

เอกสารอ้างอิง

1. Diesendorf, W. Insulation Coordination in High Voltage Electric Power Systems, Butterworth & Company LTD., 1974.
2. ANSI C62.2-1981, Guide for the Application of Valve-type Surge Arresters for Alternating-Current Systems, American National Standards Institute, pp.39-41, New York, 1981.
3. Bewley L.V., Traveling Waves on Transmission System, New York, 2 nd ed., 1951.
4. Truax, C.J., J.D. Brown, W. Neugebauer, "TNA Study of Reclosing Transients on a 765 KV Shunt Compensated Transmission Line," IEEE Transactions on Power Apparatus and System, Vol.97, pp. 1447-1457, 1978.
5. Dommel, H.W. "Digital Computer Solution of Electromagnetic Transient in Single and Multiphase Networks," IEEE Transactions on Power Apparatus and Systems, Vol.88, pp.388-399, 1969.
6. Dommel, H.W. and W.S.Meyer, "Computation of Electromagnetic Transient," Proceeding of IEEE, Vol.62, pp.983-993, 1974.
7. Humpage, W.D. and K.Wong, "Electromagnetic Transient Analysis in EHV Power Networks," Proceeding of IEEE, Vol.70, pp.1982.
8. Dommel, H.W. "Nonlinear and Time-varying Elements in Digital Simulation of Electromagnetic Transients," IEEE Transactions on Power Apparatus and Systems, Vol.90, pp.2561-2567, 1971.
9. Hesse, M.H. "Electromagnetic and Electrostatic Transmission Line Parameters by Digital Computer," IEEE Transactions on Power Apparatus and Systems, Vol.82, pp.282-291, 1963.
10. Carson, J.R. "Wave Propagation in Overhead Wires with Ground Return," Bell System Technical Journal, Vol.5, pp.539-554, 1926.

11. Stevenson, W.D. Elements of Power System Analysis, McGraw-Hill, New York, 2nd ed., 1962.
12. Tinney, W.F. and J.W. Walker, "Direct Solutions of Sparse Network Equations by Optimally Ordered Triangular Factorization," Proceeding of IEEE, Vol.55, pp. 1801-1809, 1967.
13. Tinney, W.F. and W.S.Meyer, "Solution of Large Sparse Systems by Ordered Triangular Factorization, "IEEE Transactions on Automatic Control, Vol. 18, 1973.
14. Tinney, W.F. "Compensation Methods for Network Solutions by Optimally Ordered Triangular Factorization, "IEEE Transactions on Power Apparatus and Systems, Vol.91, pp.123-127, 1972.
15. Ardito, A. G.Santagostino, "A Review of Digital and Analog Methods of Calculation of Overvoltages in Electric Systems, "CIGRE Study Committee 33, 1985.
16. Geddy, D.K. "Modelling of Power System Components II, "Residential School in Power System Electrical Engineering, Sydney, 1986.
17. Bonneville Power Administration, Electromagnetic Transients Program Rule Book, Oregon, 1982.
18. Alessandro Clerici and Leonardo Marzio, "Coordinated Use of TNA and Digital Computer for Switching-Surge Studies : Transient Equivalent of a Complex Network, "IEEE Transactions on Power Apparatus and Systems, Vol.89, pp.1717-1726, 1970.
19. Morched, A.S. and V.Brandwajn, "Transmission Network Equivalents for Electromagnetic Transient Studies, "IEEE Transactions on Power Apparatus and Systems, Vol.102, pp.2984-2994, 1983.
20. Dumrongittigule, S. Feeding Network Representation in Switching Overvoltage Calculation, Doctoral Dissertation, University of Trondheim, Norway, 1983.

21. Greenwood, Electrical Transients in Power Systems, Wiley, New York, 1971.
22. Mork, B., K.S.Rao and D.L. Stuchm, "Modelling Ferroresonance with EMTP, "EMTP Newsletter, Vol.3, pp.2-7, May, 1983.

ภาคผนวก

ตาราง ก.1

CHARACTERISTICS OF COPPER CONDUCTORS, HARD-DRAWN, 97.3% CONDUCTIVITY*



Size of conductor		Number of strands	Diameter of individual strands, in.	Outside diameter, in.	Breaking strength, lb	Weight, lb/mile	Approx. current-carrying capacity, amp	Geometric mean radius at 60 cycles, ft	Resistance, ohms/conductor/mile								Inductive reactance, ohms/conductor/mile at 1-ft spacing			Shunt capacitive reactance, megohms/conductor/mile at 1-ft spacing					
Circular mils	A.W.G. or B. & S.								25°C (77°F)				50°C (122°F)				25 cycles	50 cycles	60 cycles	25 cycles	50 cycles	60 cycles	25 cycles	50 cycles	60 cycles
									d-c	25 cycles	50 cycles	60 cycles	d-c	25 cycles	50 cycles	60 cycles									
1,000,000	...	37	0.1644	1.151	43,830	16,300	1,300	0.0368	0.0585	0.0594	0.0620	0.0634	0.0640	0.0648	0.0672	0.0685	0.1666	0.333	0.400	0.216	0.1081	0.0901			
900,000	...	37	.1560	1.092	39,510	14,670	1,220	.0349	.0650	.0658	.0682	.0695	.0711	.0718	.0740	.0752	.1693	.339	.406	.220	.1100	.0916			
800,000	...	37	.1470	1.029	35,120	13,040	1,130	.0329	.0731	.0739	.0760	.0772	.0800	.0806	.0826	.0837	.1722	.344	.413	.224	.1121	.0934			
750,000	...	37	.1424	0.997	33,400	12,230	1,090	.0319	.0780	.0787	.0807	.0818	.0853	.0859	.0878	.0888	.1739	.348	.417	.226	.1132	.0943			
700,000	...	37	.1375	.963	31,170	11,410	1,040	.0308	.0836	.0842	.0861	.0871	.0914	.0920	.0937	.0947	.1759	.352	.422	.229	.1145	.0954			
600,000	...	37	.1273	.891	27,020	9,781	940	.0285	.0975	.0981	.0997	.1006	.1066	.1071	.1086	.1095	.1799	.360	.432	.235	.1173	.0977			
500,000	...	37	.1162	.814	22,510	8,151	840	.0260	.1170	.1175	.1188	.1196	.1280	.1283	.1296	.1303	.1845	.369	.443	.241	.1205	.1004			
500,000	...	19	.1622	.811	21,590	8,151	840	.0256	.1170	.1175	.1188	.1196	.1280	.1283	.1296	.1303	.1853	.371	.445	.241	.1206	.1005			
450,000	...	19	.1539	.770	19,750	7,336	780	.0243	.1300	.1304	.1316	.1323	.1422	.1426	.1437	.1443	.1879	.376	.451	.245	.1224	.1020			
400,000	...	19	.1451	.726	17,560	6,521	730	.0229	.1462	.1466	.1477	.1484	.1600	.1603	.1613	.1619	.1909	.382	.458	.249	.1245	.1038			
350,000	...	19	.1357	.679	15,590	5,706	670	.0214	.1671	.1675	.1684	.1690	.1828	.1831	.1840	.1845	.1943	.389	.466	.254	.1269	.1058			
350,000	...	12	.1708	.710	15,140	5,706	670	.0225	.1671	.1675	.1684	.1690	.1828	.1831	.1840	.1845	.1918	.384	.460	.251	.1253	.1044			
300,000	...	19	.1257	.629	13,510	4,891	610	.01987	.1950	.1953	.1961	.1966	.213	.214	.214	.215	.1982	.396	.476	.259	.1296	.1080			
300,000	...	12	.1681	.657	13,170	4,891	610	.0208	.1950	.1953	.1961	.1966	.213	.214	.214	.215	.1957	.392	.470	.256	.1281	.1068			
250,000	...	19	.1147	.574	11,360	4,076	540	.01813	.234	.234	.235	.235	.256	.256	.257	.257	.203	.406	.487	.266	.1329	.1108			
250,000	...	12	.1443	.600	11,130	4,076	540	.01902	.234	.234	.235	.235	.256	.256	.257	.257	.200	.401	.481	.263	.1313	.1094			
211,600	4/0	19	.1055	.528	9,617	3,450	480	.01668	.276	.277	.277	.278	.302	.303	.303	.303	.207	.414	.497	.272	.1359	.1132			
211,600	4/0	12	.1328	.552	9,483	3,450	490	.01750	.276	.277	.277	.278	.302	.303	.303	.303	.205	.409	.491	.269	.1343	.1119			
211,600	4/0	7	.1739	.522	9,154	3,450	480	.01579	.276	.277	.277	.278	.302	.303	.303	.303	.210	.420	.503	.273	.1363	.1136			
167,800	3/0	12	.1183	.492	7,556	2,736	420	.01559	.349	.349	.349	.350	.381	.381	.382	.382	.210	.421	.505	.277	.1384	.1153			

ลักษณะของสายส่งไฟฟ้า

การคำนวณ ก

ตาราง ก.2 CHARACTERISTICS OF COPPER CONDUCTORS, HARD-DRAWN, 97.3% CONDUCTIVITY * (Continued)

Size of conductor		Number of strands	Diameter of individual strands, in.	Outside diameter, in.	Breaking strength, lb	Weight, lb/mile	Approx. current-carrying capacity, † amp	Geometric mean radius at 60 cycles, ft	Resistance, ohms/conductor/mile								Inductive reactance, ohms/conductor/mile at 1-ft spacing			Shunt capacitive reactance, megohms/conductor/mile at 1-ft spacing					
Circular mils	A.W.G. or B. & S.								25°C (77°F)				50°C (122°F)				25 cycles	50 cycles	60 cycles	25 cycles	50 cycles	60 cycles	25 cycles	50 cycles	60 cycles
									d-c	25 cycles	50 cycles	60 cycles	d-c	25 cycles	50 cycles	60 cycles									
167,800	3/0	7	.1548	.464	7,366	2,736	420	.01404	.349	.349	.349	.350	.381	.381	.382	.382	.216	.431	.518	.281	1.405	.1171			
133,100	2/0	7	.1379	.414	5,926	2,170	360	.01252	.440	.440	.440	.440	.481	.481	.481	.481	.222	.443	.532	.289	1.445	.1205			
105,500	1/0	7	.1228	.368	4,752	1,720	310	.01113	.555	.555	.555	.555	.606	.607	.607	.607	.227	.455	.546	.298	1.488	.1240			
83,690	1	7	.1093	.328	3,804	1,364	270	.00992	.699	.699	.699	.699	.765				.233	.467	.560	.306	1.528	.1274			
83,690	1	3	.1670	.360	3,620	1,351	270	.01016	.692	.692	.692	.692	.757				.232	.464	.557	.299	1.495	.1246			
66,370	2	7	.0974	.292	3,045	1,082	230	.00883	.881	.882	.882	.882	.964				.239	.478	.574	.314	1.570	.1308			
66,370	2	3	.1487	.320	2,913	1,071	240	.00903	.873				.955				.238	.476	.571	.307	1.537	.1281			
66,370	2	1258	3,003	1,061	220	.00836	.864				.945				.242	.484	.581	.323	1.614	.1345			
52,630	3	7	.0867	.260	2,433	858	200	.00787	1.112				1.216				.245	.490	.588	.322	1.611	.1343			
52,630	3	3	.1325	.285	2,359	850	200	.00805	1.101				1.204				.244	.488	.585	.316	1.578	.1315			
52,630	3	1229	2,439	841	190	.00745	1.090	Same as d-c			1.192	Same as d-c			.248	.496	.595	.331	1.656	.1380			
41,740	4	3	.1180	.254	1,879	674	180	.00717	1.338				1.518				.250	.499	.599	.324	1.619	.1349			
41,740	4	1204	1,970	667	170	.00663	1.374				1.503				.254	.507	.609	.339	1.697	.1415			
33,100	5	3	.1050	.226	1,505	534	150	.00638	1.750				1.914				.256	.511	.613	.332	1.661	.1384			
33,100	5	11819	1,591	529	140	.00590	1.733				1.895				.260	.519	.623	.348	1.738	.1449			
26,250	6	3	.0935	.201	1,205	424	130	.00568	2.21				2.41				.262	.523	.628	.341	1.703	.1419			
26,250	6	11620	1,290	420	120	.00526	2.18				2.39				.265	.531	.637	.356	1.779	.1423			
20,920	7	11443	1,030	333	110	.00468	2.75				3.01				.271	.542	.651	.364	1.821	.1517			
16,510	8	11285	826	264	90	.00417	3.47				3.80				.277	.554	.665	.372	1.862	.1552			

* Republished by permission of the Westinghouse Electric Corporation from "Electrical Transmission and Distribution Reference Book."

† For conductor at 75°C, air at 25°C, wind 1.4 miles/hour (2 ft/sec), frequency = 60 cycles.

CHARACTERISTICS OF ALUMINUM CABLE, STEEL-REINFORCED*
(Aluminum Company of America)



Multilayer Conductors

Circular mils or A.W.G. aluminum	Aluminum		Steel		Outside diameter, in.	Copper equiva- lent, † circular mils or A.W.G.	Ultimate strength, lb	Weight, lb/mile	Geo- metric mean radius at 60 cycles, ft	Ap- prox. current- carry- ing capac- ity, ‡ amp	Resistance, ohms/conductor/mile								Inductive reactance, ohms/conductor/ mile at 1-ft spacing all currents			Shunt capacitive reactance, megohms/ conductor/mile at 1-ft spacing			
	Strands	Layers	Strand diam- eter, in.	Strands							Strand diam- eter, in.	25°C (77°F) Small currents				50°C (122°F) Current approx. 75% capacity ‡				25 cycles	50 cycles	60 cycles	25 cycles	50 cycles	60 cycles
												d-c	25 cycles	50 cycles	60 cycles	d-c	25 cycles	50 cycles	60 cycles						
	1,590,000	54	3	0.1716							19	0.1030	1,545	1,000,000	56,000	10,777	0.0520	1,380	0.0587	0.0588	0.0590	0.0591	0.0646	0.0656	0.0675
1,510,500	54	3	.1673	19	.1004	1,506	950,000	53,200	10,237	.0507	1,340	.0618	.0619	.0621	.0622	.0680	.0690	.0710	.0720	.1508	.302	.362	.1971	.0986	.0821
1,431,000	54	3	.1628	19	.0977	1,465	900,000	50,400	9,699	.0493	1,300	.0652	.0653	.0655	.0656	.0718	.0729	.0749	.0760	.1522	.304	.365	.1991	.0996	.0830
1,351,000	54	3	.1582	19	.0949	1,424	850,000	47,600	9,160	.0479	1,250	.0691	.0692	.0694	.0695	.0761	.0771	.0792	.0803	.1536	.307	.369	.201	.1006	.0838
1,272,000	54	3	.1535	19	.0921	1,382	800,000	44,800	8,621	.0465	1,200	.0734	.0735	.0737	.0738	.0808	.0819	.0840	.0851	.1551	.310	.372	.203	.1016	.0847
1,192,500	54	3	.1486	19	.0892	1,338	750,000	43,100	8,082	.0450	1,160	.0783	.0784	.0786	.0788	.0862	.0872	.0894	.0906	.1568	.314	.376	.206	.1026	.0857
1,113,000	54	3	.1436	19	.0862	1,293	700,000	40,200	7,544	.0435	1,110	.0839	.0840	.0842	.0844	.0924	.0935	.0957	.0969	.1585	.317	.380	.208	.1040	.0867
1,033,500	54	3	.1384	7	.1384	1,246	650,000	37,100	7,019	.0420	1,060	.0903	.0905	.0907	.0909	.0994	.1005	.1025	.1035	.1603	.321	.385	.211	.1053	.0878
954,000	54	3	.1329	7	.1329	1,196	600,000	34,200	6,479	.0403	1,010	.0979	.0980	.0981	.0982	.1078	.1088	.1118	.1128	.1624	.325	.390	.214	.1068	.0890
900,000	54	3	.1291	7	.1291	1,162	566,000	32,300	6,112	.0391	970	.104	.104	.104	.104	.1145	.1155	.1185	.1185	.1639	.328	.393	.216	.1078	.0898
874,500	54	3	.1273	7	.1273	1,146	550,000	31,400	5,940	.0386	950	.107	.107	.107	.108	.1178	.1188	.1218	.1228	.1646	.329	.395	.217	.1083	.0903
795,000	54	3	.1214	7	.1214	1,093	500,000	28,500	5,399	.0368	900	.117	.118	.118	.119	.1288	.1308	.1358	.1378	.1670	.334	.401	.220	.1100	.0917
795,000	26	2	.1749	7	.1360	1,108	500,000	31,200	5,770	.0375	900	.117	.117	.117	.117	.1288	.1288	.1288	.1288	.1660	.332	.399	.219	.1095	.0912
795,000	30	3	.1628	19	.0977	1,140	500,000	38,400	6,517	.0393	910	.117	.117	.117	.117	.1288	.1288	.1288	.1288	.1637	.327	.393	.217	.1055	.0904
715,500	54	3	.1151	7	.1151	1,036	450,000	26,300	4,859	.0349	830	.131	.131	.131	.132	.1442	.1452	.1472	.1482	.1697	.339	.407	.224	.1119	.0932
715,500	26	2	.1659	7	.1290	1,051	450,000	28,100	5,193	.0355	840	.131	.131	.131	.131	.1442	.1442	.1442	.1442	.1657	.337	.405	.223	.1114	.0928
715,500	30	2	.1544	19	.0926	1,081	450,000	34,600	5,865	.0372	840	.131	.131	.131	.131	.1442	.1442	.1442	.1442	.1664	.333	.399	.221	.1104	.0920
666,600	54	3	.1111	7	.1111	1,000	419,000	24,500	4,527	.0337	800	.140	.140	.141	.141	.1541	.1571	.1601	.1715	.343	.412	.226	.1132	.0943	
636,000	54	3	.1085	7	.1085	0.977	400,000	23,600	4,319	.0329	770	.147	.147	.148	.148	.1618	.1638	.1678	.1688	.1726	.345	.414	.228	.1140	.0950
636,000	26	2	.1564	7	.1216	.990	400,000	25,000	4,616	.0335	780	.147	.147	.147	.147	.1618	.1618	.1618	.1618	.1718	.344	.412	.227	.1135	.0946
636,000	30	2	.1456	19	.0874	1.019	400,000	31,500	5,213	.0351	780	.147	.147	.147	.147	.1618	.1618	.1618	.1618	.1693	.339	.406	.225	.1125	.0937
605,000	54	3	.1059	7	.1059	0.953	380,500	22,500	4,109	.0321	750	.154	.155	.155	.155	.1695	.1715	.1755	.1775	.1739	.348	.417	.230	.1149	.0957
605,000	26	2	.1525	7	.1186	.966	380,500	24,100	4,391	.0327	760	.154	.154	.154	.154	.1700	.1720	.1720	.1720	.1730	.346	.415	.229	.1144	.0953
556,500	26	2	.1463	7	.1138	.927	350,000	22,400	4,039	.0313	730	.168	.168	.168	.168	.1849	.1859	.1859	.1859	.1751	.350	.420	.232	.1159	.0965

ตาราง ก.4 CHARACTERISTICS OF ALUMINUM CABLE, STEEL-REINFORCED * (Continued)

Circular mils or A.W.G. aluminum	Aluminum		Steel			Copper equiva- lent, † circular mils or A.W.G.	Ultimate strength, lb	Weight, lb/mile	Geo- metric mean radius at 60 cycles, ft	Ap- prox. current- carry- ing capac- ity, ‡ amp	Resistance, ohms/conductor/mile						Inductive reactance, ohms/conductor/ mile at 1-ft spacing all currents			Shunt capacitive reactance, megohms/ conductor/mile at 1-ft spacing					
	Strands	Layers	Strand diam- eter, in.	Strands	Strand diam- eter, in.						Outside diameter, in.	25°C (77°F) Small currents			50°C (122°F) Current approx. 75% capacity ‡			25 cycles	50 cycles	60 cycles	25 cycles	50 cycles	60 cycles		
												d-c	25 cycles	50 cycles	60 cycles	d-c	25 cycles							50 cycles	60 cycles
												Same as d-c			Same as d-c										
556,500	30	2	.1362	7	.1362	.953	350,000	27,200	4,588	.0328	730	.168	.168	.168	.168	.1849	.1859	.1859	.1859	.1728	.346	.415	.230	.1149	.0957
500,000	30	2	.1291	7	.1291	.904	314,500	24,400	4,122	.0311	690	.187	.187	.187	.187	.206				.1754	.351	.421	.234	.1167	.0973
477,000	26	2	.1355	7	.1054	.858	300,000	19,430	3,462	.0290	670	.196	.196	.196	.196	.216				.1790	.358	.430	.237	.1186	.0988
477,000	30	2	.1261	7	.1261	.883	300,000	23,300	3,933	.0304	670	.196	.196	.196	.196	.216				.1766	.353	.424	.235	.1176	.0980
397,500	26	2	.1236	7	.0961	.793	250,000	16,190	2,885	.0265	590	.235			.259				.1836	.367	.441	.244	.1219	.1015	
397,500	30	2	.1151	7	.1151	.806	250,000	19,980	3,277	.0278	600	.235	Same as d-c			.259	Same as d-c			.1812	.362	.435	.242	.1208	.1006
336,400	26	2	.1133	7	.0885	.721	4/0	14,050	2,442	.0244	530	.278			.306				.1872	.376	.451	.250	.1248	.1039	
336,400	30	2	.1059	7	.1059	.741	4/0	17,040	2,774	.0255	530	.278			.306				.1855	.371	.445	.248	.1238	.1032	
300,000	26	2	.1074	7	.0835	.680	188,700	12,650	2,178	.0230	490	.311			.342				.1908	.382	.458	.254	.1269	.1057	
300,000	30	2	.1000	7	.1000	.700	188,700	15,430	2,473	.0241	500	.311			.342				.1883	.377	.452	.252	.1258	.1049	
266,800	26	2	.1013	7	.0788	.642	3/0	11,250	1,936	.0217	460	.350			.385				.1936	.387	.465	.258	.1289	.1074	

* Republished by permission of the Westinghouse Electric Corporation from "Electrical Transmission and Distribution Reference Book."

† Based on copper 97%, aluminum 61% conductivity.

‡ "Current approx. 75% capacity" is 75% of the "Approx. current-carrying capacity, amp" shown in Column 9 and is approximately the current which will produce 50°C conductor temp. (25°C rise) with 25°C air temp., wind 1.4 miles/hour.

§ For conductor at 75°C, air at 25°C, wind 1.4 miles/hour (2 ft/sec), frequency = 60 cycles.



