33844		06/08/78 - 06/11/78
D-33845		07/02/78 - 07/05/78
D-33846		07/17/78 - 07/20/78
D-33847		07/21/78 - 07/23/78
D-33848		07/23/78 - 07/26/78
D-33849		08/04/78 - 08/07/78
D-33850		06/26/78 - 06/29/78
D-33851		08/07/78 - 08/10/78
D-33852		08/10/78 - 08/13/78
D-33853		08/22/78 - 08/25/78
D-33854		08/25/78 - 08/28/78
D-33855		06/30/78 - 07/02/78
D-33856		08/14/78 - 08/15/78
D-33857		08/17/78 - 08/19/78
D-33858		08/19/78 - 08/22/78
D-33859		07/27/78 - 07/29/78
D-33860		08/31/78 - 09/03/78
D-33861	2	09/06/78 - 09/08/78
D-33862		09/13/78 - 09/15/78

TABLE 1 CALO OUTPUT RECORD

ITEM NO.	NAME	TYPE	OUTPUT BYTES
1	DAY	R*8	1 - 8
2	SECONDS	R*8	9 - 16
3	SPIN RT ASCENSION	R*8	17 - 24
4	SPIN DECLINATION	R*8	25 - 32
5	ROLL RT ASCENSION	R*8	33 - 40
6	ROLL DECLINATION	R*8	41 - 48
7	ASPECT RT ASCENSION	R*8	49 - 56
8	ASPECT DECLINATION	R*8	57 - 64
9	ERROR	R*8	65 - 72
10	FLAG	R*8	73 - 80
11	PITCH ANGLE	R*8	81 - 88
12	RT ASCENSION	R*8	89 - 96
13	DECLINATION	R*8	97 -104
14	SPARE	R*8	105 -112
15	SPARE	R*8	113 -120
16	POSITION X	R*8	121 -128
17	POSITION Y	R*8	129 -136
18	POSITION Z	R*8	137 -144
19	HEIGHT	R*8	145 -152
20	MAG. FIELD B	R*8	153 -160
21	RIGHT ASCENSION	R*8	161 -168
22	DECLINATION	R*8	169 -176
23	LONGTITUDE	R*8	177 -184
24	LATITUDE	R*8	185 -192
25	SPIN	R*8	193 -200
26	PREAMPS 12 - 9	I*4	201 -202
27	PREAMPS 8 - 1	I*4	203 -204
28	HV1	I*4	205 -208
29	HV2	I*4	209 -212
30	HV3	I*4	213 -216
31	HV4	I*4	217 -220
32	HV5	I*4	221 -224
33	HV6	I*4	225 -228
34	HV7	I*4	229 -232

RECFM

LRECL= 1432

BLKSIZE=

TABLE 1 CALO OUTPUT RECORD CONTD.,

ITEM NO.	NAME	TYPE	OUTPUT BYTES
35	MØDE	I*4	233 - 236
36	STATUS	I*4	237 - 240 0-5
37	BAD WORD	I*4	241 - 244
38	SUNGATE	I*4	245 - 248
39	BRAGG	I*4	249 - 252
40	DAY/NIGHT	I*4	253 - 256
41	REFERENCE	I*4	257 - 260
42	ANTI	I*4	261 - 264
43	UPPER CUT REJ.	I*4	265 - 268
44	RISETIME	I*4	269 - 272
45	LOWER LEVEL CROSSING	I*4	273 - 276
46	BUSY	I*4	277 - 280
47	TEMPERATURE 1	I*4	281 - 284
48	TEMPERATURE 2	I*4	285 - 288
49	TEMPERATURE 3	I*4	289 - 292
50	TEMPERATURE 4	I*4	293 - 296
51	TEMPERATURE 5	I*4	297 - 300
52	TEMPERATURE 6	I*4	301 - 304
53	LV1	I*4	305 - 308
54	LI1	I*4	309 - 312
55	LV2	I*4	313 - 316
56	LI2	I*4	317 - 320
57	LV3	I*4	321 - 324
58	LI3	I*4	325 - 328
59	LV4	I*4	329 - 332
60	LV5	I*4	333 - 336
61	THRESHOLDS LL	I*4	337 - 340
62	THRESHOLDS RT	I*4	341 - 344
63	THRESHOLDS FLARE	I*4	345 - 348
64	THRESHOLDS WINDOW	I*4	349 - 352
65	RASTER 1	I*4	353 - 356
66	RASTER 2	I*4	357 - 360
67	SPARE	I*4	361 - 364
68	SPARE	I*4	365 - 368

TABLE 1 CALO OUTPUT RECORD CONTD.,

ITEM NO.	NAME	TYPE	OUTPUT BYTES
69	SPARE	I*4	369 - 372
70	SPARE	I*4	373 - 376
71	SPARE	I*4	377 - 380
72	S. ATLANTIC ANOM FLG	I*4	381 - 384
73	BRAGG USED	I*4	385 - 388
74	SUNGATE USED	I*4	389 - 392
75	REFERENCE USED	I*4	393 - 396
76	STATUS USED	I*4	397 - 400
77	NO. MISSING EVENTS	I*4	401 - 404
78	NO. OF EVENTS	I*4	405 - 408
79	DATA WORDS	I*4	409 -1432

CALOO

ABSTRACT

CALOO is primarily a FORTRAN IV computer program, written to execute on the IBM 360/65 at MD&O (Bldg. 14) at GSFC. The program reads input data tapes received from IPD at GSFC containing OSO-8 satellite data for experimenter number 5. Output data tapes and computer printouts are created. The output tapes contain the original data in a reorganized fashion, such that it is more easily accessible for further analysis. The printed output summarizes the housekeeping of the experiment so that the experimenter may easily determine the status of the instrument's operation at any given time.

HISTORY

CALOO was written in the fall of 1975 at COLUMBIA ASTROPHYSICS LAB. by Dori Miller. The program was taken to GSFC for debugging and testing. It was installed on disk and put into production mode when data tapes were received in February 1976. Several minor modifications were made between February and April 1976. A notebook describing these modifications is located in Room 1012 at COLUMBIA ASTROPHYSICS LAB. The current executing version is CALOO.04. It is compiled under FØRTRAN, PARAM = XL.

INTRODUCTION

Data tapes containing label information, orbital information and OSO-8 science data for experimenter number 5, are prepared by IPD at Goddard Space Flight Center, Greenbelt, Maryland, and released to The Columbia Astrophysics Lab., for further processing.

Each data tape contains approximately 24 hour's worth of data. There are a variable number of files on each tape (usually between 15 and 22) and the end of the tape is signified by two end of file marks. It is assumed that both the files and the data contained in a file on the tape are contiguous in time.

Each file begins with one label record, followed by a variable number of orbital data records, followed by a variable number of experimental data records. A description of the label record format is found in Table 5, and the orbital record format is described in the section SUBROUTINE OBIT, and Table 6.

Each experimental data record represents one MAJOR FRAME of data. This is a spacecraft dependent time unit and is equal to 20.48 seconds of real time. Table 16 shows the experimental data record format, as supplied to C.A.L. by IPD.

The remainder of this document describes how the data tapes are read and processed.

GENERAL PROGRAM LOGIC

INITALIZE
↓
START
↓
BUFFER (1) = BUFFER (2)
↓
READ A RECORD INTO BUFFER (2)
↓
UNPACK HOUSEKEEPING FROM BUFFER (2) TO HOUSEKEEPING (3)
↓
CHECK AND PRINT HOUSEKEEPING (3)
↓
DATA (1) = DATA (2)
↓
UNPACK DATA FROM BUFFER (1) TO DATA (2)
↓
PROCESS REVOLUTIONS FROM DATA (1)
↓
MOVE HOUSEKEEPING (1 ← 2, ← 2 ← 3)
↓
GO TO START

PROGRAM STRUCTURE

CALOO consists of a main program, and nine subroutines all written in FORTRAN IV. There are six additional routines used for I/0 and bit manipulation that are provided in-line at MD&0. There are two 360 assembly language routines included in the deck also for bit manipulation. A summary of the program structure is included in Table 1. A detailed description of each program segment follows.

MAIN PROGRAM

The main program of CALOO is essentially a driver routine that calls subroutines in the proper order for processing one data tape.

The first section of code initializes counters that tell total tape quality and reads in the NAMELIST. (NAMELIST items are described in table 2; counters are described in table 3). The input tape is mounted on unit NTAP at file NFIL.

The next section initializes counters for each file, and positions the input tape to the proper file.

The main body of the program reads the input tape and calls subroutines, performing what is described in "GENERAL PROGRAM LOGIC".

The final section of the code takes into account error conditions on the input tape, prints out summary information for each file processed, as well as summary information for the total run. (Explanation of printed summary information is contained in tables 3 and 4).

SUBROUTINE LABEL (IBAD)

When a record type =1 is read from the input tape, the main program calls SUBROUTINE LABEL. The first input record of each file should be type 1. This routine prints out selected information from the label record. TABLE 5 describes the label record. It is taken from the document OSO-I PRODUCTION DATA FORMAT FOR CAL X-RAY SPECTROMETER AND POLARIMETER EXPERIMENT #5. Items that are starred are printed.

"IBAD", the parameter returned to MAIN is set to =0 if this is our experiment (experiment #5) and set to =1 if this is not our experiment. If IBAD=1, a message is printed by subroutine label, and when control returns to MAIN, all processing stops.

SUBROUTINE OBIT (IREC)

When a record type =2 is read from the input tape, the main program calls subroutine OBIT. The purpose of this routine is to collect orbital information which is given once per min, for later interpolation so that orbital information can be attached to each revolution record.

Each orbital record read contains up to 26 blocks of 20 parameters each. There are a variable number of orbital records in each file; however, they occur consecutively and appear after the label record and before the science data records. CALOO saves 14 of the 20 available parameters, and allows for up to 250 blocks. The saving sequence ends when the first parameter of a block is equal to 88888888. The first word of each orbital record contains the record type indicator, the next 8 words (all "words" in the orbit records are IBM double precision) are fill, such that the blocking sequencing starts with the tenth word of the record.

IREC is a record counter passed from MAIN to OBIT. It is used to flag the start of a new file, so that the counter NOB which indicates the number of orbit record collected can be reset to zero.

Table 6 describes the 20 parameters in the orbital records indicating which are saved and the array location of the saved parameters. The saved parameters are passed in array X(14,250) in CØMMØN/IØBBLK/.

If more than 250 blocks of orbital data are found in a file, a message is printed once for each record thereafter. No more than the first 250 are collected; however, processing continues.

SUBROUTINE HKUPK (NFILL, NBMF)

Every time an experimental data record (ITYP=3) is read by MAIN, this subroutine is called. The data record is read by MAIN into IBUF(i,2) and passed in COMMON/IBBL/IBUF (1080,2) to HKUPK. This subroutine unpacks the housekeeping data into the array IHK (i,3), where i can range 1 - 60.

The subroutine is set up as follows: First, the array IHK (i,3) is set equal to the missing code. Next, in a loop going from 1, NPAR, each relevant minor frame is examined. The array NMF, which is set in a data statement, defines which minor frames are to be considered. If the flag field for a minor frame indicates error bits, the unpacking of that minor frame is bypassed, leaving the related IHK item set to the missing code, and incrementing the counter NFILL*. If the minor frame number in the data stream does not agree with the expected minor frame number the unpacking is bypassed and the counter NBMF* is incremented. Otherwise a GØ TØ (, , . . .), K statement defines which section of the code is to be used to unpack the data. Table 7 indicates the indexing used for the unpacking, and Table 14 shows which bits are used and any conversions necessary to obtain the final relevant IHK items.

Last, when the unpacking is completed, the preamp status (off or on) are combined as defined in Table 14, "reference" is determined as shown in Table 9, "status" is determined as shown in Table 8 and rates are decoded as shown in Table 9. Control is then returned to MAIN.

*See Table 3

SUBROUTINE HKPRNT (IREC)

For every experimental data record read, MAIN calls subroutine HKPRNT immediately after calling HKUPK. IREC, the passed parameter is a counter initalized for each file, of the number of experimental data records read (i.e. major frame number).

This subroutine always prints housekeeping from the first 5 major frames and the last major frame of each file. Other major frames are printed if any one of certain housekeeping items change by more than an input limit from the "standard frame".(described below). Table 3 indicates which items are checked and Table 14 shows which items are printed.

If IREC =1, (the first major frame of a file), constants are set and an array of the "standard frame values" is set to missing. Then, for all frames an array of trigger indicators is set to the alphanumeric blank. In a loop, each significant item is checked against the "standard frame's" item. If this item differs from the previous by strictly more than that item's limits a counter (NBAD) is incremented and the appropriate item of the trigger indicator array is set to the alphanumeric '*'. If the item or the "previous item" is missing the check is not performed, however, if the "standard frame" value is missing and the frame item is non-missing, the standard is set to this item.

Next the "z" check is performed. The right ascension and declination of the spin axis are saved from the first record of each file. The quantity Z is computed:

```
if RA1 = saved right ascension
   DE1 = saved declination
   RA  = this right ascension
   DE  = this declination
then Z  = sin (DE) *sin(DE1)+cos(DE)*cos(DE1)
                   *cos(RA-RA1)
```

SUBROUTINE HKPRNT (IREC) CONTD.

The limit for Z variance from the previous Z is .99999048 (i.e. 114°). If Z varies by more than this amount and Z is less than .99999048, a print is triggered, and RA1 and DE1 are reset to RA1=RA, DE1=DE.