ตาราง ก.5 CHARACTERISTICS OF ALUMINUM CABLE, STEEL-REINFORCED * (Continued)
Single-layer Conductors

Circular mils or A.W.G. alu- minum	Aluminum			Steel			Copper equiva- lent, † circular mils or A.W.G.	Ultimate strength, lb.	Weight, lb/ mile	Geo- metric mean radius at 60 cycles, feet, for cur- rent app- rox. 75% ca- pac- ity ‡	App- rox. cur- rent- car- rying capac- ity, † amp	Resistance, ohms/conductor/mile									Inductive reactance, ohms/conductor/mile at 1-ft spacing						Shunt capacitive reactance, megohms/ conductor/ mile at 1-ft spacing		
	Strands	Layers	Strand diameter, in.	Strands	Strand diameter, in.	Outside diameter, in.						25°C (77°F) Small currents			50°C (122°F) Current approx. 75% capacity ‡			Small currents			Current approx. 75% capacity ‡								
												d-c	25 cycles	50 cycles	60 cycles	d-c	25 cycles	50 cycles	60 cycles	25 cycles	50 cycles	60 cycles	25 cycles	50 cycles	60 cycles				
	266,800	6	1	0.2109	7	0.0703						0.633	3/0	9,645	1,802	0.00684	460	0.351	0.351	0.351	0.352	0.366	0.430	0.510	0.552	0.194	0.388	0.466	0.252
4/0	6	1	.1878	1	.1878	.563	2/0	8,420	1,542	.00814	340	.441	.442	.444	.445	.485	.514	.567	.592	.218	.437	.524	.242	.484	.581	.267	.1336	.1113	
3/0	6	1	.1672	1	.1672	.502	1/0	6,675	1,223	.00600	300	.556	.557	.559	.560	.612	.642	.697	.723	.225	.450	.540	.259	.517	.621	.275	.1377	.1147	
2/0	1	1	.1490	1	.1490	.447	1	5,345	970	.00510	270	.702	.702	.704	.706	.773	.806	.866	.895	.231	.462	.554	.267	.534	.641	.254	.1418	.1182	
1/0	1	1	.1327	1	.1327	.398	2	4,280	769	.00446	230	.885	.885	.887	.888	.974	1.01	1.08	1.12	.237	.473	.568	.273	.547	.656	.292	.1460	.1216	
1	1	1	.1182	1	.1182	.355	3	3,480	610	.00418	200	1.12	1.12	1.12	1.12	1.23	1.27	1.34	1.38	.242	.463	.580	.277	.554	.665	.300	.1500	.1250	
2	6	1	.1052	1	.1052	.316	4	2,790	484	.00418	180	1.41	1.41	1.41	1.41	1.55	1.59	1.66	1.69	.247	.493	.592	.277	.554	.665	.306	.1542	.1265	
2	7	1	.0974	1	.1299	.325	4	3,525	566	.00504	180	1.41	1.41	1.41	1.41	1.55	1.59	1.62	1.65	.247	.493	.592	.267	.535	.642	.306	.1532	.1276	
3	6	1	.0937	1	.0937	.281	5	2,250	384	.00430	160	1.78	1.78	1.78	1.78	1.95	1.98	2.04	2.07	.252	.503	.604	.275	.551	.661	.317	.1583	.1320	
4	6	1	.0834	1	.0834	.250	6	1,830	304	.00437	140	2.24	2.24	2.24	2.24	2.47	2.50	2.54	2.57	.257	.514	.611	.274	.549	.659	.325	.1627	.1356	
4	7	1	.0772	1	.1029	.257	6	2,288	356	.00452	140	2.24	2.24	2.24	2.24	2.47	2.50	2.53	2.55	.257	.515	.618	.273	.545	.655	.323	.1615	.1346	
5	6	1	.0743	1	.0743	.223	7	1,460	241	.00416	120	2.82	2.82	2.82	2.82	3.10	3.12	3.16	3.18	.262	.525	.630	.279	.557	.665	.333	.1666	.1388	
6	6	1	.0661	1	.0661	.198	8	1,170	191	.00394	100	3.56	3.56	3.56	3.56	3.92	3.94	3.97	3.98	.268	.536	.643	.281	.561	.673	.342	.1708	.1423	

* Republished by permission of the Westinghouse Electric Corporation from "Electrical Transmission and Distribution Reference Book."

† Based on copper 97%, aluminum 61% conductivity.

‡ "Current approx. 75% capacity" is 75% of the "Approx. current-carrying capacity, amp" shown in Column 9 and is approximately the current which will produce 50°C conductor temp. (25°C rise) with 25°C air temp., wind 1.4 miles/hour.

§ For conductor at 75°C, air at 25°C, wind 1.4 miles/hour (2 ft/sec), frequency = 60 cycles.

תורת ה.6 CHARACTERISTICS OF EXPANDED ALUMINUM CABLE,
STEEL-REINFORCED*

	1.6 in. expanded ACSR	1.72 in. expanded ACSR	1.75 in. expanded ACSR
Self geometric mean distance, ft at 60 cps.	0.0578	0.0631	0.0644
Resistance at 50°C, 60 cps, ohms/conductor/mile	0.0799	0.0727	0.0710
Inductive reactance at 1-ft spacing, 60 cps, ohms/conductor/mile.	0.347	0.335	0.332
Shunt capacitive reactance at 1-ft spacing, 60 cps, megohms/conductor/mile.	0.0803	0.0782	0.0777
Current-carrying capacity, amp:†			
New cable.	1,380	1,530	1,560
Weathered cable.	1,785	2,090	2,110

* Values from the booklet "Expanded ACSR for Superhigh Voltage Transmission," copyright 1954, Aluminum Company of America, by permission.

† Approximate values based on an ambient temperature of 80°F with 120°F rise, cross wind 2 mph.

ภาคผนวก ข

โปรแกรมคำนวณพารามิเตอร์ของสายส่ง

โปรแกรมคำนวณพารามิเตอร์ของสายส่ง ได้แบ่งออกเป็นโปรแกรมหลักและโปรแกรมย่อยรวม 8 ไฟล์ (file) ดังตารางข้างล่างนี้

ชื่อไฟล์	ชื่อโปรแกรมย่อย สักรหัส	จุดประสงค์
LCON		เป็นโปรแกรมหลักในการคำนวณพารามิเตอร์ของสายส่ง ทั้งอิมพีแดนซ์และคาแปซิแตนซ์
SKIN	SKIN	เรียกโดยโปรแกรมหลัก เพื่อคำนวณผลของ Skin Effect ของความต้านทานและความเหนียวมา
SYM	SYM	เพื่อเปลี่ยน (Convert) อิมพีแดนซ์และคาแปซิแตนซ์ให้อยู่ในรูปของ Symmetrical component
CXR	CXRED 2	เพื่อรวมอิมพีแดนซ์ของ Subconductor ในแต่ละเฟส เป็นอิมพีแดนซ์สมมูลของแต่ละเฟส
REDU	REDU	เพื่อรวมคาแปซิแตนซ์ของ Subconductor ในแต่ละเฟส เป็นคาแปซิแตนซ์สมมูลของแต่ละเฟส

ตาราง ข.1 โปรแกรมย่อยในการคำนวณพารามิเตอร์ของสายส่ง

ตาราง ข.1 (ต่อ)

ชื่อไฟล์	ชื่อโปรแกรมย่อย สักรูทีน	จุดประสงค์
OUTPUT	OUTPUT	เป็นโปรแกรมย่อยในการจัดรูปแบบของผลลัพธ์ เอาที่ทุก
WRIT	WRIT	เรียกโดยโปรแกรมย่อย OUTPUT
MOVEO	MOVEO	เป็นโปรแกรมย่อยที่ใช้ในการรีเซทหน่วยความจำชุด ให้มีค่าเป็นศูนย์

```

C -----
C LINE CONSTANT CALCULATION PROGRAM
C -----
CHARACTER*80 TITLE
CHARACTER*10 FNAME
CHARACTER*6 SYST
CHARACTER*1 MODES
DIMENSION IPH(50),SKN(50),RES(50),IXT(50),REA(50),IC(50)
1 ,DIA(50),TBX(50)
DOUBLE PRECISION ECS(30),CCS(30),DCS(30)
DIMENSION TBY(50),TB2(50),ITB3(50)
DIMENSION D(50),GMD(50),X(50),Y(50)
DOUBLE PRECISION WORKR1(400),WORKR2(400)
DOUBLE PRECISION R(400),P(400),Z(400),TP(400),TZ(400)
DIMENSION DUMY2(50)
DOUBLE PRECISION DELTAP,DELTAQ
WRITE(*,70)
70 FORMAT(' YOU WANT TO INPUT INTERACTIVE MODE (Y/N)')
READ(*,72)MODES
72 FORMAT(A1)
IF(MODES.NE.'Y')GOTO 75
OPEN(3,FILE=' ')
LI=3
GOTO 76
75 WRITE(*,4001)
4001 FORMAT(' NAME OF INPUT FILE ')
READ(*,4002)FNAME
4002 FORMAT(A10)
OPEN(8,FILE=FNAME,STATUS='OLD')
REWIND 8
LI=8
76 WRITE(*,4003)
4003 FORMAT(' NAME OF OUTPUT FILE')
READ(*,4002)FNAME
OPEN(9,FILE=FNAME)
REWIND 9
WRITE(9,4004)
4004 FORMAT(15X,31('*'),/,15X,'*',29X,'*',/,15X,
1 '* LINE CONSTANT CALCULATION *',/,15X,'*',29X,'*'
1,/,15X,31('*'),3(/))
C READ INPUT FILE
C =====
C METRIC =0
WRITE(*,6295)
6295 FORMAT(' INPUT TITLE 80 COL. ')
READ(LI,6300)TITLE
6300 FORMAT(A80)
WRITE(*,6248)
6248 FORMAT(' INPUT SYSTEM UNIT ')
READ(LI,6350)SYST
6350 FORMAT(A6)
WRITE(9,3444)TITLE
3444 FORMAT(5X,' TITLE: > ',A80,/,/)
IF(SYST.EQ.'METRIC')METRIC=1

```

```

C
C
C      START OF CONSTANTS DEFINITION
C      =====
C      VALU1 =5584596.2
C      VALU2 =.00085646541
C      VALU3 = .00064373888
C      VALU4 =.61593152
C      VALU5 = 8.6858896
C      PI =3.1415931516
C      PIC =360./(2*PI)
C      TENM6 =.000001
C      CORCHK =1-TENM6/10.
C
C
C      CONVERT METRIC TO ENGLISH
C      =====
C      FINPCM =1./2.54005
C      FTPM =100.*FINPCM/12.
C      FMIPKM =FTPM*1000./5280.
C      DB =.3048
C      FSPAC =0.024*2.*PI*ALOG(DB)
C
C
C      SET CARSDN'S CONSTANT
C      =====
C      CCS(2)=1.365931516
C      BCS(1) =SQRT(2.)/6.
C      BCS(2) =1./16.
C      DCS(2) =BCS(2)*PI/4.
C      DO 6000 I=3,30
C      ISN =(-1)**((I-1)/2)
C      BCS(I) =-BCS(I-2)/(I*I+2.*I)*ISN
6000  DCS(I) =BCS(I)*PI/4
C      CCS(1) =1./SQRT(2.)
C      CCS(3) =CCS(1)
C      CCS(5) =3.*SQRT(2.)/2.
C      CCS(7) =-45.*.5*SQRT(2.)
C      DO 6100 I=1,29,2
C      IF(I.GT.8)CCS(I)=0.0
6100  DCS(I) =CCS(I)
C      DCS(3) =-DCS(3)
C      DCS(7) =-DCS(7)
C      DO 6200 I=4,30,2
6200  CCS(I) =CCS(I-2)+1./I+1./(I+2)
C
C
C      -----
C
C      IF(MODES.NE.'Y')GOTO 79
C      WRITE(*,77)
77    FORMAT(' IF YOU WANT <HELP FOR INPUT CONDUCTOR DATA> INPUT 1')
C      READ(*,6113)IM
6113  FORMAT(I1)
C      IF(IM.NE.1)GOTO 79
C      WRITE(*,78)
78    FORMAT(' CONDUCTOR DATA CONSISTS OF 10 FIELDS: 1.,
1 5X. <PH. NO.> . . : PHASE NUMBER 1,2,3..... 77,

```

```

1 5X, '<SKIN EFF.>' : SKIN EFFECT = RATIO OUTER/INNER RADIUS',/,
1 5X, '          ' : 0 FOR BYPASS THIS FIELD',/,
1 5X, '<RESIST>'   : DC RESISTANCE (OHM/KM OR OHM/MILE)',/,
1 5X, '<TYPE>'     : SELECT NEXT FIELD INPUT (2 OR 4)',/,
1 5X, '<GMR>'      : TYPE =2 MEAN GMR OF CONDUCTOR (CM/INCH)',/,
1 5X, '          ' : =4 MEAN BYPASS THIS FIELD',/,
1 5X, '<DIA>'      : CONDUCTOR DIAMETER (CM OR INCH)',/,
1 5X, '<H-DIST>'   : HORIZONTAL DISTANCE (METER OR FEET)',/,
1 5X, '<V-HGHT>'  : VERTICAL HEIGHT (METER OR FEET)',/,
1 5X, '<MV-HGHT>' : MIDSPAN VERTICAL HEIGHT (METER OR FEET)',/,
1 5X, '<S-BDLE>'  : SEPARATION BETWEEN BUNDLE (CM OR INCH)',/,
1 5X, '<A-BDLE>'  : ANGLE BETWEEN BUNDLE (CM OR INCH)',/,
1 5X, '<N-BDLE>'  : NUMBER OF BUNDLE',/,
1 ' 888 FOR PHASE NUMBER MEAN END CONDUCTOR DATA')
79   LPHASE =50
      I=0
7000 I=I+1
      IF(MODES.NE.'Y')GOTO 84
      WRITE(*,82)
82   FORMAT(' <PH. NO.> <SKIN EFF.> <RESIST> <TYPE> <GMR> <DIA>')
      READ(LI,*)IPH(I),SKN(I),RES(I),IXT(I),REA(I),DIA(I)
      WRITE(*,81)
81   FORMAT(' <H-DIST> <V-HGT> <MV-HGT> <S-BDLE> <A-BDLE> <N-BDLE>')
      READ(LI,*)TBX(I),H1,H2,DB,D9,I3
      GOTO 85

C
C
C   READ CONDUCTOR CARD
C   =====
C
84   READ(LI,*)IPH(I),SKN(I),RES(I),IXT(I),REA(I),DIA(I),TBX(I),H1,H2
1,DB,D9,I3
7005 FORMAT(I3,F5.4,F6.5,I2,2F8.5,3F8.3,F6.5,F6.2,I2)
      WRITE(*,7005)IPH(I),SKN(I),RES(I),IXT(I),REA(I),DIA(I),TBX(I)
1,H1,H2,DB,D9,I3
85   IF(IPH(I).EQ.888)GOTO 7500
      TBY(I) =(H1+2*H2)/3.0
      IF(I3.LE.1)GOTO 706
      I4 =1
      XX =TBX(I)
      YY =TBY(I)
      ANGL=(PI-2.*PI/I3)/2.0
      DANG =2.*PI/I3
      RADIUS =DB/(24.0*COS(ANGL))
      IF(METRIC.EQ.1)RADIUS=DB/(200.*COS(ANGL))
      D9R =D9/FIC
703  TBX(I) =XX+RADIUS*COS(D9R-DANG*I4)
      TBY(I) =YY+RADIUS*SIN(D9R-DANG*I4)
      IF(I4.EQ.I3)GOTO 706
      I4=I4+1
      I=I+1
      IPH(I) =IPH(I-1)
      SKN(I) =SKN(I-1)
      RES(I) =RES(I-1)
      IXT(I) =IXT(I-1)

```

```

      REA(I) =REA(I-1)
      DIA(I) =DIA(I-1)
      GOTO 703
706   IF(I.LT.LPHASE)GOTO 7000
707   WRITE(9,708)
708   FORMAT('  TABLE OVERFLOW')
      GOTO 600

C
C   SORT OF CONDUCTOR
C   =====
7500  CONTINUE
      I=I-1
      IF(METRIC.NE.1)GOTO 8000
      DO 8005 K=1,I
      DIA(K) =DIA(K)*FINPCM
      TBX(K) =TBX(K)*FTPM
      TBY(K) =TBY(K)*FTPM
      RES(K) =RES(K)/FMIPKM
      IF(IXT(K).EQ.2)REA(K)=REA(K)*FINPCM
8005  CONTINUE
8000  CONTINUE
8010  CALL MOVE0(IC,LPHASE)
      LPHPL1=LPHASE+1
      KTOT =0
      KCIR =0
      K =0
10    K=K+1
      IF(DIA(K).EQ.0.)GOTO 15911
      I1=IPH(K)
      IF(I1.EQ.0)GOTO 12
      IF(I1.LT.0)GOTO 10
      IF(I1.GT.KCIR)KCIR=I1
      I=LPHPL1-I1
      IF(IC(I).EQ.0)GOTO 13
12    KTOT =KTOT+1
      I=KTOT
      IF(I1.EQ.0)GOTO 13
      IP=KTOT
18    J=I-1
      IF(J.EQ.0)GOTO 13
      IF(IC(J).GT.0)GOTO 13
      J=J
      IC(IP) =IC(J)
      TB2(IP) =TB2(J)
      ITB3(IP)=ITB3(J)
      R(IP)=R(J)
      D(IP) =D(J)
      GMD(IP) =GMD(J)
      X(IP) =X(J)
      Y(IP) =Y(J)
      IP =IP-1
      GOTO 18
13    KFULL =KTOT+KCIR
      IF(KFULL.GT.LPHASE)GOTO 707
      IC(I) =I1

```



```

      TB2(I) =SKN(K)
      ITB3(I) =IXT(K)
      R(I)=RES(K)
      D(I)=DIA(K)
      GMD(I) =REA(K)
      X(I) =TBX(K)
      Y(I) =TBY(K)
      GOTO 10
15911 IF(MODES.NE.'Y')GOTO 15
      WRITE(*,86)
86   FORMAT(' IF YOU WANT <HELP FOR FREQ-OUTPUT DATA> INPUT 1')
      READ(*,6113)IM
      IF(IM.NE.1)GOTO 15
      WRITE(*,87)
87   FORMAT(' FREQUENCY & OUTPUT DATA :-
1 5X,'<REARTH>      : EARTH RESISTIVITY (OHM) ',/,
1 5X,'<FREQ>        : FREQUENCY (HZ) ',/,
1 5X,'<CORR>       : EARTH CORRECTION 1 IF YOU WANT ',/,
1 5X,'<J1>         : INVERSE FORM OF C OR WC ',/,
1 5X,'<J2>         : INVERSE FORM OF C OR WCequivalent ',/,
1 5X,'<J3>         : INVERSE FORM OF C OR WCsymmetrical ',/,
1 5X,'<J4>         : C OR WC ',/,
1 5X,'<J5>         : C OR WCequivalent ',/,
1 5X,'<J6>         : C OR WCsymmetrical ',/,)
      WRITE(*,88)
88   FORMAT(5X,'<J7>          : SERIES IMPEDANCE Z ',/,
1 5X,'<J8>          : SERIES IMPEDANCE Zequivalent ',/,
1 5X,'<J9>          : SERIES IMPEDANCE Zsymmetrical ',/,
1 5X,'<J10>         : INVERSE FORM OF Z ',/,
1 5X,'<J11>         : INVERSE FORM OF Zequivalent ',/,
1 5X,'<J12>        : INVERSE FORM OF Zsymmetrical ',/,
1 5X,'<IW>         : SELECT MODE FOR J1-J6 ',/,
1 5X,'              : "1" - C OUTPUT ',/,
1 5X,'              : "0" - WC OUTPUT ',/,
1 5X,'<DIST>       : DISTANCE (KM OR MILE) ',/,
1 5X,'J1-J12 INPUT 1 FOR SELECTION ')

C
C   READ FREQUENCY CARD
C   =====
C
15  CONTINUE
      IF(MODES.NE.'Y')GOTO 16
      WRITE(*,89)
89  FORMAT(' INPUT FREQUENCY & OUTPUT
1 '<REARTH> <FREQ> <CORR> <J1> <J2> <J3> <J4> <J5> <J6> ')
      READ(LI,*)REARTH,FREQ,CORR,J1,J2,J3,J4,J5,J6
      WRITE(*,90)
90  FORMAT(' <J7> <J8> <J9> <J10> <J11> <J12> <IW> <DIST> ')
      READ(LI,*)J7,J8,J9,J10,J11,J12,IW,DIST
      ISEGM=0
      IMODAL=0
      GOTO 4929
16  READ(LI,*)REARTH,FREQ,CORR,J1,J2,J3,J4,J5,J6,J7,J8,J9,J10,J11
17  ,J12,IW,DIST,ISEGM,IMODAL
      FORMAT(F8.2,F10.2,F10.6,12,511,12,511,12,F8.3,212)

```

```

WRITE(*,17)REARTH,FREQ,CORR,J1,J2,J3,J4,J5,J6,J7,J8,J9,J10,J11
1,J12,IW,DIST,ISEGM,IMODAL
4929 IF(IMODAL.LT.0.DR.DIST.LE.0)GOTO 9999
      CK1 =DIST
      IF(METRIC.EQ.1)CK1=CK1*FMIPKM
      IF(REARTH.LE.0.0)GOTO 9999
C     HEADING
      WRITE(9,5008)
5008  FORMAT(1X,125(1H=),/)
      WRITE(9,5009)
      WRITE(9,5008)
      OMEGA= 2.*PI*FREQ
3008  IDIST =1
      IF(DIST.EQ.0.0)GOTO 4271
      IF(KCIR.EQ.0)GOTO 4271
      GOTO 4275
4271  IDIST =0
4275  J56=J5+J6+IDIST
C
C     FORMATION OF P MATRIX
C     =====
      IP=0
      K=0
      F1=1./OMEGA
      IF(IW.GT.0)F1=1.0
      F1=F1*VALU1
      F2 =F1*2.0
20    K=K+1
      IF(K.GT.KFULL)GOTO 30
      IF(K.LE.KCIR)GOTO 28
      J=K-KCIR
21    X1=X(J)
      Y1=Y(J)
      D1=D(J)
      IF(METRIC.EQ.0)GOTO 221
      RJ=R(J)*FMIPKM
      YJ=Y(J)/FTPM
      XJ=X(J)/FTPM
      DJ=D(J)/FINPCM
      GMDJ =GMD(J)
      JB=ITB3(J)
      IF(JB.EQ.2)GMDJ=GMDJ/FINPCM
5009  FORMAT(1X,' NO. ',3X,' PHASE',6X,' SKIN',10X,' RESIST ',4X,' TYPE '
1,15X,' GMD ',12X,' DIAMETER',3X,' HORIZONTAL',3X,' VERTICAL ')
      WRITE(9,22)K,IC(J),TB2(J),RJ,JB,GMDJ,DJ,XJ,YJ
      GOTO 225
221  WRITE(9,22)K,IC(J),TB2(J),R(J),ITB3(J),GMD(J),D(J),X(J),Y(J)
22   FORMAT(1X,15,18,2F14.5,111,F20.6,F13.5,2F13.3)
225  IF(ITB3(J).NE.4)GOTO 222
      IF(TB2(J).GT.0.0)GOTO 224
      WRITE(9,5002)
5002  FORMAT(5X,'ERROR < SKIN EFFECT=0>')
      GOTO 9999
224  GMD(J)=D(J)
222  I=0

```

```

23      I=I+1
        IF=IP+1
        IF(I.EQ.K)GOTO 24
        J=I-KCIR
        IF(J.LE.KCIR)J=LPHPL1-I
        DX=(X(J)-X1)**2
        H1=Y(J)-Y1
        H2=Y(J)+Y1
        IF(DX.EQ.0.0.AND.H1.EQ.0.0)GOTO 9999
        R1=ALOG((DX+H2*H2)/(DX+H1*H1))
        P(IP)=R1*F1
        Z(IP)=R1
        GOTO 23
24      P(IP)=ALOG(4B.*Y1/D1)*F2
        GOTO 20
28      J=LPHPL1-K
        IF(IC(J).NE.0)GOTO 21
        GOTO 9999

C
C      REDUCTION AND INVERSION
C      =====
30      CONTINUE
        IPP=IP
        DO 4089 I=1,IP
4089      TP(I)=Z(I)
        SWITCH =-1.0
        IF(J1.GT.0)GOTO 36
31      IF(J4.GT.0)GOTO 37
32      IF(KCIR.EQ.0)GOTO 80
        IF(J2+J3+J56.EQ.0)GOTO 80

C
C      BEGIN ELIMINATION OF EARTH CONDUCTOR AND BUNDLING
C
49      K=KCIR
50      K=K+1
        IF(K.GT.KFULL)GOTO 56
        I=K-KCIR
        I=IC(I)
        IF(I.EQ.0)GOTO 50
        I2=I*(I-1)/2
        K2=K*(K-1)/2
        KK=K2+I
        H1=P(KK)
        H2=Z(KK)
        KK=K2+K
        L=0
51      IF(L.LT.1)GOTO 54
        I2 =I2+L
52      IF(L.LT.K)GOTO 55
        K2=K2+L
53      L=L+1
        P(K2)=P(K2)-P(I2)
        IF(SWITCH.GT.0.)Z(K2)=Z(K2)-Z(I2)
        IF(L.NE.KFULL)GOTO 51
        P(KK)=P(KK)-H1

```

```

IF (SWITCH.GT.0.) Z(KK)=Z(KK)-H2
54  GOTO 50
    I2=I2+1
    GOTO 52
55  K2=K2+1
    GOTO 53
56  KP=KCIR*(KCIR+1)/2
    IF (SWITCH.GT.0.) GOTO 202
    CALL REDU(P,WORKR1,KFULL,KCIR)
C
C  END ELIMINATION
C
    IF (J2+J3.GT.0) GOTO 38
33  IF (J56.GT.0) GOTO 39
C
C  FORMATION OF Z-MATRIX
C  =====
80  DD 4812 I=1,IPP
4812 P(I)=TP(I)
    SWITCH =1.
    J56 =J8+J9+J11+J12
    IF (J56+J7+J10.EQ.0) GOTO 600
    K=0
    IF (REARTH.EQ.0.) GOTO 100
    F1=VALU2*SQRT(FREQ/REARTH)
100  F2=OMEGA*VALU3
    IMAX=30
    IF (CORR.EQ.0..OR.REARTH.EQ.0.) IMAX=-1
    IF (CORR.GT.CORCHK) IMAX=CORR-0.5
    IF (IMAX.GT.31) IMAX=30
    IP=0
120  K=K+1
    IF (K.GT.KFULL) GOTO 200
    IF (K.LE.KCIR) GOTO 129
    J=K-KCIR
    GOTO 121
129  J=LPHPL1-K
121  X1=X(J)
    Y1=Y(J)
    R1=R(J)
    H1=TB2(J)
    H2=1.-2.0*H1
    IF (H1.GT.0.) CALL SKIN(H2,R1,FREQ,R1,H1)
    G1 =GMD(J)
    IX =ITB3(J)
    IZERO =1
    XM =0.
    IF (IC(J).EQ.0.AND.ISEGM.GT.0) IZERO=0
    IF (IX.EQ.3) G1=G1*D(J)/2.0
    XS=ALOG(24.0*Y1/G1)*0.5
122  IF (IX.NE.4) GOTO 125
    XS *XS+H1/F2-0.5*ALOG(D(J)+0.5/G1)
125  I=0
123  I=I+1
    IP =IP+1

```

```

IF (I.EQ.K)GOTO 124
J=I-KCIR
IF (I.LE.KCIR)J=LPHPL1-I
RM=0.
IF (IZERO.EQ.0)GOTO 126
XM =P(IP)/4.0
IF (IMAX.LT.0)GOTO 126
DX=ABS(X(J)-X1)
H2=Y(J)+Y1
S=SQRT(H2*H2+DX*DX)
Z1=S*F1
IF (Z1.GT.5.0)GOTO 151
C BEGIN CARSON SERIES
RM=2.0*F1/16.
ZL =ALOG(Z1)
XM=XM+(VALU4-ZL)*0.5
IF (IMAX.EQ.0)GOTO 126
S1=DX/S
C1=H2/S
CS=C1*Z1
SN =S1*Z1
IF (IMAX.GT.1)PHI=ASIN(S1)
M5=0
ERROR=0.
1110 K5=0
IS=M5*4
1111 K5=K5+1
IS=IS+1
IF (K5.EQ.1.OR.K5.EQ.3)GOTO 1113
DELTAP =((CCS(IS)-ZL)*CS+PHI*SN)*BCS(IS)
H1=-DCS(IS)*CS
IF (K5.EQ.4)GOTO 1114
DELTAP=H1
1004 RM=RM+DELTAP
XM=XM+DELTAP
IF (IS.EQ.IMAX)GOTO 126
IF (CORR.GT.CORCHK)GOTO 1005
IF (DABS(DELTAP).LT.CORR.AND.DABS(DELTAP).LT.CORR)GOTO 1115
ERROR=0.
1005 H1=SN*C1+CS*S1
CS=(CS*C1-SN*S1)*Z1
SN=H1*Z1
IF (K5.LT.4)GOTO 1111
M5=M5+1
GOTO 1110
1113 DELTAP=BCS(IS)*CS
DELTAP=DELTAP
IF (K5.EQ.1)DELTAP=-DELTAP
GOTO 1004
1114 DELTAP=-DELTAP
DELTAP=H1
GOTO 1004
1115 IF (ERROR.GT.0.5)GOTO 126
ERROR=1.0
GOTO 1005

```

```

C
C BEGIN CARSON SERIES IF Z>5 FOR MUTUAL IMPEDANCES
C =====
151  S=S*Z1
      SN =DX/S
      CS=H2/S
      S2=SN*CS*2.0
      C2=CS*CS-SN*SN
      RM=-C2
      DO 1006 I5=1,7,2
      RM=RM+CCS(I5)*CS
      XM=XM+DCS(I5)*CS
      H1=CS*S2+SN*C2
      CS=CS*C2-SN*S2
1006 SN=H1
C    END -----
126  P(IP) =RM*F2
      Z(IP)=XM*F2
      GOTO 123
124  IF(IMAX.LT.0)GOTO 128
      R1=R1/F2
      Z1=Y1*F1*2.0
      IF(Z1.GT.5.0)GOTO 152
C    BEGIN CARSON IF Z<5
      R1=R1+2*P1/16.
      ZL=ALOG(Z1)
      XS=XS+(VALU4-ZL)*0.5
      IF(IMAX.EQ.0)GOTO 127
      M5=0
      CS=Z1
      ERROR=0.
2110 K5=0
      I5=M5*4
2111 K5=K5+1
      I5=I5+1
      IF(K5.EQ.1.OR.K5.EQ.3)GOTO 2113
      DELTAP =(CCS(I5)-ZL)*BCS(I5)*CS
      H1=-DCS(I5)*CS
      IF(K5.EQ.4)GOTO 2114
      DELTAQ=H1
2004 R1=R1+DELTAP
      XS=XS+DELTAP
      IF(I5.EQ.IMAX)GOTO 127
      IF(CORR.GT.CORCHK)GOTO 2005
      IF(DABS(DELTAP).LT.CORR.AND.DABS(DELTAP).LT.CORR)GOTO 2115
      ERROR =0.
2005 CS=CS*Z1
      IF(K5.LT.4)GOTO 2111
      M5=M5+1
      GOTO 2110
2113 DELTAQ=BCS(I5)*CS
      DELTAP=DELTAP
      IF(K.EQ.1)DELTAP=-DELTAP
      GOTO 2004
2114 DELTAQ=-DELTAP

```

```

          DELTAP =H1
          GOTO 2004
2115     IF(ERROR.GT.0.5)GOTO 127
          ERROR=1.
          GOTO 2005
C END FOR Z<5
152     CS=1.0/Z1
          C2=CS*CS
          R1=R1-C2
          DO 2006 I=1,7,2
          R1=R1+CCS(I5)*CS
          XS=XS+DCS(I5)*CS
2006     CS=CS*C2
C -----
127     R1=R1*F2
128     Z(IP)=XS*F2
          P(IP)=R1
          GOTO 120
C -----
C ROUTINE FOR Z-PRINTING
C -----
200     IF(J7.GT.0)CALL OUTPUT(METRIC,P,Z,1.0,KFULL,6,1)
          IF(J10.GT.0)GOTO 250
          IF(J56.EQ.0)GOTO 600
201     IF(KCIR.EQ.0)GOTO 600
          GOTO 49
202     CALL CXRED2(P,Z,WORKR1,WORKR2,KFULL,KCIR)
          IF(J8.GT.0)CALL OUTPUT(METRIC,P,Z,1.0,KCIR,6,2)
3203    IF(IMODAL.LE.0)GOTO 2202
          IF(KCIR.LE.10)GOTO 1202
          GOTO 9999
1202    CONTINUE
C ----- CM -----
2202    CONTINUE
          J56=J11+J12+IDIST
          IF(J9.GT.0.AND.KCIR.GE.2)GOTO 253
203     IF(J56.EQ.0)GOTO 600
          IF(J9.EQ.0)GOTO 229
          ASSIGN 229 TO INDEX
          IP=KP
          GO TO 7802
229     CONTINUE
          DO 204 I=1,KF
          P(I)=-P(I)
204     Z(I)=-Z(I)
          CALL CXRED2(P,Z,WORKR1,WORKR2,KCIR,0)
          IF(J11.GT.0)CALL OUTPUT(METRIC,P,Z,1.0,KCIR,5,2)
207     IF(J12.EQ.0.OR.KCIR.LT.2)GOTO 600
          CALL SYM(P,Z,1.0,KCIR,KK)
          CALL OUTPUT(METRIC,P,Z,1.0,KK,5,3)
          GOTO 600
250     IF(J56.EQ.0)GOTO 251
          ASSIGN 251 TO INDEX
          GOTO 7800
251     DO 252 I=1,IF

```

```

252      P(I)=-P(I)
        Z(I)=-Z(I)
        CALL CXRED2(P,Z,WORKR1,WORKR2,KFULL,0)
        CALL OUTPUT(METRIC,P,Z,1.0,KFULL,5,1)
        IF(J56.EQ.0)GOTO 600
        ASSIGN 201 TO INDEX
        GO TO 7802
253      IF(J56.EQ.0)GOTO 254
        ASSIGN 254 TO INDEX
        IP=KF
        GO TO 7800
254      CALL SYM(P,Z,1.0,KCIR,KK)
        IF(J9.6T.0)CALL OUTPUT(METRIC,P,Z,1.0,KK,6,3)
C -----
C OUTPUT - SURGE IMPEDANCE
C         - ATTENUATION
C -----
        IF(J6.EQ.0)GOTO 9998
        RZERO =P(1)
        XZERO =Z(1)
        RPOS  =P(5)
        XPOS  =Z(5)
        Z0=SQRT(RZERO**2+XZERO**2)
        Z1=SQRT(RPOS**2+XPOS**2)
        ZS0=SQRT(Z0/YZERO)
        ZS1=SQRT(Z1/YPOS)
        ZOANG=ATAN2(XZERO,RZERO)*0.5
        Z1ANG=ATAN2(XPOS,RPOS)*0.5
        PROPO=SQRT(Z0*YZERO)
        PROP1 =SQRT(Z1*YPOS)
        ZS0DRG=ZOANG*PIC-45.
        ZS1DRG=Z1ANG*PIC-45.
        D2=2.*PI/B.0
        ALPHAO=PROPO*COS(ZOANG+D2)
        ALPHA1=PROP1*COS(Z1ANG+D2)
        BETO=PROPO*SIN(ZOANG+D2)
        BET1=PROP1*SIN(Z1ANG+D2)
        D13 =2.*PI*FREQ
        IF(IMODAL.NE.0)GOTO 2254
        DOR =ZS0*COS(ZOANG-D2)
        DCI =ZS0*SIN(ZOANG-D2)
        DIR =ZS1*COS(Z1ANG-D2)
        DII=ZS1*SIN(Z1ANG-D2)
        IF(METRIC.EQ.0)GOTO 1260
        ALPHAO =ALPHAO*FMIPKM
        ALPHA1 =ALPHA1*FMIPKM
        BETO=BETO*FMIPKM
        BET1 =BET1*FMIPKM
1260      CONTINUE
2254      RZERO1=RZERO
        XZERO1=XZERO
        XPOS1=XPOS
        YZERO1=YZERO
        YPOS1=YPOS
        Z0=1000./OMEGA

```



```

Z1=1000000./OMEGA
XZERO =XZERO*Z0
XPOS =XPOS*Z0
YZERO =YZERO*Z1
YPOS =YPOS*Z1
IF(METRIC.EQ.0)GOTO 3010
RZERO =RZERO*FMIPKM
XZERO =XZERO*FMIPKM
RPOS =RPOS*FMIPKM
XPOS =XPOS*FMIPKM
3010 VOL=OMEGA/BETO
ALPHA0=ALPHA0*VALUS
ALPHA1=ALPHA1*VALUS
VIL =OMEGA/BET1
WAV0=2.*PI/BETO
WAV1 =2.*PI/BET1
IF(METRIC.EQ.1)GOTO 9992
WRITE(9,9991)ZS0,ZS0DRG,ALPHA0,VOL,WAV0,RZERO,XZERO1,YZERO1,ZS1,
1ZS1DRG,ALPHA1,VIL,WAV1,RPOS,XPOS1,YPOS1
9991 FORMAT(9H SEQUENCE,6X,15HSURGE IMPEDANCE,7X,'ATTENUATION',3X,
1'VELOCITY',4X,'WAVELENGTH',3X,'RESISTANCE',4X,'REACTANCE',3X,
1'SUSCEPTANCE',/,9X,'MAGNITUDE(OHM) ANGLE(DEGR.)',3X,'DB/MILE'
1,6X,'MILES/S',7X,'MILES',6X,'OHM/MILE',5X,'OHM/MILE',5X,
1'MHO/MILE',/,3X,'ZERO ',8E13.5,/, ' POSITIVE ',8E13.5)
GOTO 9994
9992 CONTINUE
IF(IMODAL.EQ.0)GOTO 9889
ALPHA0 =ALPHA0*FMIPKM
ALPHA1=ALPHA1*FMIPKM
VOL =VOL/FMIPKM
VIL =VIL/FMIPKM
WAV0 =WAV0/FMIPKM
WAV1 =WAV1/FMIPKM
C RZERO1 =RZERO1*FMIPKM
C RPOS1 =RPOS1*FMIPKM
9889 XZERO1 =XZERO1*FMIPKM
XPOS1 =XPOS1*FMIPKM
YPOS1 =YPOS1*FMIPKM
YZERO1=YZERO1*FMIPKM
WRITE(9,9993)ZS0,ZS0DRG,ALPHA0,VOL,WAV0,RZERO,XZERO1,YZERO1,
1ZS1,ZS1DRG,ALPHA1,VIL,WAV1,RPOS,XPOS1,YPOS1
9993 FORMAT(9H SEQUENCE,6X,15HSURGE IMPEDANCE,7X,'ATTENUATION',3X,
1'VELOCITY',4X,'WAVELENGTH',3X,'RESISTANCE',4X,'REACTANCE',3X,
1'SUSCEPTANCE',/,9X,'MAGNITUDE(OHM) ANGLE(DEGR.)',3X,'DB/KM'
1,6X,' KM/S ',7X,' KMS ',6X,' OHM/KM ',5X,' OHM/KM ',5X,
1' MHO/KM ',/,3X,'ZERO ',8E13.5,/, ' POSITIVE ',8E13.5)
9994 CONTINUE
9998 CONTINUE
9996 IF(JS6.EQ.0)GOTO 600
GO TO 203
-----
C
C END OF ROUTINES FOR Z PRINTING AND INVERSION
C
C ROUTINE FOR P - PRINTING AND INVERSION
C

```

```

C -----
36      I2=1
        IF (IW.EQ.0) I2=I2+1
        CALL OUTPUT(METRIC,P,Z,SWITCH,KFULL,I2,1)
        GOTO 31
37      I2=3
        IF (IW.EQ.0) I2=I2+1
        DO 371 I=1,IP
371     Z(I)=-P(I)
        CALL REDU(Z,WORKR1,KFULL,0)
        CALL OUTPUT(METRIC,Z,P,SWITCH,KFULL,I2,1)
        GOTO 32
38      I2=1
C
C      CAPACITANCE MATRIX FOR EQUIVALENT
C      =====
403     IF (IW.EQ.0) I2=I2+1
        IF (J2.GT.0) CALL OUTPUT(METRIC,P,Z,SWITCH,KCIR,I2,2)
        DO 481 I=1,KP
481     DUMY2(I)=P(I)
        IF (J3.EQ.0.OR.KCIR.LT.2) GOTO 33
40      CALL SYM(P,Z,SWITCH,KCIR,KK)
        I2=1
        IF (IW.EQ.0) I2=I2+1
        IF (J3.GT.0) CALL OUTPUT(METRIC,P,Z,1.0,KK,I2,3)
        GOTO 33
C
C      CAPACITANCE MATRIX
C      =====
39      I2=3
        IF (IW.EQ.0) I2=I2+1
        DO 391 I=1,KP
391     P(I)=-DUMY2(I)
        CALL REDU(P,WORKR1,KCIR,0)
        IF (J5.GT.0) CALL OUTPUT(METRIC,P,Z,SWITCH,KCIR,I2,2)
        ZERO =0,0
393     IF (J6.EQ.0.OR.KCIR.LT.2) GOTO 80
        CALL SYM(P,Z,SWITCH,KCIR,KK)
        I2=3
        IF (IW.EQ.0) I2=I2+1
C
C      SUSCEPTANCE OR CAPACITANCE FOR SYM COMP.
C      =====
        IF (J6.GT.0) CALL OUTPUT(METRIC,P,Z,1.0,KK,I2,3)
        YZERO =P(1)
        YPOS =P(5)
        IF (IW.EQ.0) GOTO 80
        YZERO =YZERO*OMEGA
        YPOS =YPOS*OMEGA
        GOTO 80
7800    CONTINUE
        DO 7071 J=1,JP
        TP(J)=P(J)
7071    TZ(J)=Z(J)
        GO TO INDEX,(251,254)

```

```
7802 CONTINUE
      DO 7072 J=1,IP
      P(J)=TP(J)
7072  I(J)=TZ(J)
      GO TO INDEX,(201,229)
9999  WRITE(9,4999)
4999  FORMAT(1X,'INPUT ERROR')
600   WRITE(9,5008)
      STOP
      END
```

```

SUBROUTINE SKIN(S,R,FREQ,RF,XF)
DOUBLE PRECISION FBE(20),FBED(20),FKE(20),FKED(20)
DOUBLE PRECISION VALU10,VALU9,VALU8,VALU11,VALU12
DOUBLE PRECISION VALU13,AAA1,AAA2
C   SET THE CONSTANT VALUE
VALU10 =.5772156649
UNITY =1.
FBE(1)=16.
FBED(1)=-4.
D1=UNITY-VALU10
FKE(1)=FBE(1)*D1
DO 888 I=2,14
ISN =(-1)**I
FBE(I)=FBE(I-1)*(16./(I*I))*(-ISN)
FBED(I)=FBE(I)/(2.*I+2.)*ISN
D1=D1+UNITY/I
FKE(I)=FBE(I)*D1
888  FKED(I-1)=FKE(I)*I/32.
FKED(14)=0.0
VALU9 =FKE(1)/32.
FBE(15)=.0110486
FBE(16)=0.0
FBE(17)=-.0000906
FBE(18)=-.00002520
FBE(19)=-.0000034
FBE(20)=.0000006
FBED(15)=-.0110485
FBED(16)=-.0009765
FBED(17)=-.00009010
FBED(18)=0.0
FBED(19)=.0000051
FBED(20)=.00000190
FKE(15)=-.0625001
FKE(16)=-.0013813
FKE(17)=.0000005
FKE(18)=.0000346
FKE(19)=.0000117
FKE(20)=.0000016
FKED(15)=-.0000001
FKED(16)=.0013811
FKED(17)=.0002452
FKED(18)=.0000338
FKED(19)=-.0000024
FKED(20)=-.0000032
TENM6=0.000001
VALU8 =.40447306E-2
VALU9 =.21139217
VALU11 =.3926991
VALU12 =.39894228
VALU13 =1.25331414
AAA1 =.785398163
AAA2 =.31830988618
C
C   START TO CALCULATE
S2 =S*S

```

```

S3=(1.-S2)*R
R2=FREQ*VALUB/S3
RF=R
XF =0
IF(R2.EQ.0)RETURN
QREMB =0.0
IF(S.LT.TENM6)GOTO 5
Q2=R2*S2
IF(S2.LT.0.8)GOTO 11
IF(Q2.LE.64.0.AND.R2.GT.64.0)WRITE(6,899)Q2,R2
899 FORMAT(5X,2(5X,F10.5))
11 IF(Q2.GT.64.0)QREMB=SQRT(Q2)*SQRT(2.)
X=SQRT(Q2)
X2=X*X/64.0
IBACK=2
IF(X2.LE.1.0)GOTO 100
GOTO 200
4 A=-BERD
B=-BEID
AREMB =GERD
BREMB=GEID
XREMB =X
5 X=SQRT(R2)
X2=X*X/64.0
IBACK=1
IF(X2.LE.1.0)GOTO 100
GOTO 200
6 G=BER
H=BEI
E=BERD
F=BEID
IF(S.LT.TENM6)GOTO 7
G=A*GER-B*GEI+AREMB*BER-BREMB*BEI
H=A*GEI+B*GER+AREMB*BEI+BREMB*BER
E=A*GERD-B*GEID+AREMB*BERD-BREMB*BEID
F=A*GEID+B*GERD+AREMB*BEID+BREMB*BERD
7 E2F2=E**2+F**2
S2=X*S3*(0.5)/E2F2
RF=(-H*E+G*F)*S2
XF=(G*E+H*F)*S2
RETURN
C CALCULATION OF KELVIN FUNCTION FOR ARG. < 3
100 Z=X2
BER =1.0
BEI=0.0
BERD=0.0
BEID=0.5
GERD =0.0
GEID =VALU9
GER=-VALU10
GEI =0.0
IALT =1
DO 103 K=1,14
IF(IALT.EQ.1)GOTO 101
BER =BER+FBE(K)*Z

```

```

BEID =BEID+FBED(K)*Z
IF(S.LT.TENM6)GOTO 102
GEID=GEID+FKED(K)*Z
IF(IBACK.EQ.2)GOTO 102
GER=GER+FKE(K)*Z
GOTO 102
101 BEI=BEI+FBE(K)*Z
BERD=BERD+FBED(K)*Z
IF(S.LT.TENM6)GOTO 102
GERD=GERD+FKED(K)*Z
IF(IBACK.EQ.2)GOTO 102
GEI=GEI+FKE(K)*Z
102 Z=Z*X2
103 IALT=-IALT
BEID=BEID*X
BERD=BERD*X
IF(S.LT.TENM6)GOTO 104
XL =ALOG(X*0.5)
GERD=-XL*BERD-BER/X+BEID*AAA1+X*GERD
GEID=-XL*BEID-BEI/X-BERD*AAA1+X*GEID
IF(IBACK.EQ.2)GOTO 104
GER=-XL*BER+BEI*AAA1+GER
GEI=-XL*BEI-BER*AAA1+GEI
104 GOTO (6,4),IBACK
C CALCULATION OF KELVIN FUNCTION FOR ARG. > 8
200 X2=8.0/X
Z =X2
BER =0.0
BEI =-VALU11
BERD =BER
BEID=BEI
GER =1.0/SQRT(2.)
GEI =GER
GERD =GER
GEID =GEI
IALT=1
DO 203 K=1,6
THETA R =FBE(K+14)*Z
THETA I =FBED(K+14)*Z
PHI R =FKE(K+14)*Z
PHI I =FKED(K+14)*Z
BER =BER+THETA R
BEI =BEI+THETA I
GER =GER+PHI R
GEI =GEI+PHI I
IF(IALT.EQ.1)GOTO 201
BERD =BERD+THETA R
BEID =BEID+THETA I
GERD =GERD+PHI R
GEID =GEID+PHI I
GOTO 202
201 BERD =BERD-THETA R
BEID =BEID-THETA I
GERD =GERD-PHI R
GEID =GEID-PHI I

```

```
202     IALT =-IALT
203     Z=Z*X2
        XL =X*SQRT(2.)
        IF(QREMB.LT.1.0)GOTO 204
        XL=XL-QREMB
204     THETAR =-XL+BERD
        THETAI=-XL+BEID
        Z=SQRT(X)
        X2=VALU12/Z
        Z=VALU13/Z*EXP(THETAR)
        FR =Z*COS(THETAI)
        FI =Z*SIN(THETAI)
        X2 =X2*EXP(BER)
        THETAR =X2*COS(BEI)
        THETAI =X2*SIN(BEI)
        Z =-FR*GERD+FI*GEID
        GERD=Z
        Z =AAA2*EXP(-QREMB)
        GR =Z*SIN(QREMB)
        GI =Z*COS(QREMB)
        BERD =THETAR*GER-THETAI*GEI+GERD*GR-GEID*GI
        BEID =THETAR*GEI+THETAJ*GER+GERD*GI+GEID*GR
        GER =FR
        GEI =FI
        BER =THETAR+GER*GR-GEI*GI
        BEI =THETAI+GER*GI+GEI*GR
        GOTO (6,4),IBACK
        END
```

```

SUBROUTINE CXRED2(A,C,B,D,N,M)
C   ELIMINATION OF VARIABLES IN SYMMETRIC MATRIX A
DOUBLE PRECISION A(400),B(400),C(400),D(400)
J=N+1
W=1.0
IF(M.GT.0)W=-W
IJ=N*J/2
3   J=J-1
IF(J.EQ.M)RETURN
H1=A(IJ)
G1=C(IJ)
X=1.0/(H1*H1+G1*G1)
H1=-H1*X
G1=G1*X
B(J)=H1
D(J)=G1
IJ=IJ-J
K=0
IK=0
C   BEGIN K-LOOP
4   IK=IK+K
I1 =IK+1
K=K+1
IF(K.GT.N)GOTO 3
IF(K.LT.J)GOTO 9
IF(W.LT.0)GOTO 3
IF(K.EQ.J)GOTO 7
I=IK+J
5   H2=A(I)
G2=C(I)
B(K)=H2*H1-G2*G1
D(K)=H2*G1+G2*H1
C   BEGIN I LOOP
I2=IK+K
L=0
DO 6 I=I1,I2
L=L+1
X=B(L)
Y=D(L)
A(I)=A(I)+X*H2-Y*G2
6   C(I)=C(I)+X*G2+Y*H2
IF(K.LT.J)GOTO 4
I=IK+J
A(I) =B(K)
C(I)=D(K)
GOTO 4
C   END -----
7   J=IJ
DO 8 L=1,J
J=J+1
A(I) =B(L)
8   C(I)=D(L)
GOTO 4
9   I=IJ+K
GOTO 5
END