Next, if this is not the first 5 frames or the last frame and no print trigger has been set, control returns to Main. Otherwise time is converted from milliseconds to hour, min, secs, the second line for printing is stored and the first line is printed. If this is the last record of the file, or the line counter has reached 60, the array of "second line" data is printed.

Last, when a print has been triggered the "standard frame values" are set to the values of this frame, and control is returned to Main.

SUBROUTINE DTUPK (IDREC)

Main calls subroutine DTUPK after calling HKPRNT for every experimental data record read, from the second record on. During program execution they are, stored in core, two major frames of raw data in IBUF and three major frames of housekeeping in IHK. DTUPK uses this information to gather and store 2 major frames of bit streamed data and its flag fields as well as other information needed to form revolutions. Saved items are described below. For each call to DTUPK saved items are first moved from the (2) to the (1) index, then data and "other information" is unpacked from IBUF (j,1) into item (i,2). In this fashion, housekeeping from IHK (i, 2) applies to the major frame being unpacked. (See "General Program Logic"). Double precision data and MIP-PSI times are pointed to in the IBUF array through an EQUIVALENCE statement. The fact that these times can be missing and/or mixed up leads to complication.

Saved parameters are passed to REV in COMMON/IRBLK/, and are as follows:

IDAT(320,2)- the bit streamed data
from IBUF (33+8N,1), IBUF (34+8N,),
and IBUF (35+8N,16) 16 left bits
only where N=0, 127 (for each minor frame)
(2.5 words from 128 minor frames =320 words)

IFLG(128,2)- error bits in a minor frame
from IBUF (38+8N,1) 7 right most
bits only. Where N=0, 127 for each minor frame.

TIM(4,2)- times of MIP or PSI pulses
if IREF = missing, no TIM's saved
if IREF = 0 MIP's saved
TIM(1,2) from IBUF(9,i)
TIM(2,2) from IBUF(10,i)
TIM(3,2) from IBUF(11,i)
if IREF = PSI's saved

TIM(i,2) from IBUF(12,i)

TIM(2,2) from IBUF(12,i)

TIM(3,2) from IBUF(14,i)

NOTE: TIM(4,2) is inserted in SUBROUTINE REV

- a check is made for the second MIP or PSI pulse being in the first slot. SPIN is computed as wheel SPIN RATE from IBUF(1057,i) and IBUF(1058,i) (a double precision variable). Then SPIN = (2Π/wheel spin rate) + .001. Now if TIM(i,2) is non-missing and is greater than or equal to SPIN, TIM(3,2) = TIM(2,2) and TIM(2,2) = TIM (1,2) and TIM (1,2) is set to missing.

ISTAT (2) - status from IHK(10,2)

IREF (2) - reference from IHK(15,2)

IBA (2) - Bragg angle from IHK(13,2)

ISAG (2) - sungate angle from IHK(12,2)

The above four items are from IHK(i,2) unless it is missing. If it is missing and IHK(i,1) = IHK(i,3) then the item is set equal to IHK(i,3)

DPDAT(15,2) - double precision data where

(1,2) is day of year of regular field from IBUF(2,i)

(2,2) is seconds of year of regular field from IBUF(3,i) $*10^{-3}$

(3,2) right ascension of spin axis (radians) from IBUF (1065,i), (1066,i)

(4,2) declination of spin axis (radians) from IBUF(1067,i) (1068,i)

(5,2) right ascension of roll axis (radians) from IBUF(1069,i)

(6,2) declination of roll axis (radians) from IBUF(1071,i), (1072,i)

(7,2) right ascension of aspect reference axis (radians) from IBUF(1073,1), (1074,1)

(8,2) declination of aspect reference axis (radians) from IBUF(1075,1) (1076,1)

(9,2) probable error of attitude solution (arc-min) from IBUF(1077,i), (1078,1)

(10,2) attitude flag field (see table 10) from IBUF (179,1), (1080,i)

(11,2) wheel spin rate (radians/sec) from IBUF(1057,1), IBUF(1058,i)

(12,2) through (15,2) are spares

SUBROUTINE REV(NREC)

Subroutine REV is the heart of CALO. Here, data that have been gathered, unpacked, interpreted and stored by HKUPK and DTUPK are finally combined to form revolution records. Subroutine REV is called by MAIN for every record experimental data read in a file from the third record to end. NREC, the passed parameter is simply the experimental data record counter, and is used by REV to initalize certain parameters at the start of each new file being processed.

Subroutine REV considers one major frame's worth of data each time it is called, although certain information from the succeeding major frame is needed and used in the determination of revolutions. Thus, generally REV works with data passed by DTUPK in COMMØN/HKBLK/ in the item (i,1) arrays, and housekeeping passed by HKUPK in COMMON/HKBLK/ in IHK(i,1).

First REV determines if there are any revolutions to process in this frame. If status, or reference as determined in DTUPK is missing or if status =2 and Sun Gate angle are missing a counter is incremented* and control returns to MAIN. If the first MIP-PSI, TIM(1,1), is non-missing, and we have previously saved a spin rate (SPINL is non-missing) then we proceed with processing; otherwise if TIM(1,1) or TIM(1,2) is missing a counter is incremented* and control returns to MAIN. However, if both are non-missing, SPINL is computed as TIM(2,1)-TIM(1,1) and processing continues.

Next REV determines how many revolutions there are to process. As TIM(1,1) and SPINL now must both be non-missing, if TIM(2,1) and/or TIM(3,1) are missing, Times are inserted as TIM(missing,1) =TIM(missing -1,1) + SPINL. Now the number of revolutions to process "NTIM" is determined; there are 2 revolutions to process unless: 1) if TIM(3,1) is less than TCØN (128./800.) there are 3 revolutions to process; or 2) if TIM(2,1) is greater than TCØN there is only one revolution to process. Next, the time of the end of the last revolution,

*See Table 4

called TIM(NTIM + 1,1) is set as the time of the first MIP-PSI of the next frame plus TCON.

Status and reference are now checked. If they are not the same in this frame and the next, NTIM is decremented by one, and a flag is set indicating that the last revolution will not be processed. Also, the seconds of this frame and the next are checked. If the time difference is within $\pm$ 1 sec of TCON the frames are considered to be consecutive. If the frames are not consecutive, NTIM is decremented by one and a flag is set indicating that the last revolution will not be processed. NOTE: no check is made for a day change, so the last revolution of the frame preceding a change of day will always be thrown out.

Finally, before processing of revolutions begins a check is made on the first time of the next frame to process. If the first time is missing and the second is non-missing, the time computed as TIM(NTIM + 1,1) for this frame is inserted so that the next frame will have a start time.

Next, the bit streamed data from this frame and the next is unpacked into 8 bit "words" into the array IBYT, so that markers can be found. (For status 0, no markers are looked for; for status 1 and 2 a marker is 20 consecutive bits on (1), starting on an 8 bit "word"; for status 3-5 a marker is 8 bits on, i.e. an 8 bit "word" $=255_{10}$).

Processing of revolutions is now begun, in a loop from 1 to NTIM (the number of revolutions in this frame). The IBYT array is searched for a marker (depending on the status) whose "time" lies between the times of the MIP or PSI pulses defining this revolution. (The diagnostic parameters NOMARK and NBMARK described in Table 4 are utilized if no marker can accurately be found in this range). The "time" of the marker is computed to the accuracy of one bit, from its location in the IBYT array. Spin is computed as the difference of the MIP or PSI pulses defining this revolution, and stored in XD(10). Time of the revolution is computed as the time of the start of the frame plus the time to the MIP or PSI pulse defining the start of the revolution.

DAY and SECS, stored in DPDAT(1,1) and DPDAT(2,1) are adjusted if a change of day occurred at this point.

Next the unpacking and storing of data occurs. As this is done depending on status, the logic for each of the three options is described separately below. Table 12 describes arrays, flags and counters used for the unpacking processes.

A. FOR STATUS 3,4,5

512 8 bit bytes of data are collected, one at a time from the IBYT array, starting at the first IBYT element after the marker for this revolution. The minor frame location of each data item collected is computed and the flag field checked. If the flag field of the minor frame is non-zero, the data item is set equal to a missing code (255_{10}) and a counter IFIX is incremented. Data items are packed, 2 per word into the array IRDAT, such that IRDAT(1) = data item (1) *256 + data item (2) . . . IRDAT(256)=date item(511) *256 + data item(512). NEVN, the number of events in a revolution is always set equal to 512.

B. FOR STATUS 1

An event for status 1 and 2 consists of 20 bits of data; therefore, starting with the bit following the marker, 256 data words are unpacked from the array IDAT into the array IRDAT. Next, the 256 data words are examined in turn. If an event is non-zero, and the flag field(s) of the minor frame or frames in which it occurred are zero, a counter NEVN is incremented and the event is stored in IBUF(NEVN). IF the flag field test was not passed, the counter IFIX is incremented and the next event is considered. This process continues up to 256 times, or until an event is examined that is zero. It is then assumed that all the data for the entire revolution have been stored IBUF. Next the "stuck word" is checked for. If NEVN=1 and the first event of this revolution is equal to the first event in the previous revolution, NEVN is set equal to zero.

C. FOR STATUS 2

The collecting of status 2 data is more complicated than the other statuses because data are collected from the sungate to the Bragg angle, rather than from a marker to the next. Therefore, for each "revolution" as defined by REV, some of the data belongs to the previous revolution and some to this revolution. Using Table 12 the logic is as follows:

Having found a marker, if status is not equal to 2, LOUT is set equal to zero. Also, if the revolution is thrown out, LOUT is set equal to zero. If status is equal to 2, LGET is set to zero and starting with the bit following the marker, 256 20 bit data words are unpacked from the array IDAT to the IRDAT array. Looping through the 256 data words, first if an IRDAT element equals zero, the end of data is assumed and the loop is exited. Next, the flag fields are checked as described in status 1, with IFIX being incremented and the next event examined if the flag fields are non-zero. The azimuth is then unpacked, giving the following options:

1). It is $\geq$ sungate *16+8; then, if LOUT=0, go on to the next event (i.e. this is the end of a revolution that we do not have the beginning of).

Otherwise, increment NEVN and store the event in the IBUF array such that IBUF(NEVN)=IRDAT(I).

2). It is < sungate *16+8; if LGET=0 this is the first event of the next revolution so counters are stored and reset as follows:

ISPR(12)= NEVN; ISPR(11) = IFIX; NEVN = 0; IFIX = 0; LGET = 1; LOUT = LOUT + 1. In either case, next NEVN is incremented and the event is stored in IRDAT such that IRDAT(NEVN)= IRDAT(I).

This process continues until either an IRDAT element is examined that is equal to zero, or all 256 events have been examined. If LGET was never set to 1 in the loop (i.e. the first event of the next revolution was never found)

the counters are stored and reset at this time. Next if LOUT >1 the stored buffers are output. Then the IBUF array is set equal to the missing code, and processing continues as for the other statuses except that the stored buffers are not output at the time they are output for the others. In this fashion, for each status revolution, the first half of the data is collected and housekeeping etc. prepared for output. When the next revolution is being processed the second half of the data is collected, the revolution is output, then housekeeping etc. prepared for output.

D. FOR STATUS 0

Status 0 is assumed to have no data. Revolutions are output as for other statuses with the following exceptions. NEVN is not necessarily zero; nor is the data buffer necessarily set to the missing code. "Rates" are not calculated, and the South Atlantic Anomaly flag is not computed.

Data having been unpacked, revolutions are next prepared for output, as described below, with the exceptions as noted above.

First the five rates, ANTI, BUSY, UCR, TRIM and LLC are approximately assigned to this revolution. If we let frame 'A' be the frame in which this revolution occurs, frame 'B' is the next frame (frame A + 1) and frame 'C' is equal to frame A + 2. Then if the MIP or PSI pulse defining the beginning of this revolution is > 16, rates are always assigned as the rates from frame 'C'. If the MIP or PSI pulse is $\leq$ 16, the rates for this revolution are set equal to zero unless this is the first revolution in this frame, and all rates in the previous revolution are zero. Then rates are assigned to this revolution from frame 'B'.

Next, REV calls GOBIT to interpolate for the orbital elements associated with this revolution. Orbital information and housekeeping is then moved into the output buffers. ASPECT is called to calculate right ascension and declination of the aspect axis. Next the South Atlantic Anomaly flag is calculated.

If status =0 it is set to missing. If status =1, HV7 and HV8 are considered. If status = 2-5, HV1 thru HV5 are considered. Then, if any one non-missing value considered is greater than 292, the flag is set to 1. If all are missing, the flag is set to missing. If all are non-missing and all are equal 292 the flag is set to zero.

Events that were stored in the IRDAT array are next moved to the output buffer IBUF, and (for all except status 2 data) subroutine OUTPUT is called to write the revolution onto tape.

Next, the IBUF array is set to the missing code, and the next revolution is considered.

When all revolutions in a major frame have been considered, control returns to MAIN.

SUBROUTINE GOBIT (DAY, SEC, ICALL)

This subroutine is called by subroutine REV once for each revolution to be output. Its purpose is to interpolate in time to attach orbital information to a revolution. The array DX(9) is computed in GOBIT from the array X(14,NOB) that was stored by subroutine OBIT. (See table 6 for a description of the X array items). The DX(9) array is then passed back to REV through COMMON/IOBBLK/, and set equal to the XD array such that XD(1) = DX(1), XD(2)=DX(2) . . . . XD(9) = DX(9). This procedure aligns the DX elements for output in the COMMON/OUTBLK/, such that of the double precision variables output, DX(1) will be the 16th, DX(2) the 17th . . . DX(9) the 24th.