```



```

SUBROUTINE SYM(P,Z,SWITCH,KCIRCT,KK)
DOUBLE PRECISION P(400),Z(400),F1,F2
DOUBLE PRECISION AR(3,3),AI(3,3),FR(3),FI(3)
IF(KCIRCT.EQ.2)GOTO 100
FR(1) =1.0
FI(1)=0.
FR(2) = -0.5
FI(2) =.5*SQRT(3.0)
FR(3) = -.5
FI(3) =-FI(2)
KK =0
63 KI =KK*(KK+1)/2
J=KI+KK
KOLD =KK
KK=KK+3
IF(KK.GT.KCIRCT)GOTO 79
75 L=KI+1
K=0
64 K=K+1
IF(K1.EQ.J)GOTO 76
65 L3=L+2
66 L2=L+1
67 DO 68 I =1,3
F1=P(L)+FR(I)*(P(L2)+P(L3))
F2=FI(I)*(P(L2)-P(L3))
IF(SWITCH.LT.0.0)GOTO 68
F1=F1+FI(I)*(Z(L3)-Z(L2))
F2=F2+Z(L)+FR(I)*(Z(L2)+Z(L3))
68 AR(I,K)=F1
681 AI(I,K)=F2
IF(K.EQ.3)GOTO 69
L=L+KOLD+K
GOTO 64
69 L=KI
K=0
70 K=K+1
DO 72 I=1,3
M=L+I
IF(K1.LT.J)GOTO 71
IF(I.GT.K)GOTO 73
71 P(M)=(AR(I,1)+FR(K)*(AR(I,2)+AR(I,3))+FI(K)*(AI(I,3)-AI(I,2)))/3.
72 Z(M)=(AI(I,1)+FR(K)*(AI(I,2)+AI(I,3))+FI(K)*(AR(I,2)-AR(I,3)))/3.
73 IF(K.EQ.3)GOTO 74
L=L+KOLD+K
GOTO 70
74 KI=KI+3
IF(K1.GT.J)GOTO 63
GOTO 75
76 IF(K.EQ.3)GOTO 65
L2=J+KK-1
L3=L2+KK
IF(K.EQ.2)GOTO 66
L3=L3-1
GOTO 67
79 KK=KK-3

```

C

100

```
RETURN
F1=(P(1)+P(3))*0.5
F2=P(2)
P(2)=(F(1)-P(3))*0.5
P(1)=F1+F2
P(3)=F1-F2
P(5)=P(3)
KK=2
IF(SWITCH.LT.0.)STOP
F1=(Z(1)+Z(3))*0.5
F2=Z(2)
Z(2)=(Z(1)-Z(3))*0.5
Z(1)=F1+F2
Z(3)=F1-F2
Z(5)=Z(3)
RETURN
END
```

```

SUBROUTINE REDU(A,B,N,M)
C   ELIMINATION OF VARIABLES IN SYMMETRIC MATRIX A
DOUBLE PRECISION A(400),B(400)
J=N+1
W=1.0
IF(M.GT.0)W=-W
IJ=N*J/2
3   J=J-1
    IF(J.EQ.M)RETURN
    H1=A(IJ)
    IF(ABS(H1).GT.(.00001))GOTO 6421
C   ERROR
RETURN
6421 H1=-1.0/H1
     B(J) =H1
     IJ=IJ-J
     K=0
     IK=0
C   BEGIN K-LOOP
4   IK=IK+K
     I1 =IK+1
     K=K+1
     IF(K.GT.N)GOTO 3
     IF(K.LT.J)GOTO 9
     IF(W.LT.0)GOTO 3
     IF(K.EQ.J)GOTO 7
     I=IK+J
5   H2=A(I)
     B(K)=H2*H1
C   BEGIN I LOOP
     I2=IK+K
     L=0
     DO 6 I=I1,I2
     L=L+1
6   A(I)=A(I)+B(L)*H2
     IF(K.LT.J)GOTO 4
     I=IK+J
     A(I) =B(K)
     GOTO 4
C   END -----
7   I=IJ
     DO 8 L=1,J
     I=I+1
8   A(I) =B(L)
     GOTO 4
9   I=IJ+K
     GOTO 5
END

```

```

SUBROUTINE OUTPUT(METRIC,P,Z,SWITCH,KMAX,IS,K2)
DOUBLE PRECISION P(400),Z(400)
CHARACTER*4 MILE,KM,TUNIT
KM='KM'
MILE='MILE'
TUNIT=MILE
UNIT =1.0
IF(METRIC.EQ.1)TUNIT=KM
IF(METRIC.EQ.1)UNIT=5280.*12.*2.54/100000.
CH =SWITCH
GOTO(305,306,307,308,332,333),IS
305 WRITE(9,301)TUNIT
301 FORMAT(/,50X,'INVERTED CAPACITANCE MATRIX (DARAF ',A4,1H))
GOTO 5
306 WRITE(9,302)TUNIT
302 FORMAT(/,50X,'INVERTED SUSCEPTANCE MATRIX (OHM-',A4,1H))
GOTO 5
307 WRITE(9,303)TUNIT
303 FORMAT(/,50X,'CAPACITANCE MATRIX (FARAD/',A4,1H))
UNIT =1.0/UNIT
GOTO 5
308 WRITE(9,304)TUNIT
304 FORMAT(/,50X,'SUSCEPTANCE MATRIX(MHO/',A4,1H))
UNIT=1.0/UNIT
GOTO 5
332 WRITE(9,330)TUNIT
330 FORMAT(/,50X,'INVERTED IMPEDANCE MATRIX (MHO-',A4,1H))
GOTO 5
333 WRITE(9,331)TUNIT
331 FORMAT(/,50X,'IMPEDANCE MATRIX (OHM-',A4,1H))
UNIT=1.0/UNIT
GOTO (320,321,322),K2
320 WRITE(9,309)
309 FORMAT(50X,'FOR SYSTEM OF PHYSICAL CONDUCTORS')
GOTO 6
321 WRITE(9,310)
310 FORMAT(50X,'FOR SYSTEM OF EQUIVALENT PHASE CONDUCTORS')
GOTO 6
322 WRITE(9,311)
311 FORMAT(50X,'FOR SYMMETRICAL COMPONENT OF EQUIVALENT PH.CON')
6 IF(K2.EQ.3)GOTO 3
4 K=0
KK=0
KI=0
1 K=K+1
IF(K.GT.KMAX)GOTO 909
ICOUNT=0
I2=KI+1
I3=I2+8
KI=KI+K
7 IF(I3.GT.KI)I3=KI
IF(ICOUNT.EQ.0)GOTO 8
CALL WRIT(P,I2,I3,1,L,UNIT)
10 IF(CH.GT.0.0)CALL WRIT(Z,I2,I3,2,L,UNIT)
IF(I3.EQ.KI)GOTO 4

```

```
      I2=I3+1
      I3=I2+8
      GOTO 7
8      ICOUNT =1
      IF(K2.EQ.3)GOTO 20
      L=K
21     CALL WRIT(P,I2,I3,3,L,UNIT)
      GOTO 10
3      IF(KMAX.EQ.2)GOTO 400
      GOTO 4
400    IF(IS.LE.4)CH=-1.0
      GOTO 4
20     L=KK
      KK=KK+1
      IF(KK.EQ.3)KK=0
      GOTO 21
909    RETURN
      END
```



```
SUBROUTINE WRIT(P,I2,I3,IFLAG,L,UNIT)
DOUBLE PRECISION P(400),R(10)
LN =9
J=0
DO 30 I=I2,I3
J=J+1
30  R(J) =P(I)*UNIT
    GOTO (1,2,3),IFLAG
1   WRITE(LN,11)(R(I),I=1,J)
    RETURN
2   WRITE(LN,22)(R(I),I=1,J)
    RETURN
3   WRITE(LN,33)L,(R(I),I=1,J)
    RETURN
11  FORMAT(4X,9E14.5)
22  FORMAT(4X,9E14.5)
33  FORMAT(1X,I3,9E14.5)
END
```

```
SUBROUTINE MOVEO(IC,N)  
DIMENSION IC(N)  
DO 100 I=1,N  
IC(I)=0  
100 CONTINUE  
RETURN  
END
```

พารามิเตอร์ต่าง ๆ ของอุปกรณ์ในระบบไฟฟ้า

ตาราง ค.1 - Typical Positive-Sequence, 60-Hz. Per Unit Impedance Values for Transformers Rated 15,000 kVA and Above

High-Voltage Winding BIL	Low-Voltage Winding BIL	Percent Reactance	
		Δ or Ungrounded Y	Grounded Y
150	110	5-7.5	—
200	150	5.5-8.25	—
250	150	5.75-8.5	—
350	200	6.25-9.25	—
450	250	7.0-10.5	—
550	350	7.75-11.5	—
650	350	8.25-12.5	—
750	450	9.25-14.0	—
825	450	9.5-14.25	—
900	550	10.25-15.5	—
1050	650	11.0-16.5	—
—	150	—	8.75-13.25
1175	350	—	9.0-13.5
—	650	—	10.0-15.0
—	900	—	11.5-16.5
1300	650	—	10.5-15.75
1425	650	—	11.5-16.5
1550	750	—	11.75-17.5
1675	750	—	12.00-18
1800	750	—	12.0-18
2050	750	—	12.75-19.25
2300	750	—	13.25-19.75

ตาราง ค.2 Typical Values of Magnetizing Current, as a Percentage of Normal Load Current, for Power Transformers

Transformer MVA	Magnetizing Current as Percent of Load Current			
	BIL = 350	650	900	1300
20	0.8	0.90	1.0	1.2
40	0.65	0.74	0.82	0.94
60	0.58	0.65	0.73	0.84
80	0.54	0.61	0.68	0.77
100	0.51	0.59	0.65	0.73
150	0.47	0.53	0.61	0.67
200	—	0.51	0.58	0.64
300	—	0.49	0.55	0.61
500	—	0.47	0.53	0.59

ตาราง ค.3 Capacitance to Ground of Outdoor Bushings

Kilovolts	Ampere Rating	Capacitance Range (pF)
15	600	160-180
	1200	190-220
23	400	200-450
	600	280
	1200	190-450
	2000	280-650
	3000	370-560
34.5	4000	500-620
	400	200-390
	600	150-220
	1200	170-390
46	2000	240-360
	3000	350-620
	400	180-330
	600	150-280
69	1200	170-330
	2000	200-330
	400	180-270
115	600	250
	1200	160-290
	2000	210-320
	800	250-450
138	1200	250-420
	1600	250-430
	800	250-450
161	1200	250-420
	1600	250-460
	800	260-440
196	1200	260-440
	1600	260-440
	800	350-550
330	1200	350-550
	1600	350-550
345	1600	530
500	800-2000	
	BIL: 1050	550
	1175	500
	1300	450
500	800-2000	
	BIL: 1425	500
	1550	500
	1675	520

ตาราง ค.4 Sixty-Hertz Reactance Values of Synchronous Generators

Two-pole turbine generator	0.12-0.21	0.07-0.14
Four-pole turbine generator	0.20-0.28	0.12-0.17
Salient pole generator		
With dampers	0.20-0.50	0.13-0.32
Without dampers	0.20-0.50	0.20-0.50
Synchronous condenser	0.3-0.50	0.19-0.30

ตาราง ค.5 Earth Resistivities

Average of large number of determinations	100 ohm meter
Sea water	0.01-1.0 ohm meter
Swampy ground	10-100 ohm meter
Dry earth	1000 ohm meter
Pure slate	10^7 ohm meter
Sandstone	10^8 ohm meter

ตาราง ค.6 Typical Capacitance Values for Power Cables

Conductor Size (Circ. mils or AWG)	Capacitance-to-Ground (μ F/mile)					
	Shielded, Paper Insulated			Oil-Filled, Paper Insulated		
	15 kV	23 kV	35 kV	35 kV	46 kV	69 kV
4	0.33	—	—	—	—	—
2	0.40	0.32	—	—	—	—
1	0.44	0.35	—	—	—	—
0	0.49	0.39	0.27	—	—	—
00	0.51	0.40	0.29	0.44	0.40	0.32
000	0.55	0.44	0.31	0.48	0.435	0.35
0000	0.60	0.47	0.37	0.55	0.48	0.39
250,000	0.66	0.51	0.39	0.58	0.51	0.41
300,000	0.68	0.54	0.41	0.63	0.55	0.44
350,000	0.74	0.58	0.44	0.68	0.59	0.465
400,000	0.78	0.60	0.465	0.72	0.63	0.49
500,000	0.86	0.68	0.51	0.78	0.685	0.525
600,000	0.92	0.72	0.54	0.83	0.73	0.56
750,000	1.02	0.78	0.59	0.86	0.79	0.61

ตาราง ค.7 Bus Capacitances

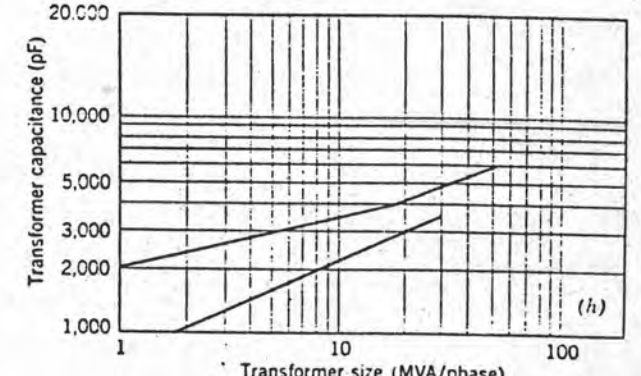
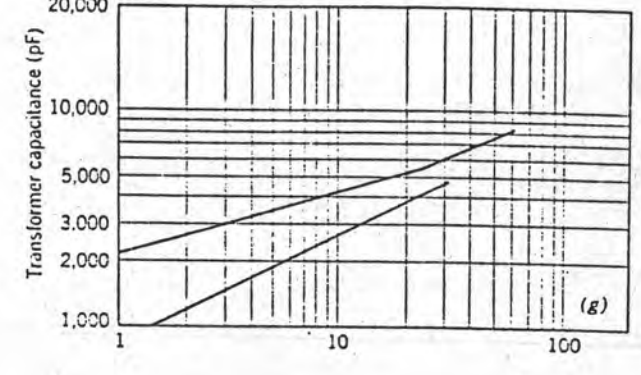
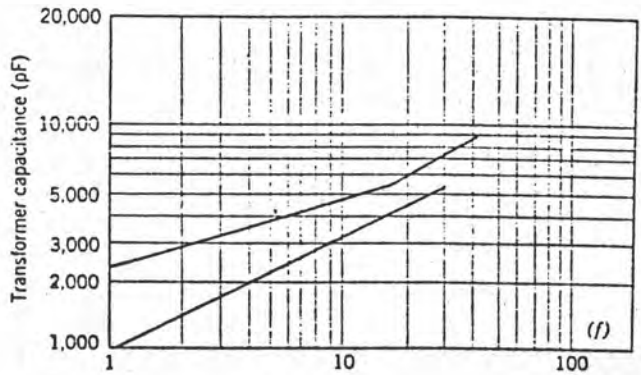
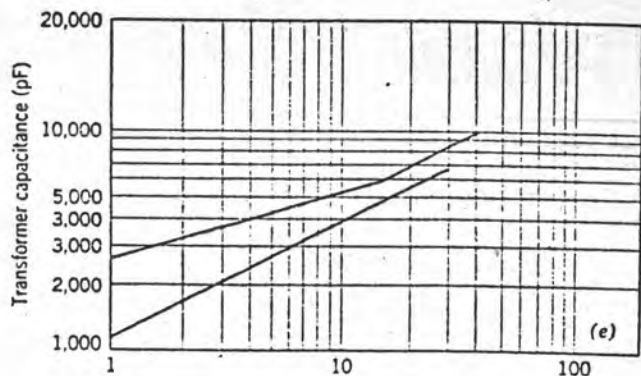
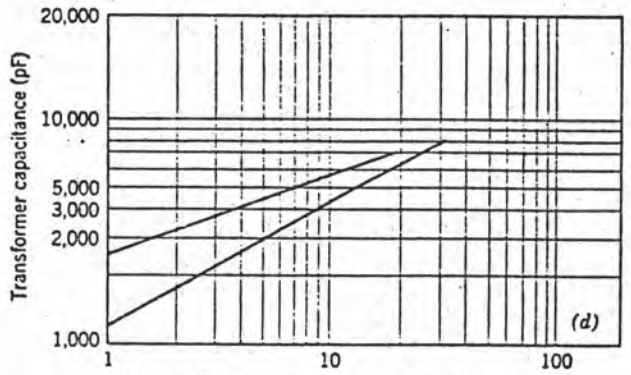
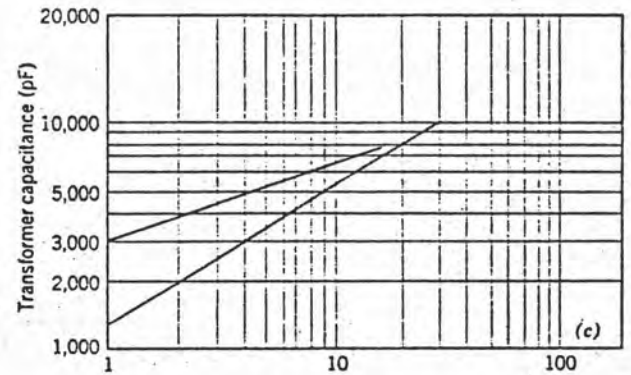
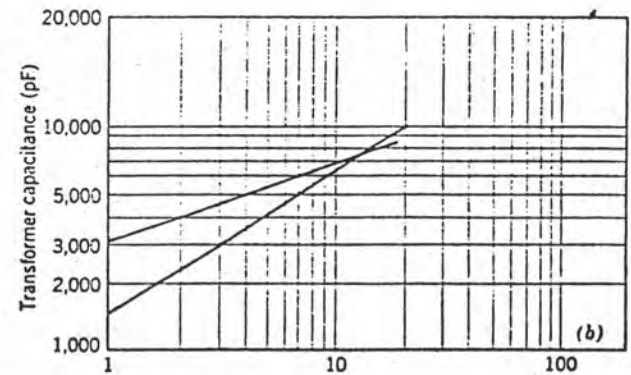
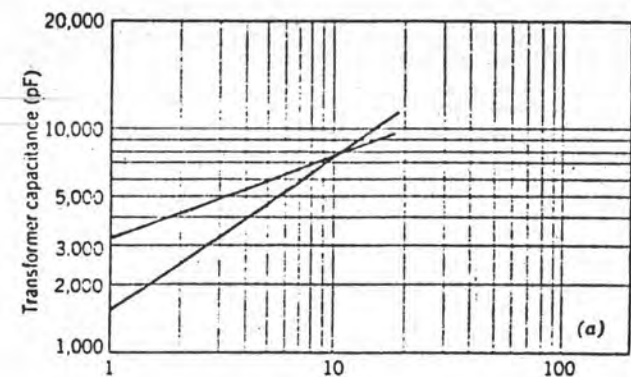
Ampere Rating	Isolated Phase Bus Capacitance		Segregated Phase Bus Capacitance
	15-kV Class 110-kv BIL (pF/ft)	23-kV Class 150-kv BIL (pF/ft)	15-kV Class 110-kv BIL (pF/ft)
1,200	8.9-14.3	8.0-12.4	10.0
2,000	10.2-14.3	9.0-12.4	10.0-10.2
2,500	10.2-14.3	9.0-12.4	
3,000	10.2-14.3	9.0-12.4	10.0-10.2
3,500	10.2-14.3	9.0-12.4	
4,000	14.0-14.3	12.4-13.5	10.0-12.6
4,500	14.0-14.3	12.7-13.5	
5,000	14.0-19.0	12.7-15.8	12.5-14.9
5,500	14.0-19.0	12.7-15.8	
6,000	14.0-19.0	13.5-15.8	15.0-17.1
6,500	14.0-19.0	13.5-15.8	
7,000	17.3-22.6	14.4-17.6	17.1
7,500	17.3-22.6	14.4-17.6	
8,000	21.7	17.6	—
9,000	21.7	18.1	—
10,000	21.7	18.1	—
11,000	23.7	20.5	—
12,000	23.7	20.5	—

ตาราง ค.8 Capacitance of Potential Transformers, Primary Winding to Ground and to Secondary with Terminals Shorted and Grounded

Insulation Class-kV	Capacitance (pF)	
	Line-Line Type	Line-Neutral Type
15	260	
25	250-440	270-800
34.5	310-440	270-900
46	350-430	300-970
69	360-440	340-1300
115	470-520	480-610
138	490-550	530-660
161	510-580	510-700
196	—	580-820
230	600-680	600-810
345	—	920

ตาราง ค.9 Capacitance of Current Transformers, Primary Winding to Ground and Secondary with Terminals Shorted and Grounded

Insulation Class-kV	Capacitance (pF)
25	180-260
34.5	160-250
46	170-220
69	170-260
115	210-320
161	310-380
196	330-390
230	350-420



รูป ค.1 Transformer size (MVA/phase)

.. Transformer winding capacitance to ground range of values for various BIL of highest voltage winding. (a) 110 kV BIL. (b) 150 kV BIL. (c) 200 kV BIL. (d) 250 kV BIL. (e) 350 kV BIL. (f) 450 kV BIL. (g) 550 kV BIL. (h) 650 kV BIL.

การแก้สมการโดยอาศัยเทคนิคการออปติไมเซชัน

ในการแก้สมการโหนด (Node Equation) ของวงจรไฟฟ้า เพื่อหาผลสัมฤทธิ์ทั้งแรงดันและกระแสไฟฟ้า การแก้สมการด้วยการอินเวอร์สเมตริกซ์โดยตรงเป็นวิธีที่ต้องใช้หน่วยความจำและเวลาดำเนินการของคอมพิวเตอร์มาก เพื่อแก้ปัญหาโดยใช้เทคนิค Ordered Triangular Factorization ซึ่งเหมาะกับการใช้สำหรับเมตริกซ์ขนาดใหญ่และอีลิเมนต์ส่วนใหญ่ของเมตริกซ์มีค่าเป็นศูนย์ (Sparse Matrix) ข้อดีอีกข้อหนึ่งของวิธีการนี้คือสามารถลดความคลาดเคลื่อนเนื่องจากการปัดเศษ (Round off Error)

เทคนิคการทำออปติไมเซชันแบบ Ordered Triangular Factorization

วิธีนี้ประกอบด้วย 2 ส่วนหลักดังนี้

1. ส่วนของการจัดลำดับ (Scheme of Ordering) เพื่อคงค่าอีลิเมนต์เดิมที่มีค่าเป็นศูนย์ (Sparse) ให้มากที่สุดเมื่อทำ Factorization อีลิเมนต์ซึ่งไม่อยู่ในตำแหน่งเส้นทแยงมุม (Off Diagonal Element) และมีค่าไม่เท่ากับศูนย์สัมพันธ์กับกิ่ง (Branch) ต่อกันระหว่างโหนด (Node) ที่กำหนดโดยแถวและคอลัมน์ แอตมิตแตนซ์เมตริกซ์เกิดจากการเชื่อมต่อกันระหว่างโหนดที่ต่างกันจึงมีความแตกต่างของอีลิเมนต์ที่ไม่เท่ากับศูนย์ในแต่ละแถว ความแตกต่างของอีลิเมนต์ในแต่ละแถวเป็นตัวบอกถึงวิธีการที่มีประสิทธิภาพในการทำ Triangular Factorization ของแอตมิตแตนซ์เมตริกซ์

วิธีการโปรแกรมจะทำการคำนวณและเก็บค่าอีลิเมนต์ที่มีค่าไม่เท่ากับศูนย์เท่านั้น ซึ่งเป็น การประหยัดการทำงานและหน่วยความจำด้วย วิธีการจัดลำดับเพื่อให้ได้ผลเข้าใกล้ออปติมัล (Optimal) มีหลายแบบ แต่แบบที่เหมาะสมสำหรับการแก้ปัญหาในระบบไฟฟ้ากำลังคือแบบที่ 2 (12) หลักการของการจัดลำดับแบบที่ 2 คือจัดเลขลำดับของแถวต่าง ๆ ใหม่ เพื่อว่าขั้นตอนการคำนวณตามลำดับแถวโดยแถวที่ต้องคำนวณถัดมามีอีลิเมนต์ที่ไม่เท่ากับศูนย์น้อยที่สุดในกลุ่มของแถวที่เหลือ การจัดเก็บข้อมูลที่ไม่เท่ากับศูนย์นี้ไว้ในตารางของแฟกเตอร์ (Table of Factors)

1.1 แพคเตอร์ที่มีผลในการจัดลำดับ

บางครั้งเราต้องจัดลำดับแถวบางแถวไว้ในตำแหน่งสุดท้าย ถึงแม้จะมีผล

ต่อจำนวนเทอมที่มีค่าเป็นศูนย์ให้ลดลง เนื่องจาก

1.1.1 การเปลี่ยนแปลงของระบบเดิมจะเกิดขึ้นเพียงไม่กี่แถวของเมตริกซ์ของระบบที่เราศึกษาอยู่ ถ้าแถวเหล่านี้ถูกกำหนดเลขในลำดับสุดท้าย การแก้ไขเปลี่ยนแปลงเกิดขึ้นกับแถวเหล่านี้ในตารางของแพคเตอร์ (Table of Factors) เท่านั้น เช่น แถวที่มีข้อมูลลัพท์ในการวิเคราะห์ฐานเขียนต์แม่เหล็กไฟฟ้า เป็นต้น

1.1.2 เมตริกซ์ซึ่งประกอบด้วยส่วนของ Nonsymmetric เป็นส่วนน้อย การแยกส่วนของ Nonsymmetric ออกมาสามารถใช้ประโยชน์ของ Symmetric ได้มากขึ้น

1.1.3 การใช้งานในรูปแบบผสม (Hybrid Operation) คือมีตัวแปรที่ไม่ทราบค่าทั้งแรงดันและกระแสไฟฟ้า ดังสมการที่ 1

$$[A] \begin{bmatrix} X_a \\ X_b \end{bmatrix} = \begin{bmatrix} C_a \\ C_b \end{bmatrix} \quad (1)$$

โดย X_a, C_b เป็นตัวแปรที่ไม่ทราบค่า

2. ส่วนของการบันทึกการทำ Triangular Factorization เพื่อใช้งานซ้ำใน

การหาผลลัพท์ที่ช่วงเวลาถัดมา

2.1 การทำ Triangular Factorization ของเมตริกซ์โดยวิธี

Gaussian Elimination (12) เป็นวิธีการกำจัดอีลิเมนต์ต่าง ๆ ที่อยู่ในตำแหน่งต่ำกว่าอีลิเมนต์ของเส้นทแยงมุมของเมตริกซ์นั้น เช่น สมการ

$$Ax = b \quad (2)$$

เมื่อ A เป็นเมตริกซ์แบบ Nonsingular
 x เป็นคอลัมน์เวกเตอร์ที่ไม่ทราบค่า
 b เป็นเวกเตอร์ที่ทราบค่า

การทำ Triangular Factorization ตามขั้นตอนต่อไปนี้เรียกว่า Forward Operation โดยรวมเมตริกซ์ A กับเวกเตอร์ b ในสมการ 3 และเริ่มทำตามขั้นตอนจากขั้นตอนที่ 1 เป็นต้นไป

$$\begin{bmatrix} a_{11} & a_{12} & \cdot & \cdot & a_{1n} & b_1 \\ a_{21} & a_{22} & \cdot & \cdot & a_{2n} & b_2 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ a_{n1} & a_{n2} & \cdot & \cdot & a_{nn} & b_n \end{bmatrix} \quad (3)$$

ขั้นตอนที่ 1 หารอิสิเมนต์ของแถวแรกด้วย a_{11} จะได้

$$\begin{aligned} a_{ij}^{(1)} &= \left(\frac{1}{a_{11}}\right)a_{ij}, \quad j = 2, n \\ b_1^{(1)} &= \left(\frac{1}{a_{11}}\right)b_1 \end{aligned} \quad (4)$$

ค่าในวงเล็บ (Superscript) บอกถึงขั้นตอนที่ค่านั้นเปลี่ยนแปลงครั้งสุดท้าย

ขั้นตอนที่ 2 แสดงในลํกการ 5 และ 6 เป็นการกำจัดอิสิเมนต์ a_{21} ของแถวที่ 2 โดยทำการลบกับผลคูณของ a_{21} กับแถวแรกและแทนค่ากลับในแถว 2 และหารแถวที่ 2 ด้วย

a_{22}

$$\begin{bmatrix} 1 & a_{12}^{(1)} & a_{13}^{(1)} & \cdot & \cdot & \cdot & a_{1n}^{(1)} & b_1^{(1)} \\ & 1 & a_{23}^{(2)} & \cdot & \cdot & \cdot & a_{2n}^{(2)} & b_2^{(2)} \\ & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ & a_{n1} & a_{n2} & a_{n3} & \cdot & \cdot & a_{nn} & b_n \end{bmatrix} \quad (5)$$

$$\begin{aligned} a_{2j}^{(1)} &= a_{2j} - a_{21}a_{1j}^{(1)} \quad j = 2, n \\ b_2^{(1)} &= b_2 - a_{21}b_1^{(1)} \end{aligned}$$

$$a_{2j}^{(2)} = \left(\frac{1}{a_{22}^{(2)}}\right) a_{2j}^{(1)} \quad j = 3, n \quad (6)$$

$$b_2^{(2)} = \left(\frac{1}{a_{22}^{(2)}}\right) b_2^{(1)}$$

ขั้นตอนที่ 3 แสดงตั้งสมการ 7 และ 8 ด้วยวิธีการเดียวกับขั้นตอนที่ 2 แต่เป็นการกำจัดอิลิเมนต์ทางซ้ายของเส้นทแยงมุมของแถวที่ 3

$$\begin{bmatrix} 1 & a_{12}^{(1)} & a_{13}^{(1)} & a_{14}^{(1)} & \cdot & \cdot & a_{1n}^{(1)} & b_1^{(1)} \\ & 1 & a_{23}^{(2)} & a_{24}^{(2)} & \cdot & \cdot & a_{2n}^{(2)} & b_2^{(2)} \\ & & 1 & a_{34}^{(3)} & \cdot & \cdot & a_{3n}^{(3)} & b_3^{(3)} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ a_{n1} & a_{n2} & a_{n3} & a_{n4} & \cdot & \cdot & a_{nn} & b_n \end{bmatrix} \quad (7)$$

$$a_{3j}^{(1)} = a_{3j} - a_{31} a_{1j}^{(1)} \quad j = 2, n$$

$$b_3^{(1)} = b_3 - a_{31} b_1^{(1)}$$

$$a_{3j}^{(2)} = a_{3j}^{(1)} - a_{32}^{(1)} a_{2j}^{(2)} \quad j = 3, n \quad (8)$$

$$b_3^{(2)} = b_3^{(1)} - a_{32}^{(1)} b_2^{(2)}$$

$$a_{3j}^{(3)} = \left(\frac{1}{a_{33}^{(2)}}\right) a_{3j}^{(2)} \quad j = 4, n$$

$$b_3^{(3)} = \left(\frac{1}{a_{33}^{(2)}}\right) b_3^{(2)}$$

ทำตามขั้นตอนเหล่านี้จนครบขั้นตอนที่ n เป็นอันจบ Forward Operation ได้ผล

ตั้งสมการ 9

$$\begin{bmatrix}
 1 & a_{12}^{(1)} & \cdot & \cdot & a_{2n}^{(2)} & b_2^{(2)} \\
 & 1 & \cdot & \cdot & a_{2n}^{(2)} & b_2^{(2)} \\
 & & \cdot & \cdot & \cdot & \cdot \\
 & & & \cdot & \cdot & \cdot \\
 & & & & 1 & b_n^{(n)}
 \end{bmatrix} \quad (9)$$

หลังจากผ่านขั้นตอนที่ K แถวที่ 1 ถึงแถวที่ K ได้ผลตามที่ต้องการ ส่วนแถว K+1 ถึงแถว N ยังต้องผ่านขั้นตอนต่อไปถึง N จึงจะได้ผลตามสมการ 9

ผลลัพธ์ได้จากการแทนค่าย้อนกลับ (Back Substitution) ดังนี้

$$\begin{aligned}
 X_n &= b_n^{(n)} \\
 X_{n-1} &= b_{n-1}^{(n-1)} - a_{n-1,n}^{(n-1)} X_n \\
 X_i &= b_i^{(i)} - \sum_{j=i+1}^n a_{ij}^{(i)} X_j
 \end{aligned} \quad (10)$$

การแทนค่าย้อนกลับเริ่มจาก X_n กลับไปถึง X_1

2.2 การบันทึกการทำ Triangular Factorization

การทำ Forward Operation จากหัวข้อ 2.1 ถูกบันทึกไว้เพื่อใช้งานซ้ำอีก ข้อมูลของการบันทึกนี้และข้อมูลของ Upper Triangle จากสมการ 9 ใช้แทนค่าย้อนกลับเพื่อหาผลลัพธ์สำหรับเวกเตอร์ b ใด ๆ โดยไม่ต้องทำ Triangular Factorization ซ้ำอีก แต่ละขั้นตอนของ Forward Operation บอกโดยตำแหน่งของแถวและคอลัมน์และค่าของอีลิเมนต์ $a_{ij}^{(j-1)}$, $i > j$ ซึ่งเกิดขึ้นในขั้นตอนนั้น

2.2.1 กฎของการบันทึก Forward Operation ของ Triangular Factorization คือ

ก. เมื่อเทอม $1/a_{ii}^{(i-1)}$ ใช้ตำแหน่งเก็บค่าไว้ในตำแหน่ง

(Location) ii

ข. เก็บเทอม $a_{ij}^{(j-1)}, i > j$ ใน Lower Triangle

ผลสุดท้ายของการทำ Triangular Factorization เมตริกซ์ A และการบันทึก

Forward Operation แสดงในสมการ 11

$$\begin{array}{cccccc}
 d_{11} & u_{12} & u_{13} & \cdot & \cdot & u_{1n} \\
 l_{21} & d_{22} & u_{23} & \cdot & \cdot & u_{2n} \\
 l_{31} & l_{32} & d_{33} & \cdot & \cdot & u_{3n} \\
 \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
 l_{n1} & l_{n2} & l_{n3} & \cdot & \cdot & d_{nn}
 \end{array} \quad (11)$$

อีลีเมนต์ของสมการ 11 นิยามในเทอมของการสร้างเมตริกซ์ A จากสมการ 3-9 คือ

$$\begin{aligned}
 d_{ii} &= \frac{1}{a_{ii}^{(i-1)}} \\
 u_{ij} &= a_{ij}^{(i)} \quad i < j \\
 l_{ij} &= a_{ij}^{(j-1)} \quad i > j
 \end{aligned} \quad (12)$$

การบันทึกการทำงานถูกเก็บในตารางของแฟคเตอร์ (Table of Factors) ผลของการอินเวอร์สเมตริกซ์ได้จากผลคูณของ Lower และ Upper Triangular

2.3 การคำนวณผลลัพธ์โดยตรง (Direct Solution) (12)

$$\begin{aligned}
 D_i &: \text{แถว } i = (0, 0, \dots, 0, d_{ii}, 0, \dots, 0, 0) \\
 L_i &: \text{คอลัมน์ } i = (0, 0, \dots, 0, 1, -l_{i+1,i}, -l_{i+2,i}, \dots, -l_{n-1,i}, -l_{n,i})^t \\
 L_i^* &: \text{แถว } i = (-l_{i,1}, -l_{i,2}, \dots, -l_{i,i-1}, 1, 0, \dots, 0, 0) \\
 U_i &: \text{แถว } i = (0, 0, \dots, 0, 1, -u_{i,i+1}, -u_{i,i+2}, \dots, -u_{i,n-1}, -u_{i,n}) \\
 U_i^* &: \text{คอลัมน์ } i = (-u_{1,i}, -u_{2,i}, \dots, -u_{i-1,i}, 1, 0, \dots, 0, 0)^t
 \end{aligned} \quad (13)$$

การอินเวอร์สเมตริกซ์ D_i คือส่วนกลับของอีลิเมนต์ d_{ii} ส่วนการอินเวอร์สของเมตริกซ์ L_i, L_i^*, U_i และ U_i^* เกี่ยวข้องกับการกลับของเครื่องหมายของ Off-diagonal

การทำ Forward และ Back Substitution กับคอลัมน์เวกเตอร์ b แสดงโดยการคูณกันของเมตริกซ์ D_i, L_i หรือ L_i^* และ U_i หรือ U_i^* ดังนั้นการหาผลลัพธ์ของสมการ 2 แสดงในสมการ 14-17 ดังนี้

$$U_1 U_2 \dots U_{n-2} U_{n-1} D_n L_{n-1} D_{n-1} L_{n-2} \dots L_2 D_2 L_1 D_1 b = A^{-1} b = x \quad (14)$$

$$U_1 U_2 \dots U_{n-2} U_{n-1} D_n L_n^* D_{n-1} L_{n-1}^* \dots L_3^* D_2^* L_2^* D_1^* b = A^{-1} b = x \quad (15)$$

$$U_2^* U_3^* \dots U_{n-1}^* U_n^* L_{n-1} D_{n-1} L_{n-2} \dots L_2 D_2 L_1 D_1 b = A^{-1} b = x \quad (16)$$

$$U_2^* U_3^* \dots U_{n-1}^* U_n^* D_n L_n^* D_{n-1} L_{n-1}^* \dots L_3^* D_2^* L_2^* D_1^* b = A^{-1} b = x \quad (17)$$

2.3.1 ตามทฤษฎีของเมตริกซ์ในการอินเวอร์สและทรานโพส (Transpose)

คือ

ทฤษฎี 1 การอินเวอร์สผลคูณของเมตริกซ์แพคเตอร์เท่ากับผลคูณของอินเวอร์สของแพคเตอร์ในลำดับที่กลับกัน

ทฤษฎี 2 การ Transpose ของผลคูณของเมตริกซ์เท่ากับผลคูณของการ Transpose ของแพคเตอร์ในลำดับที่กลับกัน

เมื่อใช้ทฤษฎี 1 กับสมการ 14 จะได้

$$D_1^{-1} L_1^{-1} D_2^{-1} L_2^{-1} \dots L_{n-1}^{-1} D_n^{-1} U_{n-1}^{-1} U_{n-2}^{-1} \dots U_2^{-1} U_1^{-1} x = Ax = b \quad (18)$$

กรณีแก้สมการ Transpose ของ $A^t y = c$ เมื่อใช้ทฤษฎี 2 กับสมการ 14 จะได้

$$D_1^t L_1^t D_2^t L_2^t \dots L_{n-1}^t D_n^t U_{n-1}^t U_{n-2}^t \dots U_2^t U_1^t c = (A^t)^{-1} c = y \quad (19)$$

2.3.2 เมตริกซ์ Symmetry

ถ้าเมตริกซ์ A Symmetry มีเพียงเทอม d_{ii} และ u_{ij} ของ Table of Factor ที่ต้องการ กรณี Symmetric สามารถหา L_i ได้จาก

$$L_i = D_i U_i^t D_i^{-1} \quad (20)$$

การแทนสมการ 18 ในสมการ 14

$$U_1 U_2 \cdots U_{n-2} U_{n-1} D_n D_{n-1} U_{n-1}^t \cdots D_2 U_2^t D_1 U_1^t b = A^{-1} b = x \quad (21)$$

เนื่องจากผลคูณของ $U_i^t D_j$ เมื่อ $i > j$ สามารถทำคอมมิวเตชันได้และ D แพคเตอร์สามารถจัดกลุ่มเป็นเมตริกซ์ที่มีอีลิเมนต์แนวเส้นทแยงมุม

$$D = D_1 D_2 \cdots D_n \quad (22)$$

$$U_1 U_2 \cdots U_{n-2} U_{n-1} D U_{n-1}^t \cdots U_2^t U_1^t b = A^{-1} b = x \quad (23)$$

คุณสมบัติของ Symmetry สามารถลดการทำ Triangular Factorization ลง
เกือบครึ่ง

2.4 การโปรแกรม (Programming)

วิธีการของโปรแกรมจะเก็บค่าและทำการคำนวณเฉพาะอีลิเมนต์ที่มีค่าไม่เท่ากับศูนย์เท่านั้น ดังนั้นจึงต้องมีตารางดัชนีบอกถึงตำแหน่งที่เก็บค่าของอีลิเมนต์นั้น ถ้าเมตริกซ์เป็นแบบ Nonsymmetric จะต้องเก็บค่าของอีลิเมนต์ใน Lower Triangle ด้วย

สำหรับเมตริกซ์แบบ Symmetric ไม่ต้องเก็บค่าอีลิเมนต์ใน Lower Triangle
ดังตัวอย่างการเก็บข้อมูลตาราง ง.1

โดยที่ LOC เป็นแอดเดรสสัมพัทธ์ (Relative Address) ในหน่วยความจำ

D TABLE ประกอบด้วยข้อมูลของอีลิเมนต์ d_{ii}

U TABLE ประกอบด้วยข้อมูลของอีลิเมนต์ U_{ij} ของแถว i โดยมี J บอก
คอลัมน์ Subscript ของ U



D TABLE			U TABLE		
LOC	D Factor	LOC in U table	LOC	U Factor	J
1	d_{11}	1	1	U_{12}	2
2	d_{22}	3	2	U_{17}	7
3	d_{33}	6	3	U_{23}	3
4	d_{44}	8	4	U_{26}	6
5	d_{55}	10	5	U_{27}	7
6	d_{66}	11	6	U_{36}	6
7	d_{77}	12	7	U_{37}	7
			8	U_{45}	5
			9	U_{46}	6
			10	U_{56}	6
			11	U_{67}	7

ตาราง ง.1 การเก็บและ Index Table of Factors

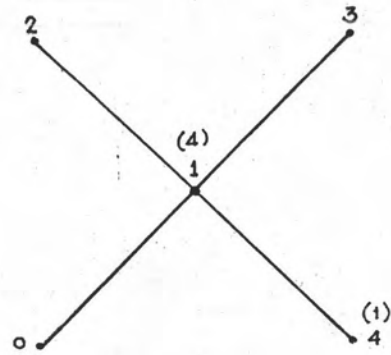
2.5 ตัวอย่างการทำ Ordered Triangular Factorization

2.5.1 ตัวอย่างแสดงถึงความสำคัญของการจัดลำดับ (Ordering)

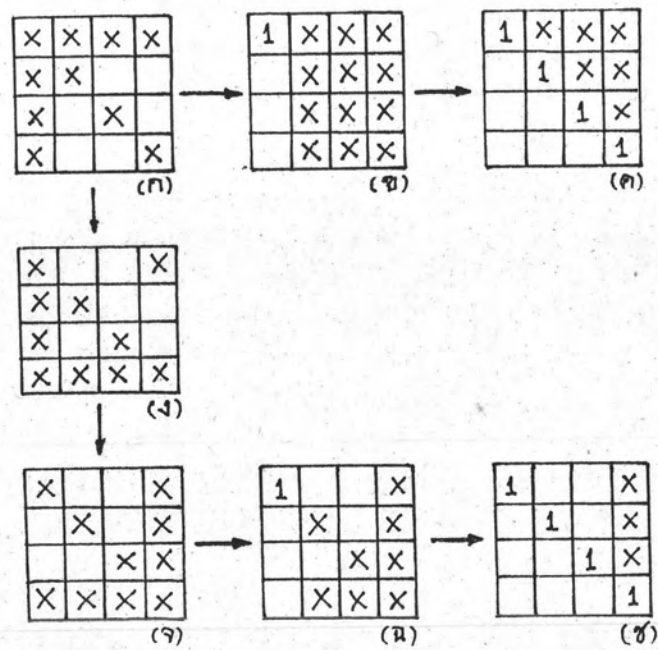
รูป ง.1 แสดงถึง Topology ของวงจรไฟฟ้าประกอบด้วย 5 โหนด เมื่อเขียนเป็นแอดมิตแตนซ์เมตริกซ์ ดังรูป ง.2 ก. หลังจากนั้นทำ Triangular Factorization ดังรูป ง.2.ข และ ง.2.ค

ในกรณีจัดลำดับเลขของโหนดใหม่โดยสลับโหนด 4 กับโหนด 1 Permutation แอดมิตแตนซ์เมตริกซ์ (สลับแถว 4 กับแถว 1 ดังรูป ง.2.ง และสลับคอลัมน์ 4 กับคอลัมน์ 1 ดังรูป ง.2.จ) และได้ทำ Triangular Factorization ดังรูป ง.2.ข

การเปรียบเทียบการคำนวณที่ใช้ทั้งสองแบบดังนี้



រូប ១.១



រូប ១.២

	การจัลดำตัวเลข	ไม่ได้จัลดำตัวเลข
Division	4	4
Multiplication	3	6
Multiplication-Addition	3	10
Nonzero Element	7	10

รูป ง.3 แสดงถึงเมตริกซ์ซึ่งไม่ได้จัลดำตัวเลขโหนดใหม่ตามวิธีการของหัวข้อ 1 หลังจากการทำ Gaussian Elimination มีเทอมที่ไม่เท่ากับศูนย์เพิ่มอีกมาก ส่วนรูป ง.4 เป็นวงจรเดียวกับรูป ง.3 แต่ได้จัลดำตัวเลขโหนดใหม่มีเทอมที่ไม่เท่ากับศูนย์น้อยลง

2.5.2 ตัวอย่างแสดงการทำ Triangular Factorization และหาค่าผลลัพธ์ของสมการ

$$Ax = b$$

โดย กรณี 1 ทราบค่าเวกเตอร์ b หาค่าเวกเตอร์ x

กรณี 2 ทราบค่าเวกเตอร์ x หาค่าเวกเตอร์ b

และหาค่าผลลัพธ์ของสมการ

$$A^t y = c$$

กรณี 3 ทราบค่าเวกเตอร์ c หาค่าเวกเตอร์ y

	1	2	3	4	5	6	7	8	9	10
1	X	X	X	X						X
2	X						X	X		
3	X					X				
4	X									
5	X					X				
6			X		X					
7		X							X	
8		X						X		
9							X			
10	X									

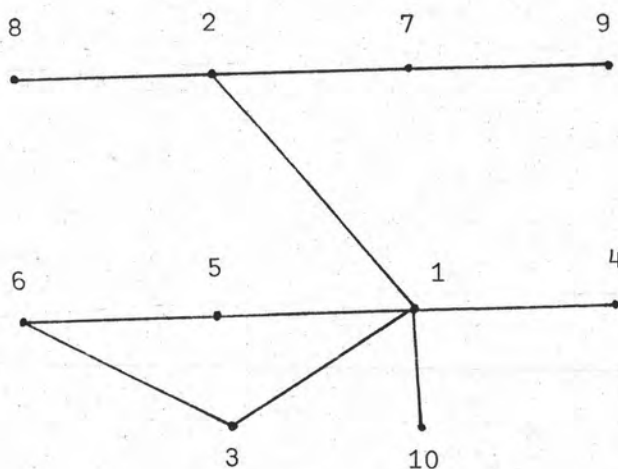
	X	X	X	X						X
X		F	F	F		X	X			F
X	F		F	F	X	F	F			F
X	F	F		F	F	F	F			F
X	F	F	F		X	F	F			F
		X	F	X		F	F			F
	X	F	F	F	F		F	X		F
	X	F	F	F	F	F		F	F	
					X	F				F
X	F	F	F	F	F	F	F	F		

รูป ๓.๓.ก อิลเมนต์ที่ไม่เท่ากับ 0 ของเมตริกซ์ [A] รูป ๓.๓.ข อิลเมนต์ที่ไม่เท่ากับ 0 หลังการทำ

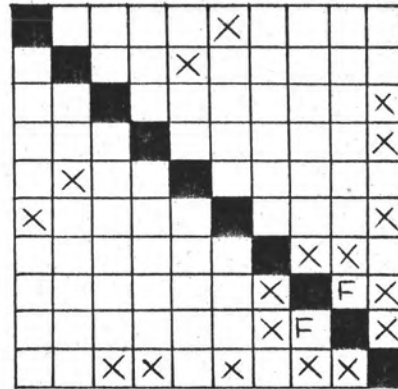
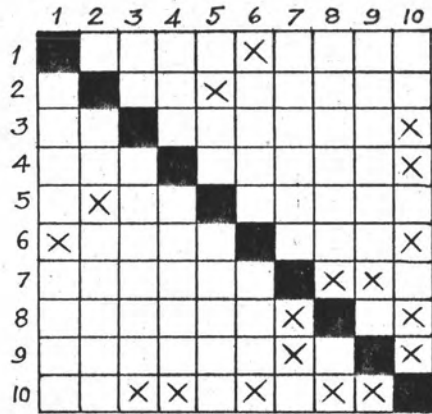
Elimination