The logic of GOBIT is as follows:

DAY and SEC are the computed time of the revolution being processed. ICALL is the number of times GOBIT has been called this file (i.e. it is reset to zero for each file). Assuming that the X array is time ordered, it is searched for the first time (using X(1,i) and X(2,i)) that is greater than or equal to the time of this revolution. The (index -1) of this occurrence is saved and called "L". The start location of the search was set to i if ICALL =1; it is now set to L to save searching time for the next revolution.

If no time was found that meets the above conditions or if L=0, the message "CANT FIND ORBIT TIME FOR DAY = . . . SEC= . . . D1, D2, SEC1, SEC2= . . . , . . . , . . . , . . . " is printed where DAY is the passed parameter "DAY" and SEC is the passed parameter "SEC" and D1, D2, SEC1, SEC2 are the first and last day of the saved X array, and the first and last seconds of the X array. Control then returns to REV which will still output this revolution output. NOTE: NO MISSING CODE OR INDICATION OF THIS OCCURRENCE IS INCLUDED IN THE OUTPUT RECORD. Where this information is critical the user

must refer to the CALO printed output to determine which revolutions are affected.

Assuming a time was found in the array, four time differences are computed

DT = SEC - X(2,L)

T1 = X(2,L) - X(2,L-1)

T2 = X(2,L + 1) - X(2,L-1)

T3 = X92,L + 1) - X(2,L)

NOTE: Although a check and fix is made on DT being affected by crossing over a day, NO CHECK IS MADE ON T1, T2, or T3.

Next, code is looped through for each item to be computed. The loop index "I" goes from 6, 14 referring to items of the X array. For I = 11 and I = 13 special code is executed to assure that the resulting DX computed items are in the proper range. (This "special code is best described by reading the code itself). Generally, for each of the 9 items to be computed X(I,L), X(I,L-1) and X(I,L + 1) are saved as XI, XMI and XPI respectively. With proper constraints on the possibility of multiplying by zero the following equation is computed.

A=(XI-XM1)/T1 + (T1/T2)* [((XP1 - XI)/T3) - ((XI -XM1)/T1)]

Given the above A the resulting DX array items are computed as shown in Table 11.

Control is then returned to REV.

SUBROUTINE ASPECT (DTHETA)

This subroutine is called by REV immediately before each revolution is output. It calculates the right ascension and declination of the spacecraft, and returns the values in the D array (see table 13). The subroutine argument DTHETA is the time of the MIP or PSI of this revolution, divided by the SPIN (as calculated in REV) all times 2Π.

[DTHETA= (TIM(K,1)/XE(10)) *6.2832.] D(3), D(4), D(7) and D(8) as described in table 13 are inputs to the routine, with D(7) and D(8) being recalculated by the routine and D(12) and D(13) being calculated. A listing of the code of ASPECT best indicates the equations used in the calculations.

SUBROUTINE OUTPUT

This subroutine is called by REV for each revolution to be written to tape. The appropriate tape number and file number were read by MAIN in the namelist and are passed to OUTPUT through COMMON/TBLK/. The output buffers are formed in subroutine REV and are passed through COMMON/OUTBLK/. A detailed description of the output record is found in table 13. Each revolution record to output is 1432 bytes long, as declared in a data statement. These are blocked 5/block, as defined on the DD statements for the output tape. Thus, the physical record length on the tape is 7160 bytes.

The first time OUTPUT is called, spare output elements are set to missing, and a call is issued to MOUNT, to mount the output tape and position it at the correct file.

Whenever OUTPUT is called by REV, a call is issued to FWRITE to write the revolution onto tape, and control returns to REV.

ADDITIONAL ROUTINES

POSN(1,NTAP,IFILE). This FORTRAN callable routine positions a tape mounted on unit =NTAP to file IFILE. The first argument indicates that the next operation on the tape will be a forward read.

MOUNT(1,NTAP,IDTI,NFIL). This routine mounts the tape named "IDTI" on unit NTAP at file NFIL. The first argument is as above.

FREAD(IBUF,NTAP,NBYTES,&850,&800). Reads a record of length NBYTES from unit NTAP into contiguous core locations starting at IBUF. Transfer is to statement 850 on EØF and statement 800 on an I/O error.

FWRITE(D,JTAP,LENG). Writes a record of length LENG onto unit JTAP from contiguous core locations starting at D.

SHIFTL(IY,N). In line function provided by PARAM=XL Fortran compiler extension to perform a left shift of N bits on word IY.

LØR(IX,IY). In line function provided by PARAM=XL Fortran extension to logically OR, IX and IY.

ADDITIONAL ROUTINES (CONTD.)

GBYTE (PCKD,UNPK,INOFST,NBIT)

GBYTES (PCKD,UNPK,INOFST,NBIT,NSKIP,ITER)

PCKD: The fullword in memory from which unpacking is to begin: successive fullwords will be fetched as required.

UNPK: The fullword in memory into which the initial BYTE of unpacked data is to be stored; subsequent BYTES will be stored into successive fullwords as required.

INOFST: A fullword integer specifying the initial offset in bits of the first BYTE, counted from the leftmost bit in PCKD.

NBITS: A fullword integer specifying the number of bits in each BYTE to be unpacked. Legal BYTE widths are in the range 1 - 32: BYTES of width .LT. 32 will be right justified in the low-order positions of the UNPK fullwords, with high-order zero fill.

NSKIP: A fullword integer specifying the number of bits to skip between successive BYTES. All non-negative skip counts are legal.

ITER: A fullword integer specifying the total number of BYTES to be unpacked, as controlled by INOFST, NBIT and NSKIP above. All non-negative iteration counts are legal.

NOTES: 1). A multiple -BYTE access (GBYTES) with iter=0 (but not 1) is exactly equivalent to a single-BYTE access (GBYTE).

2). An error detected in the calling sequence of either GBYTE or GBYTES suppresses BYTE access, and sets the first element of UNPK = X'FFFFFFFF'.

TABLE 1

Summary of Program Structure (All routines in FORTRAN unless otherwise noted).

NAME	ARGUMENTS	CALLED FROM	CALLS	COMMON BLOCKS	PURPOSE
MAIN	N/A	N/A	POSN MOUNT FREAD GBYTE LABEL ORBIT HKUPK HKPRNT DTUPK REV	/IBBL/ /HKBLK/ /TRIGBL/ /TBLK/ /SUMBLK/	MAIN PROGRAM TO INITIALIZE, READ INPUT TAPE, DRIVER FOR PROCESSING AND PRINT SUMMARY INFORMATION AT END OF PROCESSING
LABEL	IBAD	MAIN	-	/IBBL/ /TBLK/	TO PRINT LABEL INFORMATION
OBIT	IREC	MAIN	-	/IBBL/ /IOBBLK/ /TBLK/	TO COLLECT RELEVANT ORBITAL INFORMATION FOR A FILE
HKUPK	NFILL	MAIN	GBYTE	/IBBL/ /HKBLK/	TO UNPACK HOUSEKEEPING INFORMATION FOR EACH FRAME
HKPRNT	IREC	MAIN	GBYTE	/IBBL/ /HKBLK/ /TRIGBL/ /TBLK/	TO CHECK FOR CHANGES IN THE HOUSEKEEPING, AND PRINT IF CHANGES HAVE OCCURRED
DTUPK	IREC	MAIN	GBYTE LOR	/IBBL/ /HKBLK/ /IRBL/	TO SAVE TIME AND REFERENCE AND TO UNPACK THE DATA INTO A "BIT STREAM"
REV	NREC	MAIN	GBYTES GBYTE GOBIT ASPECT	/HKBLK/ /IRBL/ /IOBBLK/ /SUMBLK/	TO IDENTIFY AND GATHER DATA INTO REVOLUTIONS
GOBIT	DAY	REV		/IOBBLK/ /TBLK/	TO INTERPOLATE FOR FOR A REVOLUTION
ASPECT	-	REV	-	/OUTBLK/	TO COMPUTE ASPECT INFORMATION FOR A REVOLUTION
OUTPUT	-	REV	MOUNT FWRITE	/OUTBLK/ /TBLK/	TO WRITE REVOLUTION RECORDS TO TAPE

TABLE 1 (CONTD.)

Additional Routines

POSN MOUNT FREAD FWRITE	FORTRAN CALLABLE I/O ROUTINES PROVIDED AT MD&O FOR TAPE HANDLING
SHFTL LOR	BIT MANIPULATION ROUTINES PROVIDED IN LINE AT MD&O
GBYTE GBYTES	IBM 360 ASSEMBLY LANGUAGE ROUTINES TO ALLOW FORTRAN ACCESS TO BIT STRINGS. ROUTINES OBTAINED FROM NATIONAL CENTER FOR ATMOSPHERIC RESEARCH, BOULDER, CO.

TABLE 2

NAMELIST/IPTRIG/

1. NITR — The number of items that will cause a print trigger currently set=29. It is passed in COMMON/TRIGBL/.

2. ITR — The indices of the items that will cause a print trigger. Dimensioned ITR(40), it is passed in COMMON/TRIGBL/. It is currently set =1, 9, 10, 11, 12, 13, 14, 15, 35, 36, 37, 38, 39, 40, 2, 3, 4, 5, 6, 7, 8, 27, 28, 29, 30, 31, 32, 33, 34.

   NOTE: The indices are the same as the "ITEM NUMBER" of the integer outputs.

3. LIM — The limits for each "ITR" which if strictly exceeded (+ or -) will cause a print trigger. Dimensioned LIM(40), it is passed in CØMMON/TRIGBL/. It is currently set LIM= 0,0,0,0,0,0,0,0,0,0,0,0,0,0,4,4,4, 4,4,4,4,1,18,1,30,1,1.

4. IPALL — A flag, passed in CØMMON/TRIGBL/ which if non-zero will cause every major frame's housekeeping to be printed. Currently set =0.

5. IP2 — A flag, passed in CØMMON/TRIGBL/ which if non-zero will cause right ascension and declination to also be checked for a print trigger. It is currently set =1.

6. IPNUM1 — The number of items to be printed on line 1. Passed in CØMMON/TRIGBL/, it is currently set =20.

7. IPNUM2 — The number of items to be printed on line 2. Passed in CØMMON/TRIGBL/ it is currently set =21.

TABLE 2 (CONTD.)

8.	IMC	The missing code inserted for unavailable housekeeping items. It is passed in CØMMON/HKBLK/ and is set =1.
9.	NPAR	The number of parameters examined to collect a frame of housekeeping. It is passed in COMMON/HKBLK/ and NPAR=38.
10.	JPAR	The number of housekeeping items collected in a major frame. It is passed in CØMMON/HKBLK/ and currently =50.
11.	JTAP	The tape unit number for the output tape, passed in CØMMON/TBLK/ and set =11.
12.	JFIL	The file number to write to on unit JTAP. It is passed in CØMMON/TBLK/ and normally =1.
13.	NTAP	The tape unit number for the input tape. It is passed in CØMMON/TBLK/ and NTAP=10.
14.	NFIL	The first file to read from on the input tape. It is passed in CØMMON/TBLK/ and normally =1.
15.	IDT1	Dimensioned (2), it is the input tape number. It is passed in CØMMON/TBLK/ and normally changes with each program execution. It is entered in the form IDT1='CUxxxx'.
16.	IDT2	Dimensioned (2), it is the output tape number. It is passed in CØMMON/TBLK/ and normally changes with each program execution.

TABLE 3

MAIN PROGRAM COUNTERS

IDREC -	Counter of the number of experimental data records in each input tape file.
IERR -	No. of bad reads on input tape. If IERR exceeds 10, all processing stops.
IFILE -	Input tape file counter.
IOREC -	Counter of number of orbital records in each input file.
ISUM(6)-	Counters of the number of revolutions for each status (0-5) processed. They are printed at the end of a successful run. They are passed thru COMMON /SUMBLK/ and are incremented in subroutine REV.
JERR -	No. of bad record types on the input tape. If JERR exceeds 10, all processing stops.
KSUM(6)-	Counters of the number of revolutions of each file. They are printed as processing of each file is completed.
LREC -	The record number of the last EOF encountered.
NBMF -	No. of minor frames examined that were out of order. This number is printed at the end of each successful run.
NFILL -	No. of minor frames thrown out due to bad flag fields. This number is printed at the end of a successful run.

TABLE 4

The four diagnostic parameters at end of each file printout from CALOO are the following:

1). NOREF - The counter is incremented if reference or status is missing or if status = 2 and sungate are missing. The counter refers to the number of major frames in which no revolutions were processed for the above reasons.

2). NLØUT - If a status change occurs the last revolution of the old status is not processed and this counter is incremented.

3). NBMARK - A marker has been found but the flag field indicates error bits. The marker is therefore thrown out and the next marker is considered. A counter is incremented. NOTE: this could also cause NOMARK to be incremented.

4). NOMARK - No marker is found between the MIP's defining a revolution. NOTE: If BAD FLAG caused a marker to be thrown out and it was real then both counters would be incremented.

NOTES: These parameters are calculated in subroutine REV passed in CØMMON/SUMBLK/, and printed in MAIN.