F : อิลเมนต์ที่เติมเข้าไประหว่างการทำ

Elimination

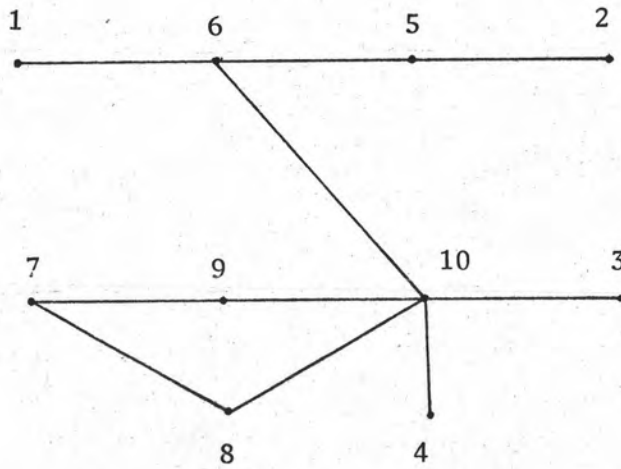


รูป ๓.๓.ค กราฟของวงจรของสมการเต็ม



รูป ง.4.ก อลิเมนต์ที่มีค่าไม่เท่ากับ 0
หลังจากจัดลำดับเลขใหม่

รูป ง.4.ข อลิเมนต์ที่มีค่าไม่เท่ากับ 0
หลังจากทำ elimination



รูป ง.4.ค กราฟของวงจรหลังจากการจัดลำดับตัวเลขใหม่

เมื่อ A เป็นเมตริกซ์ขนาด 3×3 ประกอบด้วยอีลิเมนต์คือ

$$A = \begin{bmatrix} 2 & 1 & 3 \\ 2 & 3 & 4 \\ 3 & 4 & 7 \end{bmatrix}$$

กำหนด $b = \begin{bmatrix} 6 \\ 9 \\ 14 \end{bmatrix}$

ทำการแก้สมการ $Ax = b$ เพื่อหาค่า x ตามวิธีการดังนี้

1. ทำการสัต์ลำดับ (Ordering) แต่เนื่องจากตัวอย่างนี้เมตริกซ์ A ไม่มีเทอมที่เท่ากับศูนย์เลย จึงข้ามขั้นตอนนี้ไป
2. ทำ Triangular Factorization และบันทึกการทำงานไว้ ในตารางของแฟคเตอร์ (Table of Factors) ดังนี้

$$\begin{aligned} 2x_1 + x_2 + 2x_3 &= 6 \\ 2x_1 + 3x_2 + 4x_3 &= 9 \\ 3x_1 + 4x_2 + 7x_3 &= 14 \end{aligned}$$

ทำตามขั้นตอนที่ 1 จากสมการ 4 จะได้

$$\begin{aligned} 1x_1 + \frac{1}{2}x_2 + \frac{3}{2}x_3 &= 3 \\ 2x_1 + 3x_2 + 4x_3 &= 9 \\ 3x_1 + 4x_2 + 7x_3 &= 14 \end{aligned}$$

ทำตามขั้นตอนที่ 2 จากสมการ 6 จะได้

$$\begin{aligned} 1x_1 + \frac{1}{2}x_2 + \frac{3}{2}x_3 &= 3 \\ &2x_2 + x_3 = 3 \\ &3x_1 + 4x_2 + 7x_3 = 14 \\ \text{และ} &1x_1 + \frac{1}{2}x_2 + \frac{3}{2}x_3 = 3 \end{aligned}$$

กรณี 2 เมื่อกำหนดค่า x เราสามารถหาค่า b จากสมการ 18

$$D_1^{-1} L_1^{-1} D_2^{-1} L_2^{-1} D_3^{-1} U_2^{-1} U_1^{-1} x = b$$

$$\begin{bmatrix} 2 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} 1 & & \\ & 2 & 1 \\ & 3 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ & & \\ & & 2 \end{bmatrix} \begin{bmatrix} 1 \\ & & \\ & & 1 \end{bmatrix} \begin{bmatrix} 1 \\ & & \\ & & \frac{5}{4} \end{bmatrix} \begin{bmatrix} 1 \\ & & \\ & & 1 \end{bmatrix} \begin{bmatrix} 1 & & \\ & 1 & \frac{1}{2} \\ & & 1 \end{bmatrix} \begin{bmatrix} 1 & & \\ & 1 & \frac{3}{2} \\ & & 1 \end{bmatrix} \begin{bmatrix} x \\ 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} b \\ 6 \\ 9 \\ 14 \end{bmatrix}$$

กรณี 3 เมื่อกำหนดค่า c ตามสมการ $A^t y = c$ สามารถหาค่า y จากสมการ 3.18

$$D_1^t L_1^t D_2^t L_2^t D_3^t U_2^t U_1^t c = y$$

$$\begin{bmatrix} \frac{1}{2} \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} 1 & -2 & -3 \\ & 1 & \\ & & 1 \end{bmatrix} \begin{bmatrix} 1 \\ & & \\ & & \frac{1}{2} \end{bmatrix} \begin{bmatrix} 1 \\ & & \\ & & 1 \end{bmatrix} \begin{bmatrix} 1 \\ & & \\ & & -\frac{5}{2} \\ & & & 1 \end{bmatrix} \begin{bmatrix} 1 \\ & & \\ & & \frac{4}{5} \\ & & & -\frac{1}{2} \end{bmatrix} \begin{bmatrix} 1 \\ & & \\ & & 1 \\ & & & -\frac{3}{2} \end{bmatrix} \begin{bmatrix} 1 \\ & & \\ & & 1 \\ & & & 1 \end{bmatrix} \begin{bmatrix} c \\ 9 \\ 9 \\ 17 \end{bmatrix} = \begin{bmatrix} y \\ 2 \\ 1 \\ 1 \end{bmatrix}$$

โปรแกรมการวิเคราะห์ทรานเซียนต์ในระบบไฟฟ้ากำลัง

```
C -----  
C  
C MAIN PROGRAM FOR ELECTROMAGNETIC TRANSIENT ANALYSIS  
C IN ELECTRIC POWER SYSTEM  
C DEVELOPED BY  
C  
C V. CHIVARANUNTAKUL  
C -----
```

```
C DECLARATION
```

```
*INCLUDE: 'COMMON.FOR'  
CHARACTER*80 TITLE,BUFF  
CHARACTER*12 FNAME  
CHARACTER*6 BUS1,BUS2,AUP,BUS,OVR
```

```
C  
C SET CONSTANT VARIABLE  
C -----  
C
```

```
LI=7  
LO=8  
KILL=0  
LBRNCH=100  
LBUS=100  
LSWTCH=30  
LNONL=30  
LSOURCE=30  
LNODE=10  
LYMAT=200  
LSMAT=100  
NTOT=0  
IBR=0  
KSWTCH=0  
INONL=0  
IT=0  
TWOPI=2.0*3.14159
```

```
C  
C CLEAR MEMORY  
C
```

```
CALL MOVERO(TR,100)  
CALL MOVERO(TX,100)  
CALL MOVERO(C,100)  
CALL MOVERO(NR,100)  
CALL MOVERO(LENGTH,100)  
CALL MOVERO(KBUS,100)  
CALL MOVERO(HBUS,100)
```

C
C ----- END SET CONSTANT PART -----
C

162

```
10  WRITE(*,10)
    FORMAT(' WOULD YOU LIKE help WHETHER OR NOT (Y/N) 1 FOR Y')
    FNAME='HELP.TXT'
    LIMIT=7
    ASSIGN 46 TO ISL
15  READ(*,20)MODE
20  FORMAT(I1)
    IF(MODE.NE.1)GOTO ISL
21  OPEN(9,FILE=FNAME,STATUS='OLD')
24  REWIND 9
    INDX=0
25  CONTINUE
    DO 22 I=1,22
    READ(9,26)BUFF
26  FORMAT(A80)
    WRITE(*,27)BUFF
27  FORMAT(A80)
22  CONTINUE
29  WRITE(*,30)
30  FORMAT(' WOULD YOU LIKE CONTINUE 1-NEXT,0-EXIT')
    READ(*,31)IC
31  FORMAT(I1)
    IF(IC.NE.1)GOTO 35
    INDX=INDX+1
    IF(INDX.GT.LIMIT)GOTO 45
    GOTO 25
35  IF(IC.EQ.0)GOTO 45
    GOTO 29
45  CLOSE(9,STATUS='KEEP')
    IF(ISL.EQ.50)GOTO 50
46  WRITE(*,47)
47  FORMAT(' WOULD YOU LIKE example WHETHER OR NOT (Y/N) 1 FOR Y')
    FNAME='EXAM.TXT'
    LIMIT=2
    ASSIGN 50 TO ISL
    GOTO 15
50  CONTINUE
    WRITE(*,30)
    READ(*,31)IC
    IF(IC.EQ.0)GOTO 99999
    IF(IC.NE.1)GOTO 50
    WRITE(*,60)
60  FORMAT(' NAME OF INPUT FILE :')
    READ(*,'(A12)')FNAME
    OPEN(7,FILE=FNAME,STATUS='OLD')
    REWIND 7
70  CONTINUE
    WRITE(*,80)
80  FORMAT(' NAME OF OUTPUT FILE :')
    READ(*,'(A12)')FNAME
    OPEN(8,FILE=FNAME,STATUS='NEW')
    REWIND 8
```

```

WRITE(*,90)
90  FORMAT(' NAME OF PLOT FILE : ')
    READ(*,'(A12)')FNAME
    OPEN(9,FILE=FNAME,STATUS='NEW')
    REWIND 9
C  -----HEADING-----
    WRITE(8,100)
100  FORMAT(5X,65(1H*))
    WRITE(8,115)
115  FORMAT(5X,1H*,63X,1H*)
    WRITE(8,120)
120  FORMAT(5X,1H*,5X,'A COMPUTER METHOD OF ELECTROMAGNETIC TRANSIENT
    IANALYSIS',2X,1H*,/,5X,1H*,19X,'IN ELECTRIC POWER SYSTEM',20X,1H*)
    WRITE(8,115)
    WRITE(8,100)
    WRITE(8,125)
125  FORMAT(1X,/,1X,80(1H=))
C  -----
C  ---- START PROGRAM PROCEDURE ----
C  -----
    CALL READINP
    IF(KILL.GT.0)GOTO 92000
    CALL CONECT
    IF(KILL.GT.0)GOTO 92000
    CALL RENUMB
    IF(KILL.GT.0)GOTO 92000
    CALL CONVRT
    IF(KILL.GT.0)GOTO 92000
    CALL PREPTB
    IF(KILL.GT.0)GOTO 92000
    CALL FORMY
    IF(KILL.GT.0)GOTO 92000
    CALL SETUP
    IF(KILL.GT.0)GOTO 92000
    CALL TSTEP
    IF(KILL.LE.0)GOTO 99999
92000 WRITE(LO,92005)OVR
92005 FORMAT(1X,'ERROR FIND OUT AT SUBROUTINE',5X,A10)
99999 STOP
    END

```



```
COMMON LI,LO,IPR,KILL,MODE,KPARTA,KPARTB
COMMON LBUS,LBRNCH,LSWTCH,LNONL
COMMON NTOT,IBR,KSWTCH,INONL,IT,MXPHAS,LYMAT,LSMAT
COMMON KBUS(100),MBUS(100),NR(100),LENGTH(100)
COMMON KOLUM(100),KORDER(100),LOC(100)
COMMON KCTRL(100),KOWNT(100)
COMMON NLTYPE(30),NONLK(30),NONLM(30),NUM,NONLE(100),NONLAD(100)
COMMON KHIGH(30),KLOW(30),IFORM(30),NODE(30),ISOURC(30)
COMMON KSSFRB(100),KPOS(30),KPOS2(30)
COMMON IBRNCH(20),JBRNCH(20),IBSOUT(20),NCH
COMMON TR(100),TX(100)
COMMON DELTAT,DELTA2,TWOPI,QFD(30)
COMMON QVR,BUS(100)
COMMON C(100),R(100),X(100)
COMMON GSLOPE(100),CCHAR(100),VCHAR(100)
COMMON VNONL(30),ANONL(30),CURR(30),VZERO(30)
COMMON VECNL1(30),VECNL2(30)
COMMON TIMEI(30),TSTART(30),TSTOP(30),SFREQ(30),FKPOS(30)
COMMON TCLOSE(30),TOPEN(30),CRIT(30),ADELAY(30)
COMMON CREST(30),FKPOS1(30)
COMMON FSWTCH(30)
COMMON CI(100),CK(100),CIK(100),CKI(100),CKKJM(100)
COMMON VOLT(100),VOLTI(100),VOLTK(100)
COMMON KM(100),YKM(100),KKS(100),KK(100),KMS(100)
COMMON FINIT(100),ETA(100),F(100),E(100),YS(100)
COMMON BVALUE(100),XMAX(20)
COMMON AUP(9)
```

SUBROUTINE READINP

```

C
C READ ALL INPUT DATA FROM INPUT FILE
C ARRANGE THE DATA TO FIT FOR NEXT PART
C
$INCLUDE: 'COMMON.FOR'
      CHARACTER*6 BUS1,BUS2,BUS3,BUS4,BLANK
      CHARACTER*6 BUS,AUP
      CHARACTER*80 TITLE
      BLANK='
      CALL MOVERO(CI,100)
      CALL MOVERO(CK,100)
      CALL MOVERO(CIK,100)
      CALL MOVEO(KCTRL,100)
      DO 89 I=1,100
89     BUS(I)=BLANK
      OVR='READIN'
      KCOUNT=1
      ICHECK=1
100    CONTINUE
      WRITE(LO,110)
110    FORMAT(/,' CASE STUDY TITLE',/, ' =====')
      DO 120 I=1,3
      READ(LI,150)TITLE
150    FORMAT(A80)
      WRITE(LO,200)I,TITLE
200    FORMAT(' LINE #',I2,3X,A80)
120    CONTINUE
C
C INPUT CONTROL CODE DATA (1)
C -----
      READ(LI,*,ERR=9200)DELTAT,TMAX,XOPT,COPT,IPR,IPRNT,IPLLOT
      WRITE(LO,220)DELTAT,TMAX,XOPT,COPT,IPR,IPRNT,IPLLOT
220    FORMAT(' CONTROL CODE DATA',/,1X,17(1H=),/,5X,
1' <DELTAT> <TMAX> <XOPT> <COPT> <IPR> <IPRNT> <IPLLOT>',/,
24E15.8,3I5)
C
C READ INPUT BRANCH DATA (2)
C -----
      WRITE(LO,274)
274    FORMAT(' BRANCH DATA',/,1X,11(1H=))
      BUS(1)=BLANK
      KCTRL(1)=-1
      NTOT=1
300    CONTINUE
C
C BUS NAME DATA CANNOT FREE FORMAT
C
301    READ(LI,303)ITYPE,BUS1,BUS2,BUS3,BUS4,IOUT
303    FORMAT(I2,4A6,I1)
      WRITE(LO,320)ITYPE,BUS1,BUS2,BUS3,BUS4,IOUT
320    FORMAT(I4,4A6,I2)
      IF(ITYPE.EQ.88)GOTO 999
      IBR=IBR+1
      IF(BUS1.NE.BLANK.OR.BUS2.NE.BLANK)GOTO 1100

```

```

KILL=1
GOTO 9200
1100 N1=0
      N2=0
      N3=0
      N4=0
      DO 1200 I=1,NTOT
      IF (BUS1.EQ.BUS(I))N1=I
      IF (BUS2.EQ.BUS(I))N2=I
      IF (BUS3.EQ.BUS(I))N3=I
      IF (BUS4.EQ.BUS(I))N4=I
      WRITE(7832)N1,N2,N3,N4
7832 FORMAT(1X,4I2)
1200 CONTINUE
1220 IF (N1.EQ.0)THEN
      NTOT=NTOT+1
      BUS(NTOT)=BUS1
      KBUS(1BR)=NTOT
      ELSE
      KBUS(1BR)=N1
      ENDIF
      IF (N2.EQ.0)THEN
      NTOT=NTOT+1
      BUS(NTOT)=BUS2
      MBUS(1BR)=NTOT
      ELSE
      MBUS(1BR)=N2
      ENDIF
      IF (IOUT.GT.0)MBUS(1BR)=-MBUS(1BR)
      IF (N3.EQ.0.OR.N4.EQ.0)GOTO 1600
      IF (N3.EQ.1.AND.N4.EQ.1)GOTO 1600
C
C ---- REPEAT ANOTHER BRANCH VALUE
C
      DO 1300 IX=1,1BR
      IF (N3.EQ.KBUS(IX).AND.N4.EQ.MBUS(IX)) THEN
      NR(1BR)=NR(IX)
      LENGTH(1BR)=-1
      GOTO 300
      ENDIF
1300 CONTINUE
      GOTO 300
C -----
C NEW BRANCH DATA
C -----
1600 CONTINUE
      IF (ITYPE.EQ.0)GOTO 8000
      IF (ITYPE.LT.0)GOTO 8300
      IF (ITYPE.GT.0.AND.ITYPE.LT.10)THEN
      GOTO (8200,8162),ICHECK
      ENDIF
      IF (ITYPE.GT.90.AND.ITYPE.LE.99)GOTO 8500
      KILL=2
      GOTO 9200
C

```

C < SERIES R-L-C >

```

C -----
8000 CONTINUE
      IT=IT+1
      READ(LI,*,ERR=9200)TR(IT),TX(IT),C(IT)
      WRITE(LO,8100)IT,TR(IT),TX(IT),C(IT)
8100 FORMAT(' SERIES R=L=C ',12,2X,3E11.5)
8150 D1=ABS(TR(IT))+ABS(TX(IT))+ABS(C(IT))
      IF(D1.NE.0.0)GOTO 422
      KILL=3
      GOTO 9200
422  NR(IBR)=IT
      LENGTH(IBR)=1
      GOTO 300
8200 CONTINUE

```

C

C < PI EQUIVALENT CIRCUIT DATA >

C

```

      READ(LI,*,ERR=9200)TR(IT),TX(IT),C(IT)
      WRITE(LO,8205)TR(IT),TX(IT),C(IT)
8205 FORMAT(' 1 ST OF PI EQUIVALENT DATA',1X,3E11.3)
      ICHECK=2
      GOTO 8150
8162 CONTINUE
      KCOUNT=KCOUNT+1
      IF(ITYPE.NE.KCOUNT)THEN
      ICHECK=1
      KCOUNT=1
      GOTO 1600
      ENDIF
      IT2=IT+ITYPE-1
      DO 8346 IT32=IT,IT2
      IF(IT32.LE.IT)GOTO 9200
      READ(LI,*,ERR=9200)TR(IT32),TX(IT32),C(IT32)
      WRITE(LO,8347)ITYPE,TR(IT32),TX(IT32),C(IT32)
8347 FORMAT(' PHASE',13,3E13.5)
8346 CONTINUE
      GOTO 300

```

C

C < DISTRIBUTED BRANCH PARAMETER >

C

```

8300 CONTINUE
      IF(IABS(ITYPE).EQ.1)THEN
      KCOUNT=1
      LENGTH(IBR)=1
      GOTO 8303
      ENDIF
      IF(IABS(ITYPE).GT.1)THEN
      KCOUNT=KCOUNT+1
      LENGTH(IBR)=IABS(ITYPE)
      GOTO 8303
      ENDIF
8310 K=1
      N9=IABS(LENGTH(IBR))
      N99=N9*N9

```

```

DO 5960 J=1,N9
DO 5955 I=1,N9
VOLTI(K)=1
IF(I.EQ.J.AND.I.NE.1)VOLTI(K)=1-I
IF(I.GT.J.AND.J.NE.1)VOLTI(K)=0.0
5955 K=K+1
5960 CONTINUE
      NI=1
      DO 5970 I=1,N99,N9
      K=I+N9-1
      TEMP=0
5965 DO 5965 J=1,K
      TEMP=TEMP+VOLTI(J)**2
      TEMP =SQRT(TEMP)
      DO 5968 J=1,K
      QFD(NI)=VOLTI(J)/TEMP
5968 NI=NI+1
5970 CONTINUE
      GOTO 8330
8303 CONTINUE
      READ(LI,*)H1,AA,H3,XLONG,ILINE
      WRITE(LO,8305)H1,AA,H3,XLONG,ILINE
8305 FORMAT(5X,'< R> <Z-L> <T-C> <IL>',1X,4(F8.3,2X),12)
      IF(KCOUNT.GT.1)GOTO 8310
8330 IF(ILINE.LT.0)GOTO 9200
      IF(ILINE.GT.1)GOTO 9200
      IF(ILINE.NE.1)GOTO 198
5877 CHECK =H1+AA+H3+XLONG
      IF(KCOUNT.GT.3)GOTO 9200
      IF(KCOUNT.NE.3)GOTO 198
      IF(ICHECK.EQ.0)GOTO 9200
      IF(AA.LE.0.0)GOTO 9200
      IF(H3.GT.0.0)GOTO 198
      GOTO 9200
198 IF(ILINE.GT.0)GOTO 192
C CHANGE L,C --> Z AND T
C
      D2=AA*1.E-3
      CI1=H3/1.E6
      IF(XOPT.GT.0.0)D2=D2*1000./(TWOPI*XOPT)
      IF(COPT.GT.0.0)CI1=CI1/(TWOPI*COPT)
      A=SQRT(D2/CI1)
      H2=XLONG*A*CI1
      GOTO 193
192 A=AA
      H2=H3
C
C CI STORE SURGE IMP., CK STORE R, CIK STORE TRAVELING TIME
C
193 CIK(1BR)=H2
      IF(H1.EQ.0.0)A=-A
      CK(1BR)=H1*XLONG
      CI(1BR)=A
      ICHECK=3
      LENGTH(1BR)=KCOUNT

```

```

      WRITE(*,8459)IBR,CK(IBR),CI(IBR),CIK(IBR)
8459  FORMAT(' DISTR. ',15,3E15.5)
      GOTO 300

C
C   NONLINEAR PARAMETER
C   -----
8500  CALL NONLN
      GOTO 300

C
C   < SWITCH DATA -- SOURCE DATA >
C   -----
999   CONTINUE
      KSWTCH=0

9111  CONTINUE
      READ(LI,9004,ERR=9200)IT22,BUS1,BUS2,BUS3,BUS4
9004  FORMAT(12,2A6,2F10.6)
      WRITE(LO,9100)IT22,BUS1,BUS2,BUS3,BUS4
9100  FORMAT(/,' SWITCH DATA ',1X,11(1H=)/,' TYPE ', 'BUS1 ',2X,
1'BUS2 ',2X,' TCLOSE ', ' TOPEN ',/,3X,12,2A8,2E10.4)
      WRITE(*,7023)
7023  FORMAT(' END BRANCH DATA')
      IF(IT22.EQ.88)GOTO 3000
      IF(BUS1.NE.BLANK)GOTO 350
      IF(BUS2.EQ.BLANK.AND.BUS1.EQ.BLANK)GOTO 9200
350   IF(KSWTCH.LE.LSWTCH)GOTO 370
      GOTO 9200
370   KSWTCH=KSWTCH+1
      ADELAY(KSWTCH)=0.0
      FSWTCH(KSWTCH)=0.0
      N1=0
      N2=0
      N3=0
      DO 210 I=1,NTOT
      IF(BUS1.EQ.BUS(I))N1=I
      IF(BUS2.EQ.BUS(I))N2=I
210   CONTINUE
      IF(N1.NE.0)GOTO 2110
      NTOT=NTOT+1
      IF(NTOT.GT.LBUS)GOTO 9200
      N1=NTOT
      BUS(N1)=BUS1
2110  IF(N2.NE.0)GOTO 2120
      NTOT=NTOT+1
      IF(NTOT.GT.LBUS)GOTO 9200
      N2=NTOT
      BUS(N2)=BUS2
2120  ISOURC(KSWTCH)=0
2220  IF(N1.GT.N2)GOTO 2150
      N3=N1
      N1=N2
      N2=N3
2150  KHIGH(KSWTCH)=N1
      KLOW(KSWTCH)=N2
C     IF(AT.EQ.0.0)GOTO 2160
C     GUS4=ABS(AT)

```

```

                IF(GUS3.GE.0.0)GOTO 2160
                GUS3=0.0
2160      KPOS2(KSWTCH)=0
                FKPOS1(KSWTCH)=1122
                TCLOSE(KSWTCH)=GUS3
                TOPEN(KSWTCH)=GUS4
C         CRIT(KSWTCH)=CK1
                GOTO 9111
3000      CONTINUE
                WRITE(*,3021)
3021      FORMAT(' BEGIN SOURCE DATA')
                KCONST=0
C
C   < READ SOURCE DATA >
C   -----
C
C   N2 : TYPE           BUS1 : SOURCE BUS NAME
C   N1 : V OR I        A   : AMPLITUDE
C   D1 : FREQ          GUS2 : T0
C   H1 : AMPL1        H2  : T1
C   GUS3: TSTART      GUS4 : TSTOP
C
3100      IX=0
                READ(L1,3018,ERR=9200)N2,BUS1
3018      FORMAT(I2,A6)
                WRITE(L0,3398)N2,BUS1
3398      FORMAT(' SOURCE DATA',/,1X,(11(1H=)),/, ' TYPE',2X,'BUS1',13,A6)
                IF(N2.EQ.88)GOTO 9990
                KCONST=KCONST+1
                DO 7194 I=1,NTOT
                IF(BUS1.EQ.BUS(I))THEN
                IX=I
                KCTRL(I)=-I
                ENDIF
7194      CONTINUE
                IF(IX.EQ.0)GOTO 9200
                READ(L1,*,ERR=9200)N1,A,D1,GUS2,H1,H2,GUS3,GUS4
                WRITE(*,1093)N1,A,D1,GUS2,H1,H2,GUS3,GUS4
1093      FORMAT(1X,I2,7F12.5)
                IF(GUS4.EQ.0.0)GUS4=1.0E+20
                SFREQ(KCONST)=D1
                IF(IPR.GT.0)
                1WRITE(L0,3101)A,SFREQ(KCONST),GUS2,GUS3
3101      FORMAT(' SOURCE ',2X,4E10.4)
                WRITE(*,3021)
                IF(N2.NE.14)GOTO 3170
                IF(SFREQ(KCONST).GT.0.0)GOTO 3170
                GOTO 9200
3170      CONTINUE
C         IF(N1.GE.0)GOTO 3360
C         I=-I
3         NODE(KCONST)=IX
                TSTOP(KCONST)=GUS4
                IFORM(KCONST)=N2
                CREST(KCONST)=A

```

```

IF(N2.EQ.14)GOTO 3320
IF(GUS3.LT.0.0)GUS3=0.0
C
C SFREQ STORE SLOPE FOR TYPE 13
C
IF(N2.EQ.13)SFREQ(KCONST)=(H1-A)/(H2-GUS2)
GOTO 3330
3320 GUS2=GUS2*TWOPI/360.0
IF(H1.GT.0.)GUS2=GUS2*SFREQ(KCONST)*360.0
IF(GUS3.GE.0.0)GOTO 3330
IF(OMEGA.EQ.0.0)OMEGA=TWOPI*SFREQ(KCONST)
3330 TIME1(KCONST)=GUS2
TSTART(KCONST)=GUS3
K=0
3340 K=K+1
IF(K.EQ.KCONST)GOTO 3100
IF(NODE(K).EQ.1)GOTO 3350
GOTO 3340
3350 IFORM(KCONST)=-N2
GOTO 3100
C -----
3360 CONTINUE
C
C INPUT VOLTAGE SELECTION OUTPUT
C
9990 WRITE(LO,9891)
9891 FORMAT(' SELECT O/P VOLTAGE')
READ(LI,4222)(AUP(I),I=1,9)
4222 FORMAT(9A6)
WRITE(LO,4333)(AUP(I),I=1,9)
4333 FORMAT(1X,'NODE VOLTAGE SELECTION ',9A8)
READ(LI,40917)IFIN
40917 FORMAT(I6)
IF(IFIN.EQ.999999)GOTO 9999
WRITE(LO,4482)
4482 FORMAT(' INPUT ABSENT STOP CODE')
9999 IF(IPR.LE.0)GOTO 9998
WRITE(LO,44000)(I,KBUS(I),MBUS(I),NR(I),LENGTH(I),BUS(I),I=1,IBR)
44000 FORMAT(/,' LINEAR BRANCH TABLE',/,10X,'ROW',5X,'KBUS',5X,
1'MBUS',8X,'NR',5X,'LENGTH',5X,'BUS',/, (1X,5I10,A15))
IF(INONL.LE.0)GOTO 44005
WRITE(LO,44010)(I,NONLK(I),NONLM(I),NLTYPE(I),NONLAD(I),NONLE(I)
1,ANONL(I),VZERO(I),VCHAR(I),GSLOPE(I),CCHAR(I),I=1,INONL)
44010 FORMAT(' NONLINEAR BRANCH TABLE',/,6X,'ROW',3X,'NONLK',
13X,'NONLM',2X,'NLTYPE',2X,'NONLAD',3X,'NONLE',10X,'ANONL'
2,10X,'VZERO',10X,'VCHAR',9X,'GSLOPE',10X,'CCHAR',/,
3(1X,6I8,5E15.5))
44005 WRITE(LO,44100)(I,TR(I),TX(I),C(I),I=1,IT)
44100 FORMAT(/,' LUMPED PARAMETER STORAGE TR,TX,C',/,
1(1X,1I0,3E20.6))
IF(KSWTCH.LE.0)GOTO 44203
WRITE(LO,44200)(I,KHIGH(I),KLOW(I),TCLOSE(I),TOPEN(I),
1CRIT(I),I=1,KSWTCH)
44200 FORMAT(' ROW-KHIGH-KLOW-TCLOSE-TOPEN-CRIT',/, (3I8,3E15.6))
IF(KCONST.LE.0)GOTO 9998

```



```
44203 WRITE(LO,44300)(I,IFORM(I),NODE(I),CREST(I),TIME1(I),TSTART(I),  
1SFREQ(I),I=1,KCONST)  
44300 FORMAT(/,' SOURCE TABLE VECTOR',/, ' ROW-IFORM-NODE-CREST-TIME'  
1, '-TSTART-SFREQ',/, (3I10,4E20.4))  
      GOTO 9998  
9200  CONTINUE  
      WRITE(LO,14441)  
14441 FORMAT(' ERROR READIN')  
9998  RETURN  
      END
```

```

C
C READ NONLINEAR PARAMETER INPUT
C -----
C
C DECLARATION
$INCLUDE: 'COMMON.FOR'
      DATA BLANK /' /
C -----
      OVR='NONL'
      CII=0.0
      CKK=0.0
4221  INONL=INONL+1
C
C   98 : NONLINEAR RESISTANCE
C   99 : NONLINEAR INDUCTANCE
C
      READ(LI,*)TRR,TXX,CIT,TR1
      NLTYPE(INONL)=ITYPE
      NONLK(INONL)=N1
      IF(IOUT.GT.0)N2=-N2
      NONLM(INONL)=N2
      VNONL(INONL)=TRR
      IF(ITYPE.NE.93.AND.ITYPE.NE.98)GOTO 185
C
C --- NONL. INDUCTANCE ---
C   VZERO : Isteady
C   ANONL : FLUXsteady
C
      VZERO(INONL)=TRR
      ANONL(INONL)=TXX
      VNONL(INONL)=0.0
C -----
185  CURR(INONL)=0.0 /
      IF(ITYPE.NE.99)GOTO 7342
      D3=CIT
4257  IF(D3.EQ.0.0)D3=1.0
C INPUT FOR TYPE 99 NONL. R ONLY
C -----
C   VNONL : VFLASH
C   VECNL2 : TDELAY
C   VZERO : VSEAL
C   ANONL : JUMP
C
      ANONL(INONL)=D3
      N10 =D3
      VECNL1(INONL)=TRR
      VECNL2(INONL)=TXX
      NLTYPE(INONL)=-99
C
C NUM : TOTAL NO. OF TYPE 99
C -----
      NUM=NUM+1
      CURR(INONL)=0.0
      VZERO(INONL)=TR1

```

```

      IF(VZERO(INONL).GT.VNONL(INONL))VZERO(INONL)=0.0
      IF(IPR.GT.0)
1WRITE(*,700)TRR,TXX,CIT,TRI
700  FORMAT(2X,'TYPE 99 NONLINEAR R',2X,3E11.3,15)
      GOTO 100
7342 IF(ITYPE.NE.98)GOTO 7346
      NLTYPE(INONL)=-98
      NUM=NUM+1
      CURR(INONL)=1.0
      IF(IPR.GT.0)
1WRITE(LO,720)NL1,NL2
720  FORMAT(' TYPE 98 PSEUDO NONLINEAR L',2E11.3)
      GOTO 730
7346 IF(ITYPE.EQ.93.AND.IPR.GT.0)
1WRITE(LO,740)TRR,TXX
740  FORMAT(' NONLINEAR INDUCTANCE TYPE 93',2X,2E11.3)
      IF(ITYPE.EQ.92.AND.IPR.GT.0)
1WRITE(LO,760)TRR,TXX
760  FORMAT(' NONLINEAR RESISTANCE 92',2X,2E11.3)
C
C READ NEXT PART INPUT DATA
C =====
C
730  CONTINUE
      ICHAR=0
800  CONTINUE
C
C READ VALUE PAIRS DEPEND ON TYPE OF NONLINEAR
C
      READ(LI,*)D2,XL
      IF(D2.EQ.9999)GOTO 850
      IF(IPR.GT.0)
1WRITE(LO,810)D2,XL
810  FORMAT(' BREAKPOINT',2E15.5)
      IF(D2.GT.CII)GOTO 950
      GOTO 920
950  IF(XL.GT.CKK)GOTO 970
      GOTO 920
970  ICHAR=ICAR+1
      D3=D2-CII
      D4=XL-CCK
      IF(ITYPE.NE.99)GOTO 900
      GSLOPE(ICAR)=D3/D4
      CCHAR(ICAR)=CII-GSLOPE(ICAR)*CCK
      GOTO 1000
900  IF(ITYPE.NE.98)GOTO 920
      D1=D4/D3
      GSLOPE(ICAR)=DELTA2/D1
      CCHAR(ICAR)=CCK-D1*CII
      GOTO 1000
920  CCHAR(ICAR)=D2
1000  CII=D2
      CCK=XL
      IF(ITYPE.EQ.92)XL=XL+D2*TX(IT)
      VCHAR(ICAR)=XL

```

```
      GOTO 800
850  IF (ITYPE.NE.93.AND.ITYPE.NE.98)GOTO 100
      NONLE(INONL)=-ICHAR
      VNONL(INONL)=0.0
      GOTO 100
270  CONTINUE
9200 WRITE(*,32675)
32675 FORMAT(' ERROR OF NONLINEAR PART')
100  RETURN
      END
```

```

SUBROUTINE CONECT
DIMENSION BUFF(20)
CHARACTER*6 BUFF,TERRA,BUSS,BUS,BLANK

C
C -----
C SUBROUTINE TO PREPARE TOPOLOGY OF NETWORK
C FROM INPUT DATA
C -----
C
$INCLUDE:'COMMON.FOR'
KPARTA=0
NUM=0
WRITE(*,502)NTOT,IBR,IT
502 FORMAT(' NTOT-IBR-IT',3I6)
TERRA='TERRA'
BLANK='
LAST=200
OVR='CONECT'
NTOT1=NTOT-1
N3=0
D1=1.0/(100.*1.E-10)
I=INONL
54147 IF(1.EQ.0)GOTO 4803
C
C CHECK NONLINEAR PARAMETER
C
IF(NLTYPE(I).GT.0)GOTO 54174
K=NONLK(I)
M=IABS(NONLM(I))
DO 54155 J=1,IBR
N1=IABS(KBUS(J))
N2=IABS(MBUS(J))
IF(N1.NE.K)GOTO 54154
IF(N2.EQ.M)GOTO 54174
GOTO 54155
54154 IF(N2.NE.K)GOTO 54155
IF(N1.EQ.M)GOTO 54174
54155 CONTINUE
C --- EXIT ---
IBR=IBR+1
IF(IBR.LE.LBRNCH)GOTO 4716
GOTO 9200
4716 KBUS(IBR)=K
MBUS(IBR)=M
IF(N3.GT.0)GOTO 54162
J1=K
J2=M
IT=IT+1
TR(IT)=D1
TX(IT)=0.0
C(IT)=0.0
54162 NR(IBR)=-IT
LENGTH(IBR)=1
IF(N3.GT.0)LENGTH(IBR)=-1
54174 I=I-1

```

```

IF(I.GT.0)GOTO 54147
GOTO 4803
C 4774 J=J+NI
C IF(J.LE.IBR)GOTO 4748
C N3=IBR+1
C IBR=IBR+3
C IF(IBR.LE.LBRNCH)GOTO 4782
C GOTO 9200
4803 CALL MOVE0(KOLUM,IBR)
CALL MOVE0(LOC,NTOT)
NEXT=1
I=1
5010 IF(I.GT.IBR)GOTO 5040
K=IABS(KBUS(I))
M=IABS(MBUS(I))
C IF(BUS(K).EQ.BLANK)GOTO 5020
C IF(BUS(M).EQ.BLANK)GOTO 5020
ASSIGN 5020 TO MOON
GOTO 5200
5020 I=I+1
GOTO 5010
5040 I=1
5045 IF(I.GT.INONL)GOTO 5060
K=IABS(NONLK(I))
M=IABS(NONLM(I))
ASSIGN 5050 TO MOON
GOTO 5200
5050 I=I+1
GOTO 5045
5060 I=1
5065 IF(I.GT.KSWTCH)GOTO 5250
K=IABS(KHIGH(I))
M=IABS(KLOW(I))
5080 CONTINUE
ASSIGN 5085 TO MOON
GOTO 5200
5085 I=I+1
GOTO 5065
5200 IF(K.GT.NTOT.OR.M.GT.NTOT)GOTO 5240
IPASS=0
5204 LLEFT=LOC(K)
IF(LLEFT.GT.0)GOTO 5214
LTEST=0
5208 LOC(K)=NEXT
GOTO 5235
5214 ISUBS1=IOFKOL+LLEFT
JLEFT=KOLUM(ISUBS1)
IF(M.GT.JLEFT)GOTO 5222
LTEST=LLEFT
GOTO 5208
5222 ISUBS1=IOFKOR+LLEFT
LTEST=KORDER(ISUBS1)
IF(LTEST.EQ.0)GOTO 5228
ISUBS1=IOFKOL+LTEST
JTEST=KOLUM(ISUBS1)

```

```

IF(JTEST.GE.M)GOTO 5228
LLEFT=LTEST
GOTO 5222
5228 ISUBS1=IOFKOR+LLEFT
KORDER(ISUBS1)=NEXT
5235 ISUBS1=IOFKOL+NEXT
KOLUM(ISUBS1)=M
ISUBS1=IOFKOR+NEXT
KORDER(ISUBS1)=LTEST
IF(NEXT.LT.LAST)GOTO 5237
GOTO 9200
5237 NEXT =NEXT+1
IF(IPASS.GT.0)GOTO 5240
IPASS=K
K=M
M=IPASS
GOTO 5204
5240 GOTO MODN, (5020,5050,5085)
5250 CONTINUE
WRITE(LO,5254)
5254 FORMAT(/, 'LIST OF INPUT ELEMENTS CONNECTED TO EACH BUS.'
1,/,10X,'1) ONLY THE PHYSICAL CONNECTIONS OF MULTIPHASE LINES ARE
2 SHOWN (CAPACITIVE AND INDUCTIVE COUPLING IGNORED)'
3,/,10X,'2) REPEATED ENTRIES IMPLY PARALLEL CONNECTIONS',/
4,10X,'3) SOURCES ARE OMITTED,ALTHOUGH SWITCHES ARE INCLUDED',/)
WRITE(LO,5261)
5261 FORMAT(1X,'FROM BUS NAME 1 NAMES OF ALL ADJACENT BUSES')
WRITE(LO,5266)
5266 FORMAT(1X,122('-',))
WRITE(LO,5267)
5267 FORMAT(1H+,14X,1H1)
IPASS=0
K=2
5268 IF(LOC(K).EQ.0)GOTO 5294
BUS5=BUS(K)
5270 LTEST=LOC(K)
IF(LTEST.EQ.0)GOTO 5294
5272 ISUBS1=IOFKOL+LTEST
M=KOLUM(ISUBS1)
IPASS=IPASS+1
BUFF(IPASS)=BUS(M)
IF(M.EQ.1)BUFF(IPASS)=TERRA
IF(IPASS.LT.15)GOTO 5285
5277 WRITE(LO,5278)BUS5,(BUFF(M),M=1,IPASS)
5278 FORMAT(9X,A6,1H1,15(A6,1H*))
BUS5=BLANK
IPASS=0
IF(LTEST.EQ.0)GOTO 5294
5285 ISUBS1=IOFKOR+LTEST
LTEST=KORDER(ISUBS1)
IF(LTEST.GT.0)GOTO 5272
GOTO 5277
5294 K=K+1
IF(K.EQ.2)GOTO 5320
IF(K.LE.NTOT)GOTO 5268

```

```
BUS5=TERRA
K=1
GOTO 5270
5320 WRITE(LO,5266)
      WRITE(LO,5267)
      WRITE(LO,7659)(BUS(I),I=1,NTOT)
7659 FORMAT(/,' BUS NAME',/,(1X,10A12))
C
C ----- EXIT FROM CONNECTIVITY -----
C
C FOR DEBUG ONLY
C
C   WRITE(LO,8880)(I,KBUS(I),MBUS(I),NR(I),LENGTH(I),KOLUM(I),
C   1KORDER(I),LOC(I),KOWNT(I),BUS(I),I=1,30)
C8880 FORMAT(1X,'I KBUS MBUS NR LG KOL KOR LOC KWN BUS'
C   1,/,1X,9I5,A10)
      GOTO 99999
9200 CONTINUE
      WRITE(LO,9201)
9201 FORMAT(' ERROR RETURN ')
99999 RETURN
      END
```


SUBROUTINE PREPTB

```

C
C   SUBROUTINE PREPARE TABLE FOR EACH TYPE
C   OF INPUT DATA :-
C       1. LUMPED PARAMETER TABLE
C       2. DISTRIBUTED DATA TABLE
C       3. NONLINEAR PARAMETER TABLE
C       4. SWITCH TABLE
C
C $INCLUDE: 'COMMON.FOR'
C
C ----- PREPARING R,X,C TABLES -----
C
598   D2=1.0/(DELTAT*500.)
      C11=D2/1000.
      IF(XOPT.GT.0.0)D2=D2*1000./(TWOPI*XOPT)
      IF(COPT.GT.0.0)C11=C11/(TWOPI*COPT)
901   CONTINUE
      IF(IPR.GT.0)
1WRITE(LO,65435)(TR(I),TX(I),R(I),C(I),I=1,IT)
65435 . FORMAT(/, ' LUMPED PARAMETER VALUE FROM LU4',/,18X,'TR',
118X,'TX',19X,'R',19X,'C',/, (1X,4E20.10))
      K=1
5527   J=LENGTH(K)
      IF(J.EQ.0)J=1
      J=IABS(J)
      IF(KBUS(K).LT.0)GOTO 5539
      IF(LENGTH(K).LT.0)GOTO 5539
      N1=IABS(NR(K))
      N2=N1-1+J*(J+1)/2
      DO 5534 I=N1,N2
          X(I)=TR(I)+TX(I)*D2
5534   TX(I)=-X(I)
5539   K=K+J
      IF(K.LE.IBR)GOTO 5527
      IF(IPR.GT.0)
1WRITE(*,5543)(X(I),I=1,IT)
5543   FORMAT(/, ' X(I),I=1-IT AFTER CONVS R+2*L/DELTAT',/, (1X,10E13.5))
      IF(IPR.GT.0)
1WRITE(*,2712)(KCTRL(K),K=1,NTOT)
2712   FORMAT(/, ' KCTRL (1,NTOT)',1X,10I12)
      DO 410 I=1,IT
410   C(I)=C(I)*C11
C
C   SWITCH SET PART
C
433   IF(KSWTCH.EQ.0)GOTO 2841
C
C -- INITIALIZE KCTRL VECTOR FOR OPEN SWITCHES
C
      DO 422 K=1,KSWTCH
          N1= KHIGH(K)
          N2=KLOW(K)
          N3=KCTRL(N1)
          IF(N3.LT.0)GOTO 423

```

```

IF(N3.NE.0)GOTO 9200
KCTRL(N1)=N1
421 N3=KCTRL(N2)
IF(N3.LT.0)GOTO 423
IF(N3.NE.0)GOTO 9200
KCTRL(N2)=N2
GOTO 425
423 KHIGH(K) =-N1
GOTO 421
424 KLOW(K)=-N2
425 ICHECK=0
422 CONTINUE
IF(IPR.GT.0)
1WRITE(*,2712)(KCTRL(K),K=1,NTOT)
C -----
2841 CONTINUE
C
C INVERSION OF MATRICES
C PRESETTING TABLES FOR LOSSLESS LINE
C
430 K=1
ITADD=IT+1
N1=IBR+1
IF(N1.LT.10)N1=10
IF(IPR.GT.0)
1WRITE(*,5306)(I,KBUS(I),MBUS(I),LENGTH(I),CIK(I),CK(I),CKKJM(I)
2,I=1,N1)
5306 FORMAT(/,' ROW KBUS MBUS LENGTH CIK CKI CKKJM',
I/, (4I8,3E15.6))
N5=K
ILINE=1
500 IF(KBUS(K).LT.0)GOTO 535
IT2=LENGTH(K)
IF(IT2.LT.0)GOTO 520
I=NR(K)
IF(IT2.LE.1)GOTO 510
CALL REDU(TX(I),IT2,0)
II=I-1
N2=1
501 DO 502 L=1,IT2
IF(L.LE.N2)GOTO 504
N1=N1+L-1
GOTO 502
504 N1=II+L
502 VOLT(L)=TX(N1)
CALL MULT(TR(I),VOLT,VOLTI,IT2,0)
CALL MULT(TX(I),VOLTI,VOLT,IT2,0)
DO 503 L=1,N2
II=II+1
503 R(II)=(TX(II)-VOLT(L))*2.0
N2=N2+1
IF(N2.LE.IT2)GOTO 501
GOTO 520
C -----
530 I=IABS(I)

```

```

GUS1=C(I)
IF(GUS1.NE.0.)GUS1=1.0/GUS1
GUS2=1.0/(X(I)+GUS1)
R(I)=GUS2*(TR(I)*2.0-X(I)+GUS1)
X(I)=GUS2
C(I)=GUS1
GOTO 520
535 IT2=IABS(LENGTH(K))
N1=K
N3=1
5350 IF(CK1(N1).LT.0.0)GOTO 5353
N1=N1+1
N3=N3+1
GOTO 5350
5353 N2=K+N3-1
LENGTH(N2)=N3
IT2=0
N1=K+IT2-1
N2=ITADD
DO 5348 I=K,N1
NR(I)=N2
N2=N2+1
5348 CONTINUE
K=K+N3-IT2
GOTO 520
5349 A=IT2
II=1
534 IF(CK(K).GE.0.0)GOTO 6482
6482 H2=CI(K)
H3=H2/DELTAT
I=H3
IF(ISTEAD.GT.0)GOTO 531
IF(H2.LE.TMAX)GOTO 531
I=-2
531 N4=H3
D1=N4
CI(K)=H3-D1
IF(II.EQ.1)LENGTH(K)=ILINE
ILINE=ILINE+I+2
NR(K)=I
IF(CK(K).GT.0.0)GOTO 533
IF(CK(K).GE.0.0)GOTO 5074
CK(K)=-1.0
GOTO 533
5074 CK1=CK(K)+0.5
YX=CI(K)
IF(YX.LT.0.)GOTO 532
YX=YX+CK1/2.0
CK(K)=(YX-CK1)/YX
IF(CK(K).GT.0.0)GOTO 533
GOTO 9200
532 CK(K)=EXP(CK1/YX)
533 CI(K)=1.0/(YX+A)
IF(IPR.GT.0)
1WRITE(LO,5334)K,II,LENGTH(K),ILINE,IT2,I,N3,H3,CI(K)

```



```
5334  FORMAT(/, ' AFTER PROCESS NEW MODE', /, 718, 2E15.6)
5335  IF(I1.LT.IT2)GOTO 5375
      IT2=1
      GOTO 520
5375  IF(I1.GT.1)GOTO 538
      LENGTH(K+1)=LENGTH(K)
      LENGTH(K)=-IT2
538   I1=I1+1
      K=K+1
      GOTO 534
510   IF(I.LT.0)GOTO 530
      TX(I)=1.0/X(I)
      R(I)=2.0*(X(I)-TR(I))/X(I)**2
520   IF(KBUS(K).GE.0)GOTO 81520
81520  K=K+IABS(IT2)
      N5=K
      IF(K.LE.IBR)GOTO 500
      IF(IPR.GT.0)
        IWRITE(*,208)(K,KBUS(K),MBUS(K),NR(K),LENGTH(K),K=1,IBR)
208   FORMAT(/, 'BRANCH TABLE VECTOR', /, ' ROW  KBUS  MBUS
      INR  LENGTH', /, (518))
      GOTO 9999
9200  CONTINUE
      WRITE(LO,9922)
9922  FORMAT(' ERROR AT SUBROUTINE PREPTB')
9999  CONTINUE
      RETURN
      END
```