6/21/75 launch

TABLE 5

Label Record Description

BYTES	FORM	CONTENTS
1-2	Integer *2	Data type indicator -1 for label record
3-4	Integer *2	Record number
5-8	Integer *4	Year
* 9-12	"	Station number
* 13-16	"	Station tape number
* 17-20	"	Analog field number
* 21-24	"	Buffer field number
25-28	"	Day count of year
29-32	"	Start time (m sec) of data file
33-36	"	End time (m sec) of data file
37-40	"	Experimenter no. (= 5 for C.A.L)
* 41-44	"	Decom number
* 45-48	"	Reel number
* 49-52	" "	Master tape file
* 53-56	"	Master tape number
* 57-60	" "	Type of attitude computations
61-62	Integer *2	Dwell mode I.D. of word dwelled on
63-64	Integer *2	Format generator I.D.
* 65-68	Integer *4	Day of year of Epoch for orbital elements
Π * 69-72	Integer *4	Sec of day of Epoch for orbital elements
* 73-76	"	Semi-major axis, a (km)
* 77-80	"	Eccentricity e (ratio)
* 81-84	"	Inclination, I (deg)
* 85-88	"	Right ascension of ascending node
* 89-92	"	Rate of change of R.A. ascending node
* 93-96	"	Argument of perigee

TABLE 5 (CONTD.)

BYTES	FORM	CONTENTS
*97-100	Integer *4	Rate of change of argument of perigee
*101-104	"	Period
*105-108	"	Rate of change of period
109-112	"	Date
113-116	"	Day of year
117-120	"	Apparent sidereal time coordinate system ref. data time and position
*121-124	"	No. of frames in file
*125-128	"	No. of frames with a loss of synch
*129-132	"	No. of frames with bit errors
133-136	"	Spare
*137-140	"	No. of frames with zero bit errors
141-144	"	No. of frames with 1 or more bit errors
*145-148	"	No. of data records in file
149-4320	"	Spare

* Means printed by subroutine label of CALOO

Π Converted to HR, MIN, SEC and printed

NOTE: BYTES 73-108 and 117-120 are actually floating point numbers which were converted to fixed point by multiplying by 10^5.

TABLE 6

ORIBITAL PARAMETERS

(20 IBM Double Precision words)

ORBITAL NUMBER	NAME	STORED IN
1	Type of data	-
2	Day count, time	X(1, NOB)
3	Seconds, time	X(2, NOB)
4	Position X	X(3, NOB)
5	Position Y	X(4, NOB)
6	Position Z	X(5, NOB)
7	Velocity X	X(6, NOB)
8	Velocity Y	X(7, NOB)
9	Velocity Z	X(8, NOB)
10	Longitude	X(13, NOB)
11	Latitude	X(14, NOB)
12	Height	X(9, NOB)
13	Solar Vector Sx	-
14	Solar Vector Sy	-
15	Solar Vector Sz	-
16	McIllwain L	-
17	Magnetic field B	X(10, NOB)
18	Right Ascension	X(11, NOBO
19	Declination	X(12, NOB)
20	Ascending node	-

TABLE 7

INDEXING OF HKUPK

K 1,NPAR	NMF (0-127)	GØ TØ	INDEX OF IHK	NAME
1	56	10	1	Preamps 8-1
2	88	15	1,35,36	Preamps 12-9, Thresholds 1, & 2
3	76	20	2	HV1
4	77	20	3	HV2
5	78	20	4	HV3
6	79	20	5	HV4
7	80	20	6	HV5
8	81	20	7	HV6
9	82	20	8	HV7
10	24	25	9,11,37,38	Mode, badword, thresholds 3 & 4
11	96	30	12	Sungate angle
12	120	35	13	Bragg angle
13	30	40	14	Sensed day/night
14	0	45	41	7 bytes saved for decoding rates into IHK indices of 16, 17, 18, 19, 20 (see "Rates determined")
15	1	45	42	
16	2	45	43	
17	3	45	44	
18	97	45	45	
19	98	45	46	
20	99	45	47	
21	110	50	21	Temperature 1
22	111	50	22	Temperature 2
23	112	50	23	Temperature 3
24	113	50	24	Temperature 4

TABLE 7 (CONTD.)

INDEXING OF HKUPK

K 1,NPAR	NMF	GO TO	INDEX OF IHK	NAME
25	114	50	25	Temperature 5
26	115	50	26	Temperature 6
27	83	55	27	LV1
28	89	60	28	LI1
29	90	65	29	LV2
30	121	70	30	LI2
31	25	75	31	LV3
32	26	80	32	LI3
33	122	85	33	LV4
34	57	90	34	LV5
35	28	95	39,40	Raster 1, Raster 2
36	30	100	48	Day for reference
37	72	105	49	B string for reference
38	95	105	50	A string for reference

TABLE 8

STATUS DETERMINED AS FOLLOWS:

A). MODE = IHK(9,3) STATUS = IHK(10,3)

MODE = 1	THEN	STATUS = 0
MODE = 2		STATUS COMPUTED
MODE = 3		STATUS = 2
MODE = 4		STATUS = 1
MODE = MISSING		STATUS COMPUTED

B). TO COMPUTE STATUS

GIVEN THE FOLLOWING MATRIX OF INEQUALITIES:

	0	1	2	3	4	5
IPA1 IHK(1,3)	> -1 < 1	> -1 < 1	> 254 < 256	> 240 < 255	> 0 < 16	> 0 < 16
IPA2 IHK(1,3)	> -1 < 1	> 0 < 2	> -1 < 1	> -1 < 1	> -1 < 1	> -1 < 1
MODE IHK(9,3)	> 0 < 2	> 3 < 5	> 2 < 4	> 1 < 3	> 1 < 3	> 1 < 3
HV1 IHK(2,3)	> 291 < 293	> 291 < 345	> 393 < 1124	> 393 < 1124	> 393 < 1124	> 356 < 394
HV2 IHK(3,3)	> 291 < 293	> 291 < 345	> 393 < 1124	> 393 < 1124	> 393 < 1124	> 356 < 394
HV3 IHK(4,3)	> 291 < 293	> 291 < 345	> 393 < 1124	> 393 < 1124	> 393 < 1124	> 356 < 394
HV4 IHK(5,3)	> 291 < 293	> 291 < 345	> 393 < 1124	> 393 < 1124	> 393 < 1124	> 356 < 394
HV5 IHK(6,3)	> 291 < 293	> 291 < 345	> 393 < 1124	> 393 < 1124	> 356 < 394	> 393 < 1124
HV6 IHK(7,3)	> 291 < 293	> 393 < 1124	> 291 < 345	> 291 < 345	> 291 < 345	> 291 < 345
HV7 IHK(8,3)	> 291 < 293	> 393 < 1124	> 291 < 345	> 291 < 345	> 291 < 345	> 291 < 345

When the value of an item listed meets any of the six inequalities for that item, a counter is incremented for that status. The status with conditions most often met is the status assigned to this major frame. However, if more than one status meets the maximum number of conditions, the assigned status is declared as missing.

TABLE 9

REFERENCE AND RATES

A. REFERENCE - inserted in IHK (15,3)

Day for reference - IHK(48,3)

= get D(6330) bit 6 if =0 get D(62,30)

bit 7

if =1 get D(63,30)

bit 5

B string - IHK(49,3)

= get D(96,72) bits 1-8 if $\leq$ 64 IHK(49,3)=0

if $\leq$ 64 IHK(49,3)=1

A string - IHK(50,3)

= get D(96,95) bits 1-8 if $\leq$ 64 IHK(50,3)=0

if > 64 IHK(50,3)=1

If IHK(48,3), IHK(49,3) or IHK(50,3) are missing then IHK(15,3) is left set to the missing code.

Otherwise IHK(48,3)=1, unless IHK(49,3)=0 and IHK(50,3)=1 then IHK(15,3)=1

B. RATES: inserted as follows:

IHK(16,3)=ANTI

IHK(17,3)=UPPER CUT REJECTS

IHK(18,3)= RISETIME (missing for status 3,4,5)

IHK(19,3)= LOWER LEVEL CROSSINGS

IHK(20,3) = BUSY

From Table 7 we see 7 bytes were saved for the rates in IHK(41,3) thru IHK(47,3). These were saved from D(98,NMF) all 8 bits.

TABLE 9 (CONTD.)

To decode rates, if any one of the seven saved bytes are missing, all rates are left set to the missing code. Otherwise

1). Status = 3,4,5

ANTI = IHK(16,3) = IHK(41,3) (bits 1-8) *128 + IHK(42,3) (bits 1-6*) *2

BUSY = IHK(20,3), = IHK(42,3) (bits 7-8) *8192 + IHK (43,3) (bits 1-8) *32
+ IHK (44,3) (bits 1-4) *2

UCR = IHK(17,3) = IHK(44,3) (bits 5-8) *512 + IHK (45,3) (bits 1-8) *2

LLC = IHK(19,3) = IHK(46,3) (bits 1-8) *512 + IHK (47,3) (bits 1-8) *2

2). Status 0, 1, 2

BUSY = IHK(20,3) = IHK(41,3) (bits 1-8) *16 + IHK (42,3) (bits 1-3) *2

ANTI = IHK(16,3) = IHK(42,3) (bits 4-8) *32 + IHK (43,3) (bits 1-4) *2

RTIM = IHK(18,3) = IHK(43,3) (bits 5-8) *512 + IHK (44,3) (bits 1-8) *2

UCR = IHK(17,3) = IHK(45,3) (bits 1-8) *8 + IHK (46,3) (bits 1-2) *2

LLC = IHK(19,3) = IHK(46,3) (bits 3-8) *512 + IHK (47,3) (bits 1-8) *2

*bit 1 = MSB

TABLE 10

The attitude flag field contains 20 bits of information. The flag field is in IBM double precision floating point format. When the double precision floating point word is set equal to an integer, then the least significant 20 bits of the 32 bit integer contain the following information. Bit 1 is the least significant bit and when the respective bit is set to one, the condition is present.

Bit	Significance
1	A gas firing has occurred
2	Roll angle ϕ was extrapolated
3	Pitch angle η was extrapolated
4	Aspect angle B_s was extrapolated
5	Spin rate ω was extrapolated
6	Roll angle ϕ was interpolated
7	Pitch angle η was interpolated
8	Aspect angle B_s was interpolated
9	Spin rate ω was interpolated
10	Roll angle ϕ was calculated from a star solution
11	Pitch angle η was calculated (in day time from sun sensor, at night from a star solution)
12	Aspect angle B_s was calculated
13	Spin rate ω was calculated
14	Roll angle ϕ)
15	Pitch angle η) Raw data values

TABLE 10 (CONTD.)

Bit	Significance
16	Aspect angle B_s) were present
17	Spin rate ω)
18	Aspect angle B_s was present from MIP
19	Spin rate ω was present from MIP
20	1 = night at the beginning of major frame
	0 - day at the beginning of major frame

The general guidelines in interpreting the bit settings are as follows: When the lower 9 bits are not set except for bit 6 in the day time, there were good solutions for the attitude state vector. The interpolation of missing values yields, generally, good results. Extrapolation over long periods of time, generally, yields less confident results and should be carefully interpreted by the user. Gas firings should be carefully interpreted because of larger uncertainty in the state vector, particularly during the gas firings.

TABLE 11 - ORBITAL ELEMENTS

NAME	DX	OUTPUT ITEM	
POSITION X	DX(1)	16	= X(3,L) + X(6,L)*DT + (A*DT*DT)*.5
POSITION Y	DX(2)	17	= X(4,L) + X(7,L)*DT + (A*DT*DT)*.5
POSITION Z	DX(3)	18	= X(5,L) + X(8,L)*DT + (A*DT*DT)*.5
HEIGHT	DX(4)	19	= X(9,L) + DT*A
MAGNETIC FIELD B	DX(5)	20	= X(10,L) + DT*A
RIGHT ASCENSION	DX(6)	21	= X(11,L) + DT*A
DECLINATION	DX(7)	22	= X(12,L) + DT*A
LONGITUDE	DX(8)	23	= X(13,L) + DT*A
LATITUDE	DX(9)	24	= X(14,L) + DT*A

TABLE 12

ARRAYS, FLAGS AND COUNTERS USED

BY REV FOR UNPACKING EVENTS

IBUF(256)	- Output buffer for events, passed to OUTPUT in COMMON/OUTBLK/
IDAT(320,2)	- The bit streamed data (for 2 frames), collected in DTUPK and passed to REV in COMMON/IRBLK/
IFIX	- Counter of the number of events in a revolution that were thrown out due to bad minor frame flag fields.
IRDAT(256)	- Temporary storage buffer used in unpacking and storing the bit streamed data.
ISPR(12)	- Number of events that are output (i.e. the final NEVN)
ISPR(11)	- Number of events thrown out. The final IFIX.
JHK(40)	- The output buffer for housekeeping
LGET	- Flag used for status 2 data to say we have found the start of the second revolution's data. (i.e. we found the first azimuth less than the sungate *2 + 8)
LOUT	- Counter that if non-zero indicates that the last revolution processed was status 2
NEVN	- Counter of the number of events in a revolution

TABLE 13 CALO OUTPUT RECORD

NAME	TYPE	BUFFER	OUTPUT BYTES	FROM
DAY	R*8	D(1)	1 - 8	See DTUPK where D(1) = DPDAT(1) adjusted if DAY crossing in REV
SECONDS	R*8	D(2)	9 - 16	See DTUPK where D(2)=DPDAT(2) adjusted plus MIP or PSI in REV
SPIN RT ASCENSION	R*8	D(3)	17 - 24	See DTUPK, D(3) = DPDAT(3)
SPIN DECLINATION	R*8	D(4)	25 - 32	
ROLL RT ASCENSION	R*8	D(5)	33 - 40	
ROLL DECLINATION	R*8	D(6)	41 - 48	
ASPECT RT ASCENSION	R*8	D(7)	49 - 56	See DTUPK, D(7) = DPDAT(7) recalculated in ASPECT
ASPECT DECLINATION	R*8	D(8)	57 - 64	See DTUPK D(8)= DPDAT(8) re-calculated in ASPECT
ERROR	R*8	D(9)	65 - 72	See DTUPK D(9) = DPDAT(9)
FLAG	R*8	D(10)	73 - 80	See DTUPK D(10) = DPDAT (10) and Table 10
PITCH ANGLE	R*8	D(11)	81 - 88	See DTUPK D(11) = DPDAT(11)
RT ASCENSION	R*8	D(12)	89 - 96	Calculated in ASPECT
DECLINATION	R*8	D(13)	97 -104	Calculated in ASPECT
SPARE	R*8	D(14)	105 -112	Set to - .9999 DO IN OUTPUT
SPARE	R*8	D(15)	113 -120	Set to - .9999 DO IN OUTPUT
POSITION X	R*8	XD(1)	121 -128	See Table 11 XD(1) = DX(1)
POSITION Y	R*8	XD(2)	129 -136	" " XD(2) = DX(2)
POSITION Z	R*8	XD(3)	137 -144	" " XD(3) = DX(3)
HEIGHT	R*8	XD(4)	145 -152	" " XD(4) = DX(4)
MAG. FIELD B	R*8	XD(5)	153 -160	" " XD(5) = DX(5)
RIGHT ASCENSION	R*8	XD(6)	161 -168	" " XD(6) = DX(6)
DECLINATION	R*8	XD(7)	169 -176	" " XD(7) = DX(7)
LONGITUDE	R*8	XD(8)	177 -184	" " XD(8) = DX(8)
LATITUDE	R*8	XD(9)	185 -192	" " XD(9) = DX(9)
SPIN	R*8	XD(10)	193 -200	Calculated in REV
PREAMPS 12 - 9	I*4	JHK(1)	201 -202	See Table 14
PREAMPS 8 - 1	I*4	JHK(1)	203 -204	"
HV1	I*4	JHK(2)	205 -208	"
HV2	1*4	JHK(3)	209 -212	"
HV3	I*4	JHK(4)	213 -216	"
HV4	I*4	JHK(5)	217 -220	"
HV5	I*4	JHK(6)	221 -224	"
HV6	I*4	JHK(7)	225 -228	"
HV7	I*4	JHK(8)	229 -232	"
MØDE	I*4	JHK(9)	233 -236	"
STATUS	I*4	JHK(10)	237 -240	"
BAD WORD	I*4	JHK(11)	241 -244	"
SUNGATE	I*4	JHK(12)	245 -248	"
BRAGG	I*4	JHK(13)	249 -252	"
DAY/NIGHT	I*4	JHK(14)	253 -256	"
REFERENCE	I*4	JHK(15)	257 -260	See Table 9
ANTI	I*4	JHK(16)	261 -264	See Table 9 and REV
UPPER CUT REJ.	I*4	JHK(17)	265 -268	See Table 9 and REV
RISETIME	I*4	JHK(18)	269 -272	See Table 9 and REV
LOWER LEVEL CROSSINGS	I*4	JHK(19)	273 -276	See Table 9 and REV
BUSY	I*4	JHK(20)	277 -280	See Table 9 and REV

TABLE 13 CALO OUTPUT RECORD (CONTD.)

NAME	TYPE	BUFFER	OUTPUT BYTES	FROM
TEMPERATURE 1	I*4	JHK(21)	281 - 284	See Table 14
TEMPERATURE 2	I*4	JHK(22)	285 - 288	"
TEMPERATURE 3	I*4	JHK(23)	289 - 292	"
TEMPERATURE 4	I*4	JHK(24)	293 - 296	"
TEMPERATURE 5	I*4	JHK(25)	297 - 300	"
TEMPERATURE 6	I*4	JHK(26)	301 - 304	"
LV1	I*4	JHK(27)	305 - 308	"
LI1	I*4	JHK(28)	309 - 312	"
LV2	I*4	JHK(29)	313 - 316	"
LI2	I*4	JHK(30)	317 - 320	"
LV3	I*4	JHK(31)	321 - 324	"
LI3	I*4	JHK(32)	325 - 328	"
LV4	I*4	JHK(33)	329 - 332	"
LV5	I*4	JHK(34)	333 - 336	"
THRESHOLDS LL	I*4	JHK(35)	337 - 340	"
THRESHOLDS RT	I*4	JHK(36)	341 - 344	"
THRESHOLDS FLARE	I*4	JHK(37)	345 - 348	"
THRESHOLDS WINDOW	I*4	JHK(38)	349 - 352	"
RASTER 1	I*4	JHK(39)	353 - 356	"
RASTER 2	I*4	JHK(40)	357 - 360	"
SPARE	I*4	ISPR(1)	361 - 364	Set to -1 in OUTPUT
SPARE	I*4	ISPR(2)	365 - 368	Set to -1 in OUTPUT
SPARE	I*4	ISPR(3)	369 - 372	Set to -1 in OUTPUT
SPARE	I*4	ISPR(4)	373 - 376	Set to -1 in OUTPUT
SPARE	I*4	ISPR(5)	377 - 380	Set to -1 in OUTPUT
S. ATLANTIC ANOM FLG	I*4	ISPR(6)	381 - 384	See REV
BRAGG USED	I*4	ISPR(7)	385 - 388	See Table 14
SUNGATE USED	I*4	ISPR(8)	389 - 392	"
REFERENCE USED	I*4	ISPR(9)	393 - 396	"
STATUS USED	I*4	ISPR(10)	397 - 400	"
NO. MISSING EVENTS	I*4	ISPR(11)	401 - 404	See Table 14 and Table 12
NO. OF EVENTS	I*4	ISPR(12)	405 - 408	See Table 14 and Table 12
DATA WORDS	I*4	IBUF(1)	409	See REV For STATUS 3, 4, 5:
		THRU	THRU	512 events are packed 2 per word
		IBUF(256)	1432	For STATUS 1,2:- A maximum of 256 events - one per word

TABLE 14

LOCATION AND CONVERSION OF 1*4 OUTPUT ITEMS

NAME	FROM D(WORD,MF)	MSB=1 - 8=LSB BIT - BIT	CONVERTED	PRINTED	TRIGGER	LIMITS
PREAMPS 12-9 = IPA2	98, 88	5 - 8	IF EITHER MISSING = MISSING	YES	YES	0
PREAMPS 8-1 = IPA1	98, 56	1 - 8	OR+IPA2*256+IPA1		YES	
HV1	98, 76	1 - 8	292+(3.26*BITS)+0.5	YES	YES	±4
HV2	98, 77	1 - 8	"	YES	YES	±4
HV3	98, 78	1 - 8	"	YES	YES	±4
HV4	98, 79	1 - 8	"	YES	YES	±4
HV5	98, 80	1 - 8	"	YES	YES	±4
HV6	98, 81	1 - 8	"	YES	YES	±4
HV7	98, 82	1 - 8	"	YES	YES	±4
MODE	98, 24	6 - 7	BITS+1	YES	YES	0
STATUS	SEE TABLE 8			YES	YES	0
BAD WORD	98, 24	8	NO	YES	YES	0
SUNGATE	98, 96	1 - 8	NO	YES	YES	0
BRAGG	98, 120	1 - 8	NO	YES	YES	0
DAY/NIGHT	62, 30	7	NO	YES	YES	0
REFERENCE	SEE TABLE 9			YES	YES	0
ANTI	SEE TABLE 9 AND REV			YES	NO	
UPPER CUT REJ.	"			YES	NO	
RISETIME	"			YES	NO	
LOWER LVL CROSS.	"			YES	NO	
BUSY	"			YES	NO	
TEMP 1	97, 110	1 - 8	+65.4579	YES	NO	
TEMP 2	97, 111	1 - 8	-1.02546*BITS	YES	NO	
TEMP 3	97, 112	1 - 8	+0.0053256*BITS^2	YES	NO	
TEMP 4	97, 113	1 - 8	-1.013*10^{-5}*BITS^3	YES	NO	
TEMP 5	97, 114	1 - 8		YES	NO	
TEMP 6	97, 115	1 - 8		YES	NO	
LV1	98, 83	1 - 8	(4.54+0.00438*BITS)*100+0.5	YES	YES	± 1
LI1	98, 89	1 - 8	1.79*BITS+0.5	YES	YES	±18
LV2	98, 90	1 - 8	(4.53+0.00458*BITS)*100+0.5	YES	YES	± 1
LI2	98, 121	1 - 8	1.22*BITS+0.5	YES	YES	±18

TABLE 14 (CONTD.)

LOCATION AND CONVERSION OF 1*4 OUTPUT ITEMS

NAME	FROM D(WORD,MF)	MSB=1 - 8=LSB BIT - BIT	CONVERTED	PRINTED	TRIGGER	LIMITS
LV3	98, 25	1 - 8	(4.51+0.00442*BITS)*10+0.5	YES	YES	1
LI3	98, 26	1 - 8	5.77*BITS+0.5	YES	YES	±30
LV4	98, 122	1 - 8	(13.68+0.0139*BITS)*10+0.5	YES	YES	1
LV5	98, 57	1 - 8	(25.45+0.0258*BITS)*10+0.5	YES	YES	1
THRESHOLDS LL	98, 88	3 - 4	NO	YES	YES	0
THRESHOLDS RT	98, 88	1 - 2	NO	YES	YES	0
THRESHOLDS FLARE	98, 24	3 - 4	NO	YES	YES	0
THRESHOLDS WINDOW	98, 24	1 - 2	NO	YES	YES	0
RASTER 1	63, 28	7	NO	YES	YES	0
RASTER 2	63, 28	8	NO	YES	YES	0
SPARE				NO	NO	
SPARE				NO	NO	
SPARE				NO	NO	
SPARE				NO	NO	
SPARE				NO	NO	
SOUTH ATLANTIC ANOMALY FLAG (SEE REV)				NO	NO	
BRAGG USED = BRAGG			UNLESS { IF THE ITEM IS MISSING AND PREVIOUS = NEXT, THEN THIS = NEXT; IF PREVIOUS ≠ NEXT, THEN THIS = MISSING (SEE DTUPK)	NO	NO	
SUNGATE USED = SUNGATE				NO	NO	
REFERENCE USED = REFERENCE				NO	NO	
STATUS USED = STATUS				NO	NO	

NO. OF MISSING EVENTS = IFIX (SEE TABLE 12)
NO. OF EVENTS = 512 FOR STATUS 3,4,5
" " " = NO. OF VALID EVENTS FOR STATUS 1,2 (NOT INCLUDING THROW OUT)

TABLE 15

COMMON BLOCKS AND VARIABLES IN COMMON

BLOCK NAME	ROUTINE LOCATION	VARIABLES IN THE BLOCK
/IBBL/	MAIN	IBUF(1080,2)- input buffer for tape reading
	LABEL OBIT HKUPK DTUPK	
/HKBLK/	MAIN	IHK(60,3)- housekeeping element (see tables 7 and 14)
	HKUPK	IMC-missing code, input through name-list (see table 2)
	HKPRNT	NPAR-no. of minor frames examined, (see table 2)
	DTUPK	JPAR- no. of housekeeping elements to get (see table 2)
/TRIGBL/	MAIN	NITR- no. of items causing print trigger (see table 2)
	HKPRNT	ITR(40)- indices of print trigger items (see table 2) LIM(40)- limits of trigger items (see table 2) IPALL- flag to print all housekeeping (see table 2) IP2- flag to check right ascension and declination for print trigger (see table 2) IPNUM1- no. of items printed on line 1 (see table 2)
/SUMBLK/	MAIN	ISUM(6)- no. of revs of each status (see table 3) KSUM(6)- no. of revs of each status in each file (see table 3) NOREF- diagnostic parameter (see table 4) NLOUT- diagnostic parameter (see table 4) NBMARK-diagnostic parameter (see table 4) NOMARK- diagnostic parameter (see table 4)
/TBLK/	MAIN LABEL OBIT HKPRNT GOBIT OUTPUT	JTAP- unit no. of output tape (see table 2) JFIL- file no. of output tape (see table 2) NTAP- unit no. of input tape (see table 2) NFIL- file no. of input tape (see table 2) IDT1(2)- input tape no. (see table 2) IDT2(2)- output tape no. (see table 2) IPAGE- counter to number output by page number JLINE- output line counter, to determine when a new page should be started

TABLE 15 (CONTD.)

COMMON BLOCKS AND VARIABLES IN COMMON

BLOCK NAME	ROUTINE LOCATION	VARIABLES IN THE BLOCK
		NOREF- diagnostic parameter (see table 4)
		NLOUT- diagnostic parameter (see table 4)
		NBMARK- diagnostic parameter (see table 4)
		NOMARK- diagnostic parameter (see table 4)
/TBLK/	MAIN	JTAP- unit no. of output tape (see table 2)
	LABEL	JFIL- file no. of output tape (see table 2)
	OBIT	NTAP- unit no. of input tape (see table 2)
	HKPRNT	NFIL- file no. of input tape (see table 2)
	GOBIT	IDT1(2)- input tape no. (see table 2)
	OUTPUT	IDT(2)- output tape no. (see table 2)
		IPAGE- counter to number output by page number
		JLINE- output line counter, to determine when a new page should be started
/IOBBLK/	OBIT	X(14,250)- saved orbital parameters (see table 6)
	REV	DX(9)- output orbital parameters (see table 11)
	GOBIT	NOB- the number of sets of orbital parameters read
/IRBLK/	DTUPK	IDAT(320,2)- bit streamed data (see DTUPK)
	REV	IFLG(128,2)- error bit flag fields (see DTUPK)
		TIM(4,2)- times of MIP-PSI pulses (see DTUPK)
		DPDAT(15,2)- saved double precision data (see DTUPK)

TABLE 15 (CONTD.)

COMMON BLOCKS AND VARIABLES IN COMMON

BLOCK NAME	ROUTINE LOCATION	VARIABLES IN THE BLOCK
		ISTAT(2)- status (see DTUPK)
		IBA(2)- Bragg angle (see DTUPK)
		ISGA(2)- sungate angle (see DTUPK)
		IRF(2)- reference (see DTUPK)
/OUTBLK/	REV	D(15)- output double precision data (see table 13)
	ASPECT	XD(10)- output orbital parameters (see table 13)
	OUTPUT	JHK(40)- output housekeeping (see table 13)
		ISPR(12)- output spares and flags (see table 13)
		IBUF(256)- output data

TABLE 16

OSO-1 PRODUCTION DATA FORMAT FOR CAL X-RAY SPECTROMETER

AND POLARIMETER EXPERIMENT #5

BYTE	CONTENTS	
1-2	Data Type Indicator 1 = Label Data 2 = Orbit Data 3 = Experiment Data	
3-4	Record Number	
5-8	Day of year of regular time field	
9-12	Milliseconds of day of regular time field	
13-16	Spare	
17-24	Day of year of fine time field	
25-32	Microseconds of day of fine time field	
33-36	Time interval in seconds from the beginning of the major frame to the first MIP in the major frame (floating point).	
37-40	Time interval in seconds from the beginning of the major frame to the second MIP in the major frame (floating point).	
41-44	Time interval in seconds from the beginning of the major frame to the third MIP in the major frame (zero if there are only two MIP's in the major frame) (floating point).	
45-48	Time interval in seconds from the beginning of the major frame to the first sun pulse in the major frame (floating point).	
49-52	Time interval in seconds from the beginning of the major frame to the second sun pulse in the major frame (zero if there are only two sun pulses in the major frame) (floating point).	
53-56	Time interval in seconds from the beginning of the major frame to the third sun pulse in the major frame (zero if there are only two sun pulses in the major frame) (floating point).	
57-128	Rotation Matrix from OSO body coordinates to celestial coordinates	
129+32N	D (4,j)	
130+32N	D (5,j)	
131+32N	D (6,j)	
132+32N	D (36,j)	
133+32N	D (37,j)	
134+32N	D (38,j)	Science Data
135+32N	D (68,j)	
136+32N	D (69,j)	
137+32N	D (70,j)	
138+32N	D (100,j)	
139+32N	D (32,j)	Spacecraft Clock (Minor Frame Count)
140+32N	D (33,j)	
141+32N	D (62,j)	Control Subsystem (Raster Status, Coarse
142+32N	D (63,j)	Sun Sensor Amplitude, Error, etc.)

TABLE 16 (CONTD.)

OSO-1 Production Data Format for CAL X-Ray Spectrometer

and Polarimeter Experiment #5 (contd.)

BYTE	CONTENTS	
143+32N	D (66,j)	
144+32N	D (67,j)	Wheel Azimuth Angle
145+32N	D (77,j)	Wheel Control Signal Unit Housekeeping
146+32N	D (96,j)	
147+32N	D (97,j)	Subcommutator Data (Housekeeping Data)
148+32N	D (98,j)	
149+32N	D (99,j)	
150+32N	D (126,j)	
151+32N	D (127,j)	Wheel Aspect Data
152+32N	D (F_1,j)	
153+32N	D (F_2,j)	
154+32N	Spares	
155+32N	Spares	
156+32N	Spares	
157+32N	Spares	
158+32N	Spares	
159+32N	Spares	
160+32N	Spares	
4225-4232	Wheel spin rate (radians/sec)	
4233-4240	Pitch angle (radians)	
4241-4248	Roll angle (radians)	
4249-4256	Aspect angle (radians)	
4257-4264	Right ascension of spin axis (radians)	
4265-4272	Declination of spin axis (radians)	
4273-4280	Right ascension of roll axis (radians)	
4281-4288	Declination of roll axis (radians)	
4289-4296	Right ascension of aspect reference axis (radians)	
4297-4304	Declination of aspect reference axis (radians)	
4305-4312	Probable error of attitude solution (arc min)	
4313-4320	Attitude flag field	

IBM

IBM System/360 Assembler Coding Form

GX20-8160-6 U/M 050*
Printed in U.S.A.

PROGRAM FØRTRAN CAL ØSØ-8 STATUS SØRT
PROGRAMMER
DATE
PUNCHING INSTRUCTIONS: GRAPHIC / PUNCH
PAGE 1 OF
CARD ELECTRO NUMBER

STATEMENT

Name (1–8)	Operation (10–14)	Operand (16–71)	Identification Sequence (73–80)
C**	EXAMPLE OF A PRØGRAM TØ READ CAL ØSØ-8 PRØDUCTIØN DATA TAPES		
C	AND SPLIT ØFF THE THREE EXPERIMENTS ØNTØ 3 ØUTPUT TAPES		
C	( ) AT END ØF A LINE ARE NOT TO BE INCLUDED BUT MERELY EXPLANITØRY		
	DIMENSION IBUF(358)		
	REAL*8 C(2)		
	EQUIVALENCE(IBUF(1), C(1))		
	NAMELIST/ INPTS/ NFIL		
C**	IBUF IS BUFFER INTO WHICH DATA IS READ		
C	C(2) EQUIVALENCED AS ABOVE WILL CONTAIN DAY AND SECONDS OF EACH RECORD		
C	NFIL IS INPUT=NUMBER OF FILES ON THE INPUT TAPE (NORMALLY NFIL=3)		
	NREC=0	(RECØRD CØUNTER FØR EACH INPUT FILE)	
	JFIL=1	(INPUT FILE COUNTER)	
	NBYT=1432	(NØ. ØF BYTES TØ BØTH READ AND WRITE=LOGICAL REC)	
	NTAPI=10	(UNIT NØ. FØR INPUT TAPE)	
	NTAPØ1=21	(UNIT NO. FOR STATUS=1=POLARIMETER)	
	NTAPØ2=22	(UNIT NO. FOR STATUS=2=STELLAR SPECTROMETER)	
	NTAPØ3=23	(UNIT NO. FOR STATUS=3,4 AND 5=SOLAR SPECTROMETER)	
	NRØ1=0	(RECORD COUNTER FOR UNIT 21)	
	NRØ2=0	(RECORD COUNTER FOR UNIT 22)	
	NRØ3=0	(RECORD COUNTER FOR UNIT 23)	
C**	READ THE NAMELIST -- AND PRINT IT		
	READ(5,INPTS)		
	WRITE(6,INPTS)		

A standard card form, IBM electrostatic (GX20) is available for punching source statements from this form. Instructions for using this form are in the IBM System/360 Assembler Language reference manual. Address comments concerning this form to IBM Nordic Laboratory, Publications Development, Box 962, S 181 09 Lidingö 9, Sweden.

*No of forms per pad may vary slightly.

PROGRAM FØRTRAN CAL ØSØ-8 STATUS SØRT | PUNCHING INSTRUCTIONS | GRAPHIC | PUNCH | PAGE 2 OF 4

PROGRAMMER | DATE | CARD ELECTRO NUMBER

STATEMENT

```
CXX INSERT HERE CØDE TØ MØUNT TAPES ØN UNITS 10,21,22, AND 23
CXX    AND PØSITIØN UNIT 10 AT BEGINNING AND UNITS 21,22 AND 23 AT
CXX    WHERE YØU WANT THEM--PØSSIBLY YØU WØULD READ DØWN THE THREE TAPES
CXX    TØ AN EØF, AND BACKSPACE ØVER THE FILE MARK

CXXEND ØF INITIALIZATION CØDE
CXXREAD (IN A LØØP) A RECØRD FRØM UNIT 10 INTØ BUFFER IBUF
CXX    ALLØWING FØR ERRØR AND EØF ESCAPES AND WRITE IBUF TØ APPRØPRIATE UNIT
    4 CØNTINUE
      CALL FREAD(IBUF,NTAPI,NBYT,&97,&98)      (REPLACE WITH EQUIVALENT I/O ROUTINE)
      NREC=NREC+1
      IF(NREC.EQ.1) WRITE(6,100) JFIL,NREC,C(1),C(2)
  100 FORMAT(1X,'FØR FILE=',I2,' RECORD=',I5,' DAY=',F8.2,' SECS=',F10.2)
CXXSTATUS IS LØCATED IN IBUF(100)-- CHECK IT
CXX   IF STATUS=0 THE EXPERIMENT IS OFF, SO BYPASS THE RECORD
      IF (IBUF(100).EQ.0) GØ TØ 4
      IF (IBUF(100).LT.1.OR.IBUF(100).GT.5) GØ TØ 96
      GØ TØ (10,20,30,30,30), IBUF(100)
   10 NRØ1=NRØ1+1
      CALL FWRITE(IBUF,NTAPØ1,NBYT)            (REPLACE WITH EQUIVALENT I/O ROUTINE)
      GØ TØ 4
   20 NRØ2=NRØ2+1
      CALL FWRITE(IBUF,NTAPØ2,NBYT)            (REPLACE WITH EQUIVALENT I/O ROUTINE)
      GØ TØ 4
```

PROGRAM: FØRTRAN ØSØ-8 STATUS SØRT
PROGRAMMER:
DATE:
PUNCHING INSTRUCTIONS: GRAPHIC / PUNCH

CARD ELECTRO NUMBER:

STATEMENT

```
   30 NRØ3=NRØ3 +1
      CALL EWRITE(IBUF, NTAPØ3, NBYT)          (REPLACE WITH EQUIVALENT I/O ROUTINE)
      GØ TØ 4
CXXEND ØF READ- SORT WRITE
CXX
CXX ERRØR IN STATUS- PRINT AND IGNØRE THE RECØRD
   96 WRITE(6, 200)  JFIL, NREC, C(1), C(2), IBUF(100)
  200 FORMAT(1X, 'BAD STATUS AT FILE=', I2, ' REC=', I5, ' DAY=', F8.2, ' SEC='
     1 , F10.2, ' STATUS=', I10)
      GØ TØ 4
CXX ERRØR ØN INPUT TAPE READ- PRINT AND IGNORE THE RECØRD
   97 WRITE(6, 201) JFIL, NREC, C(1), C(2)
  201 FØRMAT(1X, 'BAD READ--FILE=', I2, ' REC=', I5, ' DAY=', F8.2, ' SECS=',
     1 F10.2)
      GØ TØ 4
C**ENDFILE ENCOUNTERED ON INPUT TAPE
C** PRINT, INCREMENT FILE COUNTER, CHECK TO CONTINUE OR END
   98 WRITE(6, 202) JFIL, NREC, C(1), C(2)
  202 FORMAT(1X, 'EØF FOR FILE=', I2, 'AFTER REC=', I5, ' LAST DAY=', F8.2,
     1 ' LAST SECS=', F10.2)
      JFIL=JFIL+1
      IF( JFIL .GT. NFIL) GØ TØ 50
      NREC=0
      GØ TØ 4
```

IBM

IBM System/360 Assembler Coding Form

Printed in U.S.A.

PROGRAM FØRTRAN ØSØ-8 STATUS SØRT

PROGRAMMER

DATE

PUNCHING INSTRUCTIONS: GRAPHIC / PUNCH

PAGE 4 OF 4

CARD ELECTRO NUMBER

STATEMENT

Identification Sequence

```
C** END ØF DATA ØN INPUT TAPE
   50 WRITE(6,203) NFIL, NRØ1, NRØ2, NRØ3
  203 FORMAT(1X,'JØB CØMPLETE AFTER',I3,' FILES PROCESSED',/,
     1      1X,I5,'RECØRDS WRITTEN FØR STATUS 1',/,
     2      1X,I5,'RECØRDS WRITTEN FØR STATUS 2',/,
     3      1X,I5,'RECØRDS WRITTEN FOR STATUS (4,5')
CXXENDFILE THE ØUTPUT TAPES
      ENDFILE(NTAPØ1)
      ENDFILE(NTAPØ2)
      ENDFILE(NTAPØ3)
      STOP
      END

(AN EXAMPLE OF THE DD CARD FØR THE INPUT TAPE WOULD BE--)

//FT10F001 DD UNIT=2400-4,DSN=XSDJH.INPUT,VOL=SER=(tape number),DISP=ØLD,
      LABEL=(1,BLP), DCB=(DEN=3,RECFM=FB,LRECL=1432,BLKSIZE=7160)
```

A standard card form, IBM electro 1000, is available for punching source statements from this form. Instructions for using this form are in any IBM System/360 assembler language reference manual. Address comments concerning the form to IBM Yorktown Laboratory, Publications Development, Box 962, Yorktown Heights, N.Y. 10598.

No of forms per pad may vary slightly.

SPLIT1 – SPLIT3

ASS 8 to Input
9 to OUTPUT
6 to RTE, LPS

```
[redacted]
C     SPLIT3 WAS WRITTEN TO READ CAL OSO-8 PRODUCTION DATA TAPES AND
C       SPLIT OFF THE THREE EXPERIMENTS ONTO 3 OUTPUT TAPES
      DIMENSION IBUF(1790),IOUT(1790)
      REAL*8 C(2)
      EQUIVALENCE (IBUF(1),C(1))
      ASSIGN 100 TO IEOF
      ASSIGN 200 TO IERR
      IREAD = 0
      JFILE = 1
      NFILE = 3
      NOSTAT = 0
      NO1 = 0
      NO2 = 0
      NO3 = 0
      LL = 0
    1 L = 0
      CALL READ (8,IBUF,7160,LEN,IEOF,IERR)
      IREAD = IREAD+5
      IF (IREAD.EQ.5) WRITE(6,600) JFILE,C(1),C(2)
  600 FORMAT (1X,'FOR FILE #',I2,' RECORD #1 DAY #',F8.2,' SECS = '
     *        ,F10.2)
    2 ISTAT = IBUF(L+100)
      IF (ISTAT.NE.0) GO TO 5
      NOSTAT = NOSTAT+1
      GO TO 25
    5 IF ((ISTAT.LT.1).OR.(ISTAT.GT.5)) GO TO 96
      GO TO (10,20,30,30,30),ISTAT
   10 NO1 = NO1+1
      DO 15 I=1,358
      IOUT(I+LL) = IBUF(I+L)
   15 CONTINUE
      LL = LL+358
      IF (LL.LT.1790) GO TO 25
      LL = 0
      CALL WRITE (9,IOUT,7160)
      IWRITE = IWRITE+5
   25 L = L+358
      IF (L.LT.1790) GO TO 2
      GO TO 1
   20 NO2 = NO2+1
      GO TO 25
   30 NO3 = NO3+1
      GO TO 25
   96 WRITE (6,601) JFILE,IREAD,C(1),C(2),IBUF(100)
  601 FORMAT (1X,'BAD STATUS AT FILE #',I2,' REC #',I5,' DAY #',F8.2,' S
     *        EC ',F10.2,' STATUS= ',I10)
      GO TO 25
  100 WRITE (6,602) JFILE,IREAD,C(1),C(2)
  602 FORMAT(' EOF FOR FILE # ',I2,' AFTER REC #',I5,' LAST DAY = ',F8.2
     *        ,' LAST SECS = ',F10.2)
      JFILE = JFILE+1
      IF (JFILE.GT.NFILE) GO TO 50
      IREAD = 0
      GO TO 25
  200 WRITE (6,666) JFILE,IREAD,C(1),C(2)
  666 FORMAT (' BAD READ ON FILE #',I2,' REC #',I5,' DAY #',F8.2,' SECS=
     * ',F10.2)
```

1st + last file
1st + last Rec

```
      GO TO 25
   50 WRITE (6,603) NFILE,NO1,NO2,NO3,NOSTAT,IREAD,IWRITE
  603 FORMAT (' EOF ON INPUT AFTER ',I3,' FILES PROCESSED',/,
     *        1X,I5,' RECORDS FOR STATUS 1',/,
     *        1X,I5,' RECORDS FOR STATUS 2',/,
     *        1X,I5,' RECORDS FOR STATUS 3,4,5',/,
     *        1X,I5,' NO-STATUS RECORDS',/,
     *        1X,I5,' TOTAL RECORDS READ',/,
     *        1X,I5,' TOTAL RECORDS WRITTEN')
      CALL WEOF (9)
      CALL REWD (9)
      CALL REWD (8)
      STOP
      END
POS SPLIT2
LIST
```

```
[redacted]
C     SPLIT3 WAS WRITTEN TO READ CAL OSO-8 PRODUCTION DATA TAPES AND
C        SPLIT OFF THE THREE EXPERIMENTS ONTO 3 OUTPUT TAPES
      DIMENSION IBUF(1790),IOUT(1790)
      REAL*8 C(2)
      EQUIVALENCE (IBUF(1),C(1))
      ASSIGN 100 TO IEOF
      ASSIGN 200 TO IERR
      IREAD = 0
      JFILE = 1
      NFILE = 3
      NOSTAT = 0
      NO1 = 0
      NO2 = 0
      NO3 = 0
      LL = 0
    1 L = 0
      CALL READ (8,IBUF,7160,LEN,IEOF,IERR)
      IREAD = IREAD+5
      IF (IREAD.EQ.5) WRITE(6,600) JFILE,C(1),C(2)
  600 FORMAT (1X,'FOR FILE #',I2,' RECORD #1 DAY #',F8.2,' SECS = '
     *        ,F10.2)
    2 ISTAT = IBUF(L+100)
      IF (ISTAT.NE.0) GO TO 5
      NOSTAT = NOSTAT+1
      GO TO 25
    5 IF ((ISTAT.LT.1).OR.(ISTAT.GT.5)) GO TO 96
      GO TO (10,20,30,30,30),ISTAT
   10 NO1 = NO1+1
      GO TO 25
   25 L = L+358
      IF (L.LT.1790) GO TO 2
      GO TO 1
   20 NO2 = NO2+1
      DO 22 I=1,358
      IOUT(I+LL) = IBUF(I+L)
   22 CONTINUE
      LL = LL+358
      IF (LL.LT.1790) GO TO 25
      LL = 0
      CALL WRITE (9,IOUT,7160)
      IWRITE = IWRITE+5
      GO TO 25
   30 NO3 = NO3+1
      GO TO 25
   96 WRITE (6,601) JFILE,IREAD,C(1),C(2),IBUF(100)
  601 FORMAT (1X,'BAD STATUS AT FILE #',I2,' REC #',I5,' DAY #',F8.2,' S
     *        EC ',F10.2,' STATUS= ',I10)
      GO TO 25
  100 WRITE (6,602) JFILE,IREAD,C(1),C(2)
  602 FORMAT(' EOF FOR FILE # ',I2,' AFTER REC #',I5,' LAST DAY = ',F8.2
     *        ,' LAST SECS = ',F10.2)
      JFILE = JFILE+1
      IF (JFILE.GT.NFILE) GO TO 50
      IREAD = 0
      GO TO 25
  200 WRITE (6,666) JFILE,IREAD,C(1),C(2)
  666 FORMAT (' BAD READ ON FILE #',I2,' REC #',I5,' DAY #',F8.2,' SECS=
```

```
      * ',F10.2)
       GO TO 25
   50 WRITE (6,603) NFILE,NO1,NO2,NO3,NOSTAT,IREAD,IWRITE
  603 FORMAT (' EOF ON INPUT AFTER ',I3,' FILES PROCESSED',/,
      *        1X,I5,' RECORDS FOR STATUS 1',/,
      *        1X,I5,' RECORDS FOR STATUS 2',/,
      *        1X,I5,' RECORDS FOR STATUS 3,4,5'/,
      *        1X,I5,' NO-STATUS RECORDS'/,
      *        1X,I5,' TOTAL RECORDS READ'/,
      *        1X,I5,' TOTAL RECORDS WRITTEN')
       CALL WEOF (9)
       CALL REWD (9)
       CALL REWD (8)
       STOP
       END
POS SPLIT3
LIST
```

```
      PROGRAM SPLIT3
C   SPLIT3 WAS WRITTEN TO READ CAL OSO-8 PRODUCTION DATA TAPES AND
C       SPLIT OFF THE THREE EXPERIMENTS ONTO 3 OUTPUT TAPES
      DIMENSION IBUF(1790),IOUT(1790)
      REAL*8 C(2)
      EQUIVALENCE (IBUF(1),C(1))
      ASSIGN 100 TO IEOF
      ASSIGN 200 TO IERR
      IREAD = 0
      JFILE = 1
      NFILE = 3
      NOSTAT = 0
      NO1 = 0
      NO2 = 0
      NO3 = 0
      LL = 0
    1 L = 0
      CALL READ (8,IBUF,7160,LEN,IEOF,IERR)
      IREAD = IREAD+5
      IF (IREAD.EQ.5) WRITE(6,600) JFILE,C(1),C(2)
  600 FORMAT (1X,'FOR FILE #',I2,' RECORD #1 DAY #',F8.2,' SECS = '
     *        ,F10.2)
    2 ISTAT = IBUF(L+100)
      IF (ISTAT.NE.0) GO TO 5
      NOSTAT = NOSTAT+1
      GO TO 25
    5 IF ((ISTAT.LT.1).OR.(ISTAT.GT.5)) GO TO 96
      GO TO (10,20,30,30,30),ISTAT
   10 NO1 = NO1+1
      GO TO 25
   25 L = L+358
      IF (L.LT.1790) GO TO 2
      GO TO 1
   20 NO2 = NO2+1
      GO TO 25
   30 NO3 = NO3+1
      DO 35 I=1,358
      IOUT(I+LL) = IBUF(I+L)
   35 CONTINUE
      LL = LL+358
      IF (LL.LT.1790) GO TO 25
      LL = 0
      CALL WRITE (9,IOUT,7160)
      IWRITE = IWRITE+5
      GO TO 25
   96 WRITE (6,601) JFILE,IREAD,C(1),C(2),IBUF(100)
  601 FORMAT (1X,'BAD STATUS AT FILE #',I2,' REC #',I5,' DAY #',F8.2,' S
     *        EC ',F10.2,' STATUS= ',I10)
      GO TO 25
  100 WRITE (6,602) JFILE,IREAD,C(1),C(2)
  602 FORMAT(' EOF FOR FILE # ',I2,' AFTER REC #',I5,' LAST DAY = ',F8.2
     *        ,' LAST SECS = ',F10.2)
      JFILE = JFILE+1
      IF (JFILE.GT.NFILE) GO TO 50
      IREAD = 0
      GO TO 25
  200 WRITE (6,666) JFILE,IREAD,C(1),C(2)
  666 FORMAT (' BAD READ ON FILE #',I2,' REC #',I5,' DAY #',F8.2,' SECS=
```

```
      *  ',F10.2)
       GO TO 25
   50 WRITE (6,603) NFILE,NO1,NO2,NO3,NOSTAT,IREAD,IWRITE
  603 FORMAT (' EOF ON INPUT AFTER ',I3,' FILES PROCESSED',/,
      *        1X,I5,' RECORDS FOR STATUS 1',/,
      *        1X,I5,' RECORDS FOR STATUS 2',/,
      *        1X,I5,' RECORDS FOR STATUS 3,4,5',/,
      *        1X,I5,' NO-STATUS RECORDS',/,
      *        1X,I5,' TOTAL RECORDS READ',/,
      *        1X,I5,' TOTAL RECORDS WRITTEN')
       CALL WEOF (9)
       CALL REWD (9)
       CALL REWD (8)
       STOP
       END
EXI
$SPOOLEND
$NOTE  /$$$$$/     /END OF OUTPUT/     /$$$$$/  15:31:20
$ASS LO NO
$$
```

```
$END DO
$ASS 6 LPS
$EXE SPLIT ULM
FOR FILE # 1 RECORD #     5 DAY # 1568.00 SECS =  328360.57
EOF FOR FILE #  1 AFTER REC # 7225 LAST DAY =     0.00 LAST SECS =  325931.24
FOR FILE # 2 RECORD #     5 DAY # 1688.00 SECS =  326406.35
EOF FOR FILE #  2 AFTER REC # 7475 LAST DAY =     0.00 LAST SECS =  324611.30
FOR FILE # 3 RECORD #     5 DAY # 4632.00 SECS =  325220.58
EOF FOR FILE #  3 AFTER REC # 7350 LAST DAY =     0.00 LAST SECS =  322849.34
EOF ON INPUT AFTER    3 FILES PROCESSED
22050 RECORDS FOR STATUS 1
    0 RECORDS FOR STATUS 2
    0 RECORDS FOR STATUS 3,4,5
    8 NO-STATUS RECORDS

$SPOOLEND
$NOTE  /$$$$$/      /END OF OUTPUT/      /$$$$$/  14:06:34
$ASS LO NO
```

$$

DUMP OF TAPE MIN-8

7/24/75-7/27/75

INPUT TAPE MIN-8 ON MS1
DATA INPUT H9 NF 3 FL 1 1 1 SR 3 1 1 SR 3 LAST 1

D-31192

FILE 1 RECORD 1 LENGTH 7160BYTES

(0)	42CD0000	00000000	4515178F	CB7CE353	41618FD6	50000000	4110D816	20000000	4122461A	30000000
(40)	405F7FF9	C0000000	41225D30	00000000	4060042E	00000000	00000000	00000000	45E18600	00000000
(80)	BF6FB6A0	B0000000	413DBAD5	00000000	4056F453	00000000	C0FFF972	474538EF	C0FFF972	474538EF
(120)	441700C4	FD8ED4A3	C37A5C25	AE98B05D	C3BF5691	C6A3681A	43221BB3	D54AEE20	4042BDE5	BAF75465
(160)	4314C27C	06281A31	4130FC74	874616AE	4227A0A4	0927428B	C21A7F3C	0B618104	419F547A	00000000
(200)	00000F00	00000124	00000124	00000124	00000124	000002DF	000002DF	00000004	00000001	
(240)	00000000	00000001	000000D2	00000000	00000000	00000000	00000000	00000000	00000000	00000000
(280)	00000012	00000012	00000010	00000010	00000012	0000001D	000001F2	0000010C	000001F3	000000DE
(320)	000001EC	0000035C	00000097	00000121	00000001	00000000	00000007	00000000	00000000	00000000
(360)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	00000001	000000D2	00000001	00000000	00000001
(400)	00000000	00000007	00015A29	00021E09	00038B0A	0008A32C	0008E667	0009653B	000B9656	77777777
(440)	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777
(480)	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777
(520)	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777
(560)	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777
(600)	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777
(640)	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777
(680)	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777
(720)	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777
(760)	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777
(800)	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777
(840)	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777
(880)	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777
(920)	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777
(960)	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777
(1000)	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777
(1040)	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777
(1080)	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777
(1120)	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777
(1160)	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777
(1200)	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777
(1240)	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777
(1280)	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777
(1320)	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777
(1360)	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777
(1400)	77777777	77777777	77777777	77777777	77777777	77777777	77777777	77777777	42CE0000	00000000
(1440)	412F1FF6	E3530000	41618FD6	50000000	4110D816	20000000	4122461A	30000000	405F7FF9	C0000000
(1480)	41225D49	00000000	406004C1	00000000	00000000	00000000	45E18600	00000000	BF6FB6A0	B0000000
(1520)	413DBAEE	00000000	4056F3B4	00000000	C0FFF972	474538EF	C0FFF972	474538EF	4417219F	3026FBB6
(1560)	C3766B36	83445879	C3BDD7BB	A63960DD	43221A35	36F7C6C8	4043195B	BE94CC24	4314CAE3	DDFE0865
(1600)	413BB23F	D5EAF389	422828E7	F1EF5C2E	C21A2CE7	93545F5D	419F546F	00000000	00000F00	00000124
(1640)	00000124	00000124	00000124	00000124	000002DF	000002DF	00000004	00000001	00000000	00000001
(1680)	000000D2	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000012	00000012
(1720)	00000010	00000010	00000012	0000001D	000001F2	0000010C	000001F3	000000DE	000001EC	0000035C
(1760)	00000097	00000121	00000001	00000000	00000007	00000000	00000000	00000000	FFFFFFFF	FFFFFFFF
(1800)	FFFFFFFF	FFFFFFFF	FFFFFFFF	00000001	000000D2	00000001	00000000	00000001	00000000	00000004
(1840)	00006D35	0001FE3B	00023F38	000BE417	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(1880)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(1920)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(1960)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(2000)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(2040)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(2080)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(2120)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(2160)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(2200)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(2240)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(2280)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF

(5160)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(5200)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(5240)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(5280)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(5320)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(5360)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(5400)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(5440)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(5480)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(5520)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(5560)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(5600)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(5640)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(5680)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(5720)	FFFFFFFF	FFFFFFFF	42CF0000	00000000	4514F471	C53E9168	40356AEE	98000000	40E7BE76	80000000
(5760)	41234B40	40000000	40505F62	30000000	412366DE	00000000	40517C27	00000000	00000000	00000000
(5800)	45801680	00000000	3F81E4F4	40000000	413FDFB2	00000000	408FB319	00000000	C0FFF972	474538EF
(5840)	C0FFF972	474538EF	441A0AEC	684F6906	C325303A	1AD9A7F1	C36D60E5	056007F4	4321D9F7	00B4B66F
(5880)	4047E0A3	08BEEE9A	4315A464	1A2E3CF0	421BC032	A74FE256	42353B09	91048E76	C1E3F58F	F97DF329
(5920)	419EA34F	00000000	00000F00	00000124	00000124	00000124	00000124	00000124	000002DF	000002DF
(5960)	00000004	00000001	00000000	00000001	000000D2	00000000	00000000	00000000	00000000	00000000
(6000)	00000000	00000000	00000012	00000012	0000000F	00000010	00000012	0000001C	000001F2	0000010E
(6040)	000001F3	000000DE	000001EC	00000361	00000097	00000121	00000001	00000000	00000007	00000000
(6080)	00000000	00000000	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	00000001	000000D2	00000001
(6120)	00000000	00000001	00000000	00000000	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(6160)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(6200)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(6240)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(6280)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(6320)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(6360)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(6400)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(6440)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(6480)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(6520)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(6560)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(6600)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(6640)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(6680)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(6720)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(6760)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(6800)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(6840)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(6880)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(6920)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(6960)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(7000)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(7040)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(7080)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF
(7120)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF

FILE 3 RECORD 1397 LENGTH 5728BYTES

(0)	42D00000	00000000	4514D198	0EEEF9DA	405095AF	B0000000	40D11185	C0000000	4123AE6E	A8000000
(40)	404EA47E	F0000000	4123BF8D	00000000	404F7FF7	00000000	412605DC	10000000	45FFE000	00000000
(80)	3F9C4804	80000000	4140E75E	00000000	40A8D0E0	00000000	C0FFF972	474538EF	C0FFF972	474538EF
(120)	44190090	528D610A	C36B99CE	D78D1FBF	C37C00C6	0C461875	4321E3DC	F429BE3F	404499F2	22C0AA7A
(160)	4315115B	42E0B6EE	42169956	A5A7D4DB	422C9CFE	F446C9A4	C210C39C	EBD3D965	419E9F57	00000000
(200)	000000FF	00000344	0000032A	00000314	00000341	0000031D	00000124	00000124	00000003	00000002
(240)	00000000	00000096	00000013	00000000	00000000	00000000	00000000	00000000	00000000	00000000
(280)	00000012	00000013	0000000F	00000012	00000013	0000001D	000001F2	0000012F	000001F3	000000E8
(320)	000001EA	000000BD	00000097	0000011C	00000001	00000000	00000007	00000000	00000000	00000000
(360)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	00000001	00000013	00000096	00000000	00000002
(400)	00000000	0000003E	00003411	0000BDF6	0000FCB1	00098095	00099710	00099819	0009ACAF	0009B7D2
(440)	0009C547	0009DD0E	0009F014	000A0C84	000A1C0D	000A299A	000A2B5A	000A2E8D	000A3492	000A4287
(480)	000A4376	000A4993	000A550A	000A5C77	000A5F7A	000A631C	000A6D66	000A758E	000A904D	000A9A9A
(520)	000AA154	000AA649	000AB452	000AB685	000AC56C	000ACDF1	000AD98F	000AD9B1	000ADC9B	000ADDD1

DUMP OF TAPE T-5724

D-31257

DATA INPUT H9 NF 3 FL 1 1 1 SR 3 1 SR 3 LAST

9/19/75 – 9/22/75

(0)	43106000	00000000	4314F563	36CB4393	4117DDC3	70000000	407085CE	20000000	413136CC	20000000
(40)	3F750ADC	90000000	412A252F	00000000	C0B93BD8	00000000	00000000	00000000	457D8600	00000000
(120)	C32D52F6	25A2C5AA	C4180C9D	B2434221	C3C45C23	9F38CDD0	4322D7B3	252AED23	403CE391	D3D4F244
(160)	4311669B	0254A5A8	421F3DBE	ACF57AAE	C25AC7FC	9EBDA5FC	C21B20C1	A36AA972	41A90873	00000000
(240)	00000000	000000CB	0000008B	00000001	00000001	00000000	00000000	00000000	00000000	00000000
(280)	00000014	00000013	00000010	00000014	00000013	0000001E	FFFFFFFF	0000010E	000001F3	000000E0
(360)	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	FFFFFFFF	00000001	0000008B	000000CB	00000001	00000003
(400)	00000032	00000200	0000FFFF	0000FFFF	0000FFFF	0000FFFF	0000FFFF	0000FFFF	0000FFFF	0000FFFF
(480)	0000FFFF	0000FFFF	0000FFFF	0000FFFF	0000FFFF	0000FFFF	0000FFFF	00000000	00000101	00000001
(520)	00000000	00000000	00000000	00000101	00000001	00000000	00000001	00000000	00000000	00000000
(600)	00000000	00000100	00000000	00000000	00000000	00000000	00000100	00000000	00000100	00000001
(640)	00000000	00000001	00000001	00000101	00000000	00000000	00000000	00000000	00000101	00000000
(720)	00000201	00000000	00000101	00000000	00000000	00000100	00000000	00000000	00000100	00000000
(760)	00000100	00000000	00000001	00000000	00000100	00000200	00000200	00000101	00000000	00000000
(840)	00000101	00000000	00000100	00000000	00000200	00000101	00000001	00000000	00000100	00000101
(880)	00000000	00000100	00000000	00000000	00000000	00000200	00000000	00000000	00000001	00000002
(960)	00000201	00000102	00000104	00000000	00000202	00000000	00000101	00000200	00000201	00000301
(1000)	00000003	00000202	00000101	00000003	00000101	00000001	00000203	00000103	00000201	00000102
(1080)	00000202	00000505	00000105	00000101	00000202	00000102	00000001	00000100	00000102	00000000
(1120)	00000101	00000202	00000002	00000002	00000204	00000001	00000101	00000200	00000101	00000000
(1200)	00000001	00000000	00000002	00000200	00000002	00000101	00000200	00000003	00000001	00000201
(1240)	00000101	00000003	00000002	00000000	00000001	00000101	00000101	00000301	00000202	00000203
(1320)	00000102	00000000	00000201	00000000	00000202	00000005	00000401	00000400	0000060C	00001212
(1360)	0000181A	00001915	00001107	00000805	00000602	00000604	00000603	00000503	00000604	00000406
(1440)	4514F60E	3F334395	4117DDC3	70000000	407085CE	20000000	413136CC	28000000	3F750ADC	90000000
(1480)	412A2516	00000000	C0B93D5D	00000000	00000000	00000000	457D8600	00000000	BEC25E89	20000000
(1560)	C4180666	5EBE6873	C3C5D310	78609C40	4322DA41	2DFE3DE4	403CBBE5	FCD3CD75	43117056	2375A9D1
(1600)	421EE796	80003834	C25A2356	51EC7DED	C21B5DD2	8C22102B	41A908A0	00000000	000000F0	0000036F
(1680)	0000008B	00000001	00000001	0000009C	00000510	FFFFFFFF	000008D2	00000000	00000014	00000013
(1720)	00000010	00000014	00000013	0000001E	FFFFFFFF	0000010E	000001F3	000000E0	000001EA	00000356
(1800)	FFFFFFFF	FFFFFFFF	FFFFFFFF	00000001	0000008B	000000CB	00000001	00000003	00000000	00000200
(1840)	00000100	00000000	00000000	00000000	00000101	00000001	00000001	00000000	00000101	00000002
(1920)	00000000	00000100	00000100	00000100	00000000	00000001	00000000	00000100	00000000	00000000
(1960)	00000001	00000002	00000000	00000000	00000000	00000000	00000000	00000000	00000000	00000000
(2040)	00000000	00000000	00000000	00000000	00000001	00000000	00000000	00000101	00000000	00000000
(2080)	00000000	00000100	00000000	00000000	00000100	00000000	00000000	00000000	00000000	00000003
(2160)	00000001	00000001	00000001	00000000	00000001	00000000	00000000	00000001	00000000	00000100
(2200)	00000002	00000001	00000100	00000000	00000100	00000000	00000001	00000000	00000100	00000002
(2280)	00000000	00000000	00000300	00000100	00000000	00000100	00000000	00000100	00000003	00000101

```
( 6360)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF

( 6440)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF
( 6480)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF

( 6560)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF
( 6600)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF

( 6680)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF
( 6720)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF

( 6800)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF
( 6840)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF

( 6920)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF
( 6960)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF

( 7040)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF
( 7080)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF

FILE      3  RECORD    1427  LENGTH    4296BYTES

(   40)  3E18A327  38000000  412BFE9C  00000000  C0C6EED3  00000000  00000000  00000000  457DC200  00000000
(   80)  3EC456E6  C0000000  4136D2F2  00000000  40AB9414  00000000  C0FFF972  474538EF  C0FFF972  474538EF

(  160)  43112D70  D66A0C22  4215C7CE  34DBAE80  C25BAF5B  80CE0C93  C21F0926  089FD6DC  41AC2D70  00000000
(  200)  00000F00  00000124  00000124  00000124  00000124  00000124  000002DF  000002EC  00000004  00000001

(  280)  00000014  00000012  00000011  00000011  00000013  0000001D  000001E2  0000010E  000001F3  000000DE
(  320)  000001EC  0000034A  00000097  00000121  00000001  00000000  00000007  00000000  00000000  00000000

(  400)  00000000  00000006  00002108  00002A6A  00071F48  00090970  0009712B  000B3159  FFFFFFFF  FFFFFFFF
(  440)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF

(  520)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF
(  560)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF

(  640)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF
(  680)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF

(  760)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF
(  800)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF

(  880)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF
(  920)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF

( 1000)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF
( 1040)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF

( 1120)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF
( 1160)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF

( 1240)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF
( 1280)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF

( 1360)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF
( 1400)  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  FFFFFFFF  43109000  00000000

( 1480)  412BFEA2  00000000  C0C6EF1F  00000000  00000000  00000000  457DC200  00000000  3EC42708  20000000
( 1520)  4136D2ED  00000000  40AB9420  00000000  C0FFF972  474538EF  C0FFF972  474538EF  C32EFBCC  FDD52E3F

( 1600)  4215AC63  58EA4836  C25AF889  AA31110B  C21F3204  22D4AB8C  41AC2D6F  00000000  00000F00  00000124
( 1640)  00000124  00000124  00000124  00000124  000002DF  000002EC  00000004  00000001  00000000  00000001

( 1720)  00000011  00000011  00000013  0000001E  000001F2  0000010E  000001F3  000000DF  000001EC  0000034A
```