SUBROUTINE FORMY

```

C
$INCLUDE: 'COMMON.FOR'
C
C   BEGINNING OF FORMATION OF THE REAL
C   ADMITTANCE MATRIX (Y)
C
      DIMENSION KENT99(100)
      L=1
      II=0
      KSS=LYMAT+1
      IALTER=0
      CALL MOVERO(FINIT,NTOT)
      I=INONL
      GOTO 73577
73571  IF(NLTYPE(I).GT.0)GOTO 73574
      CURR(I)=0.0
      IF(ANONL(I).GE.0.0)GOTO 74572
      CURR(I)=1.0
      ANONL(I)=-1.0/1.E+20
74572  K=NONLK(I)
      N1=NONLM(I)
      M=IABS(N1)
      IF(KCTRL(K).EQ.0)KCTRL(K)=K
      IF(KCTRL(M).EQ.0)KCTRL(M)=M
      N15=K
      IF(N15.EQ.1)N15=M
      N16=IABS(KSSFRO(N15))
      OMEGA =TWOPI*SFREQ(N16)
      D1=(F(K)-F(M))/OMEGA
      D2=E(K)-E(M)
73584  IF(NLTYPE(I).NE.-98)GOTO 73574
      NONLE(I)=-NONLE(I)
      JO=KENT99(I)
      ANONL(I)=6SLOPE(JO)*(D1/DELTA2-D2)
73585  K=NONLK(I)
      M=IABS(NONLM(I))
      FINIT(M)=FINIT(M)+ANONL(I)
      FINIT(K)=FINIT(K)+ANONL(I)
      IF(IPR.GT.0)
        IWRITE(LO,594)(FINIT(J),J=1,NTOT)
594   FORMAT(1X,5E25.15)
73574  I=I-1
73577  IF(I.GT.0)GOTO 73571
      IF(INONL.GT.0.AND.IPR.GT.0)
        IWRITE(LO,73578)(I,NONLK(I),NONLM(I),NLTYPE(I),NONLAD(I),NONLE(I),
          2VNONL(I),CURR(I),ANONL(I),VZERO(I),I=1,INONL)
73578  FORMAT(/,'NONLINEAR ELEMENT TABLE BEFORE (Y) FORMATION',/,
          1(6I10,4E15.5))
      IF(KPARTA.GT.1)GOTO 600
      DO 597 I=2,KPARTB
      IF(KCTRL(I).EQ.0)KCTRL(I)=I
597   CONTINUE
C
C -----

```

C ESTABLISH L-TH Y ROW FROM BRANCH LIST

```

C -----
C
600  L=L+1
      KK(L)=L
      CALL MOVERO(F,NTOT)
      K=1
620  N1=KBUS(K)
      IT2=LENGTH(K)
      IF(N1.LT.0)GOTO 690
      IT2=IABS(IT2)
      IF(IT2.EQ.1)GOTO 650
      IT1=K
      DO 630 I=1,IT2
      IF(KBUS(IT1).EQ.L)GOTO 700
      IF(MBUS(IT1).EQ.L)GOTO 710
      GOTO 630
700  ICHECK=0
      GOTO 720
710  ICHECK=1
720  IOLD=IT2-1
      I1=I-1
      N1=NR(IT1)
      DO 750 JO=1,IT2
      J=JO-1
      N2=K+J
      IF(ICHECK.NE.0)GOTO 760
      N3=KBUS(N2)
      N4=MBUS(N2)
730  IF(J.LE.I1)GOTO 770
      N2=NR(N2)+I1
740  YX=TX(N2)
C -----
      F(N3)=F(N3)+YX+C(N2)/2.0
      F(N4)=F(N4)-YX
      GOTO 750
760  N3=MBUS(N2)
      N4=KBUS(N2)
      GOTO 730
770  N2=N1+J
      GOTO 740
750  CONTINUE
630  IT1=IT1+1
640  K=K+IT2
      GOTO 780
650  MABS=IABS(MBUS(K))
      IF(N1.EQ.L)GOTO 660
      IF(MABS.EQ.L)GOTO 670
      GOTO 640
660  N4=MABS
      GOTO 680
670  N4=N1
680  N3=NR(K)
      N2=IABS(N3)
      YX=X(N2)

```

```

      IF(N3.GT.0)YX=1.0/YX
      F(L)=F(L)+YX
      F(N4)=F(N4)-YX
      IF(N3.LT.0)GOTO 640
      F(L)=F(L)+C(N2)/2.0
      GOTO 640
C ----- DISTRIBUTED PARAMETER DATA -----
C
690   YX=ABS(CI(K))
      IT2=-IT2
      IF(IT2.LT.0)IT2=1
      N2=K+IT2-1
      DO 21690 I=K,N2
      IF(CK(I).GE.0.0)GOTO 21690
      J=CI(K)
      CI(I)=CI(I)/ETA(J)
21690 CONTINUE
      IT1=K
      JO=K
      DO 693 I=1,IT2
      IF((-KBUS(IT1)).EQ.L)GOTO 694
      IF(MBUS(IT1)).EQ.L)GOTO 695
      GOTO 693
694   JO=0
695   CONTINUE
      N2=K
      DO 698 N4=1,IT2
      IF(JO.EQ.0)GOTO 696
      N1=MBUS(N2)
      GOTO 697
696   N1=-KBUS(N2)
697   YX=0.0
      KJ=K
C     IJ=LITYPE(K)-1
      N4J=IJ+N4
      IJ=IJ+1
      DO 1696 J=1,IT2
      YX=YX+QFD(IJ)*ABS(CI(KJ))*QFD(N4J)
      IF(IPR.GT.0)
2696  IWRITE(LO,2696)J,IT2,KJ,CI(KJ),N4J,QFD(N4J),IJ,QFD(IJ),YX
      FORMAT(' J,IT2,KJ,N4J, QFD(N4J) IJ QFD(IJ)',/,20X,318,
1E15.6,18,E15.6,18,2E15.6)
      IJ=IJ+IT2
      N4J=N4J+IT2
1696  KJ=KJ+1
      YX=YX*FLOAT(IT2)
      F(N1)=F(N1)+YX
698   N2=N2+1
      IF(IPR.GT.0)
      IWRITE(LO,1697)CI1,L,F(L)
1697  FORMAT(' CI1 L AND F(L)',5X,E15.6,18,E15.6)
693   IT1=IT1+1
C ----- TO RECOVER THE ARRAY CI
      N2=K+IT2-1
      DO 31690 I=K,N2

```

```

IF(CK(I).GE.0.0)GOTO 31690
J=CIK(I)
CI(I)=CI(I)*ETA(J)
31690 CONTINUE
GOTO 640
6920 IT2=IABS(KODEBR(K))
NI=CIK(K)
DO 13200 I=1,IT2
IF(NI.GT.0)CALL MOVERO(VOLT,IT2)
N9=K+I-1
IF(-KBUS(N9).NE.L)GOTO 13010
N3=1
D1=1.0
GOTO 13020
13010 IF(IABS(MBUS(N9)).NE.L)GOTO 13200
N3=-1
D1=1.0
13020 KBR=K
13030 N4=ABS(CKI(KBR))-1.0
N5=N4/IT2+1
N4=N4-(N5-1)*IT2+1
IF(N1.LT.0.AND.N4.NE.1)GOTO 13140
IF(N1.LT.0)GOTO 13120
C
C -----VOLT() IS THE I TH ROW OF QFD*YMODAL
C
N6=(N4-1)*IT2+I+N1-1
VOLT(N5)=VOLT(N5)+QFD(N2)*D1*D2
GOTO 13140
13120 N6=K+N5-1
C
C ADD(SUBTRACT) ITH ROW OF Y PHASE TO FROM L TH ROW
C OF Y TOTAL
IF(N3.GT.0)GOTO 13130
N6=IABS(MBUS(N6))
GOTO 13140
13130 N6=-KBUS(N6)
E(N6)=E(N6)+D1*D2
13140 KBR=KBR+1
IF(CKI(KBR-1).GT.0.0)GOTO 13030
IF(N1.LT.0)GOTO 13190
N4=K+IT2-1
DO 13180 KBR=K,N4
IF(N3.GT.0)GOTO 13150
N5=IABS(MBUS(KBR))
GOTO 13160
13150 N5=-KBUS(KBR)
13160 N6=N1+KBR-K
D1=0.0
DO 13170 J=1,IT2
D1=D1+VOLT(J)*QFD(N6)
N6=N6+IT2
13170 CONTINUE
F(N5)=F(N5)+D1
13180 CONTINUE

```



```

IF(CK(I).GE.0.0)GOTO 31690
J=CIK(I)
CI(I)=CI(I)*ETA(J)
31690 CONTINUE
      GOTO 640
6920  IT2=IABS(KODEBR(K))
      N1=CIK(K)
      DO 13200 I=1,IT2
      IF(N1.GT.0)CALL MOVERO(VOLT,IT2)
      N9=K+I-1
      IF(-KBUS(N9).NE.L)GOTO 13010
      N3=1
      D1=1.0
      GOTO 13020
13010 IF(IABS(MBUS(N9)).NE.L)GOTO 13200
      N3=-1
      D1=1.0
13020 KBR=K
13030 N4=ABS(CKI(KBR))-1.0
      N5=N4/IT2+1
      N4=N4-(N5-1)*IT2+1
      IF(N1.LT.0.AND.N4.NE.1)GOTO 13140
      IF(N1.LT.0)GOTO 13120

C
C -----VOLT() IS THE I TH ROW OF QFD*YMODAL
C
      N6=(N4-1)*IT2+I+N1-1
      VOLT(N5)=VOLT(N5)+QFD(N2)*D1*D2
      GOTO 13140
13120 N6=K+N5-1
C
C ADD(SUBTRACT) ITH ROW OF Y PHASE TO FROM L TH ROW
C           OF Y TOTAL
      IF(N3.GT.0)GOTO 13130
      N6=IABS(MBUS(N6))
      GOTO 13140
13130 N6=-KBUS(N6)
      E(N6)=E(N6)+D1*D2
13140 KBR=KBR+1
      IF(CKI(KBR-1).GT.0.0)GOTO 13030
      IF(N1.LT.0)GOTO 13190
      N4=K+IT2-1
      DO 13180 KBR=K,N4
      IF(N3.GT.0)GOTO 13150
      N5=IABS(MBUS(KBR))
      GOTO 13160
13150 N5=-KBUS(KBR)
13160 N6=N1+KBR-K
      D1=0.0
      DO 13170 J=1,IT2
      D1=D1+VOLT(J)*QFD(N6)
      N6=N6+IT2
13170 CONTINUE
      F(N5)=F(N5)+D1
13180 CONTINUE

```

```

13190 IF(N3.GT.0)GOTO 13010
13200 CONTINUE
      IT2=1
13210 IF(CKI(K).LT.0.0)GOTO 640
      K=K+1
      GOTO 13210
780   IF(NUM.LE.0)GOTO 2801
      DO 2793 I=1,INONL
      IF(NLTYPE(I).GT.0)GOTO 2793
      IF(NLTYPE(I).EQ.-98)GOTO 2781
      IF(ANONL(I).GE.0.0)GOTO 2793
2781  K=NONLK(I)
      M=IABS(NONLM(I))
      IF(K.NE.L)GOTO 2784
      N4=M
      GOTO 2787
2784  IF(M.NE.L)GOTO 2793
      N4=K
2787  N3=KENT99(I)
      YX=6SLOPE(N3)
      F(L)=F(L)+YX
      F(N4)=F(N4)-YX
      IF(IPR.GT.0)
        1WRITE(LO,2789)I,K,M,L,N4,YX,F(L),F(N4)
2789  FORMAT(/,' TYPE 98',5I10,/,3E25.15)
2793  CONTINUE
2801  IF(L.LE.KPARTA)GOTO 782
C
C NODE L IS IN PARTITION B OF Y FOR WHICH THE FULL ROW OF Y
C TEMP STORED YKM,KM
C
      KKS(L)=KSS
      IF(F(L).NE.0.0)GOTO 4308
      GOTO 9200
4308  DO 2010 I=2,NTOT
      IF(F(I).EQ.0.1)GOTO 2010
      KSS=KSS-1
      KM(KSS)=I
      YKM(KSS)=F(I)
2010  CONTINUE
      IF(KSS.GT.11)GOTO 1408
      GOTO 9200
1408  KM(KSS)=-KM(KSS)
      GOTO 840
782  CONTINUE
C
C ELIMINATION PROCESS ON L TH ROW
C
      KLIMIT =L-1
790  K=1
      IF(IPR.GT.0)
        1WRITE(LO,7013)L,KPARTA,IBR,IT,(F(I),I=1,NTOT)
7013  FORMAT(' ROW OF [Y] - L-KPARTA-IBR-IT > F(I)',4I10,/,1X,5E25.15)
C -----
800  K=K+1

```

```

      IF(K.GT.KLIMIT)GOTO 815
      A=F(K)
      IF(A.EQ.0)GOTO 800
      I=IABS(KK(K))
810   NI=KM(I)
      IF(NI.LT.0)GOTO 800
      F(NI)=F(NI)-A*YKM(I)
      I=I-1
      GOTO 810
815   IF(IALTER.EQ.1)GOTO 880
C
C STORE OF ROW L OF REDUCED [Y]
C
      A=F(L)
      A=1.0/A
      II=II+1
      YKM(II)=A
      KM(II)=-L
      IF(L.GE.NTOT)GOTO 834
      N4=L+1
      DO 830 K=N4,NTOT
      IF(F(K).EQ.0.)GOTO 830
      II=II+1
      YKM(II)=F(K)*A
      KM(II)=K
830   CONTINUE
834   KK(L)=II
      IF(II.LT.KSS)GOTO 840
      GOTO 9200
840   IF(L.LT.KPARTB)GOTO 600
      I=KPARTB
843   I=I+1
      IF(I.GT.NTOT)GOTO 844
      KK(I)=I
      GOTO 843
844   IUPPER=II
      IEND=II
C -----
C BEGIN ELIMINATION OF PARTITION 'A' VARIABLES
C FROM PARTITION 'B' EQUATION YBB/YBC IN KMS,YS
C -----
      KSS=0
      IF(KPARTB.LE.KPARTA)GOTO 900
      IALTER=1
      KLIMIT =KPARTA
      L=KPARTA
849   L=L+1
      CALL MOVERO(F,NTOT)
      J=KKS(L)
860   J=J-1
      I=KM(J)
      IF(I.LT.0)GOTO 870
      F(I)=YKM(J)
      GOTO 860
870   I=IABS(I)

```

```

      F(I)=YKM(J)
      GOTO 790
880   KKS(L)=KSS
      IF(NUM.LE.0)GOTO 73620
      DO 73610 I=1,INONL
      IF(NLTYPE(I).GT.0)GOTO 73610
      K=NONLK(I)
      M=IABS(NONLM(I))
      IF(K.NE.L)GOTO 73600
      IF(F(M).EQ.0.0)F(M)=1.0/1.0E+20
      GOTO 73610
73600 IF(M.NE.L)GOTO 73610
      IF(F(K).EQ.0.0)F(K)=1.0/1.0E+20
73610 CONTINUE
73620 CONTINUE
      N3=KPARTA+1
      DO 885 I=N3,NTOT
      IF(F(I).EQ.0.0)GOTO 885
      KSS=KSS+1
      KMS(KSS)=I
      YS(KSS)=F(I)
885   CONTINUE
      KMS(KSS)=-KMS(KSS)
      IF(KSS.LE.LSMAT)GOTO 894
      GOTO 9200
900   CONTINUE
894   IF(L.LT.KPARTB)GOTO 849
      IF(IPR.GT.0)
      IWRITE(LO,903)KPARTA,KPARTB,NTOT,IUPPER,IEND,(KK(I),I=1,NTOT)
903   FORMAT(//,'KPARTA  KPARTB  NTOT  IUPPER  IEND',S110,/,
      1(1X,10I11))
      N3=KPARTB
      IF(KSS.GT.N3)N3=KSS
      GOTO 9999
9200  CONTINUE
9999  RETURN
      END

```

```

SUBROUTINE SETUP
C
C   SET INITIAL BEFORE TIME STEP LOOP
C   -----
C
C   $INCLUDE: 'COMMON.FOR'
      DIMENSION ALOWER(13)
      CHARACTER*6 ALOWER,AUP,BUS1,BUS,BLANK
C
C   LPEAK : EACH PARTITION OF Xmax,Tmax etc.= 20 arrays
C
      LPEAK=20
      NOUT=20
      NUMV=0
      READ(LI,*) (AUP(I),I=1,13)
3047  L=0
      DO 3063 K=1,13
      BUS1=AUP(K)
      IF (BUS1.EQ.BLANK)GOTO 3063
      L=L+1
      DO 3052 I=2,NTOT
      IF (BUS1.EQ.BUS(I))GOTO 3056
3052  CONTINUE
      GOTO 3063
3056  NUMV=NUMV+1
      IBSOUT(NUMV)=I
      IF (NUMV.LE.NOUT)GOTO 3063
      KILL=1
      GOTO 9200
3063  CONTINUE
      IF (L.EQ.0)GOTO 3081
      IF (IPR.GT.0)
1WRITE (LO,3072)
3072  FORMAT(' BUS NAMES FOR NODE VOLTAGE OUTPUT ')
      WRITE (LO,3176) (AUP(I),I=1,13)
3176  FORMAT(1X,7A10)
3081  IF (IPR.LE.0)GOTO 3094
      WRITE (LO,53166) IUPPER
53166  FORMAT(' END NODE NAMES FOR OUTPUT VOLTAGE',/, ' UPPER TRIANGLE FA
      ICTORS ', 'FOR NONSWITCH/SOURCE ROWS OF COEFF. MATRIX NO.',IS)
3094  IF (KTAB.GT.0)GOTO 4752
      IOUTCS=0
      KLNOUT=0
4752  N5=NC+IOUTCS+NUMOUT
      N6=1
      N16=1
      IF (NUMBUS.GT.0)N16=NUMBUS
      N9=IOUTCS+1
      N3=NTOT+N6
      N8=N3+N9
      N18=N8+N16
      IF (N5.LE.NOUT)GOTO 3201
      KILL=2
      GOTO 9200
3201  NDX1=KLNOUT+N9

```

```

3204 NCSAVE=NC
3209 I=0
3211 I=I+1
      IF(I.GT.10UTCS)GOTO 3218
      NC=NC+1
      IBRNCH(NC)=N8
      JBRNCH(NC)=N3+I
      GOTO 3211
3218 CONTINUE
3098 CONTINUE
C
C INITIALIZE XMAX,XMIN-TMAX,TMIN
C
1063 DO 1064 I=1,K
      XMAX(I)=-1.0E+20
      NDX1=LPEAK+I
      XMAX(NDX1)=0.0
      NDX1=LPEAK+NDX1
      XMAX(NDX1)=1.0E+20
      NDX1=LPEAK+NDX1
1064 XMAX(NDX1)=0.0
C
C -----
C
54242 K=0
      NI=0
      KPRSTA=1
      GOTO 3152
3119 K=K+1
      J=K+1
      J=IBSOUT(K)
      M=M+1
      AUP(M)=BUS(J)
      ALOWER(M)=BLANK
      IF(M.LT.9)GOTO 3158
      ASSIGN 3152 TO MOON
3126 IF(IPR.LT.0)GOTO 7113
C
C PRINT OUTPUT VOLTAGE SELECTION HEADING
C -----
      WRITE(LO,3131)(AUP(I),I=1,9)
3131 FORMAT(/,15X,9(7X,A6))
      IF(N1.GT.0)GOTO 3144
      N1=1
      WRITE(LO,3137)
3137 FORMAT(1H+, ' STEP TIME' )
3144 WRITE(LO,3145)(ALOWER(I),I=1,9)
3145 FORMAT(15X,9(7X,A6))
      KPRSTA=1
7113 GOTO MOON, (3152,3166,3177)
3152 M=0
3158 IF(K.LT.NUMV)GOTO 3119
      K=0
      GOTO 3169
C

```

```
C CURRENT SELECTION BRANCH
C
3164 K=K+1
      L=IBRNCH(K)
      J=JBRNCH(K)
      M=M+1
      AUP(M)=BUS(L)
      IF(L.EQ.1)AUP(M)=TERRA
      ALOWER(M)=BUS(J)
      IF(J.EQ.1)ALOWER(M)=TERRA
      IF(M.LT.9)GOTO 3169
      ASSIGN 3166 TO MOON
      GOTO 3126
3166 M=0
3169 IF(K.LT.NCSAVE)GOTO 3164
3198 IF(M.EQ.0)GOTO 3177
3171 M=M+1
      AUP(M)=BLANK
      ALOWER(M)=BLANK
      IF(M.LT.9)GOTO 3171
      ASSIGN 3177 TO MOON
      GOTO 3126
3177 NC=NCSAVE
9200 CONTINUE
9999 RETURN
      END
```

SUBROUTINE TSTEP

```
C
C TIME STEP LOOP CALCULATION PROCESS
C CALL TS1,TS2,TS3
C ( MAIN CALCULATION SUBROUTINE FOR ELECTROMAGNETIC TRANSIENT ANALYSIS )
C
$INCLUDE: 'COMMON.FOR'
      NCH=0
2400  IF(NCH.EQ.9)GOTO 9999
      IF(NCH.NE.0)GOTO 9200
301   CALL TS1
      IF(NCH.NE.0)GOTO 9200
302   CALL TS2
      IF(NCH.NE.0)GOTO 9200
303   CALL TS3
      GOTO 2400
9200  WRITE(LO,6665)NCH
6665  FORMAT(' < ERROR TSO > NCH=',I6)
      GOTO 99999
9999  WRITE(LO,6666)TMAX
6666  FORMAT(' FINISH THE CALCULATION PROCESS AT TIME TMAX =',E15.6)
99999  RETURN
      END
```



```

SUBROUTINE TSI
C
C TSI CALLED BY SUBROUTINE TSTEP
C STEP 1&2 OF TIME STEP LOOP PROCESS
C
$INCLUDE: 'COMMON.FOR'
  DIMENSION KURL(30), ZNONL(30), KKNONL(30), VOLTBC(30)
  DIMENSION FSWABC(30), ISUBEG(30), ILAST(30), VF99(30)
  DIMENSION KFIN99(30)
  IF(IPR.GT.0)
    IWRITE(LO,548) ISTEP, KSWTCH, INONL, IUPPER
548  FORMAT(' BEGIN 1 ST PART OF TIME STEP LOOP ',/,
1' STEP KSWTCH INONL IUPPER ',/,10X,41B)
1000  KCOUNT =NV
C
C
      IF(KSWTCH.EQ.0)GOTO 1009
C
C
C CHECK SWITCH POSITION
C -----
      ISWENT =0
      IJUMPS=0
      N7=1
      DO 1003 K=1,KSWTCH
      ISWRRR=0
      IF(IABS(IJUMPS).GE.K)GOTO 55432
      ISWENT =ISWENT+1
      IJUMPS=JUMPIN(ISWENT)
55432 CONTINUE
      II=KPOS(K)
      IT1=FKPOS(K)
      ICHECK =KPOS2(K)
      I=IABS(II)
      VSL=0
      IF(I.EQ.5)GOTO 1002
      IF(I.EQ.0)GOTO 1002
      N3=KHIGH(K)
      N4=KLOW(K)
      N1=IABS(N3)
      N2=IABS(N4)
      TCL=TCLOSE(K)
      NN1=N1
      IF(IT1.GT.0)NN1=KBUS(IT1)
42103 IF(I.EQ.10)GOTO 2100
      IF(II.NE.2)GOTO 2097
      KPOS(K)=11
      IF(I.LE.0)KPOS(K)=-KPOS(K)
      I=11
      TCL=-1
2097 IF(I.EQ.11)GOTO 2108
      IF(IT1.LE.0)GOTO 2110
      GUS4=(E(NN1)-E(N2))*DELTA2
      C11=CK(IT1)
      CK1=C11+GUS4

```

```

CK(IT1)=CK1+GUS4
IF(ABS(CK1).GE.CRIT(K))GOTO 1002
IF(ABS(CK1).GT.ABS(CI1))GOTO 1002
GUS4=GUS4/DELTA2
IIT1=-NR(IT1)
CK1=X(IIT1)*GUS4
CIK(IT1+1)=CK1+CIK(IT1)+CIK(IT1+1)
CIK(IT1)=-CK1
GOTO 2113

```

```

C
C ---- PART OF CHECKING SWITCH POSITION CHANGE
C
2100 IF(T.LT.TCL)GOTO 1002
      IF(IT1.LE.0)GOTO 2101
      IIT1=-NR(IT)
      E(N1)=E(N2)
      GUS4=E(NN1)-E(N2)
      CIK(IT1)=-X(IT1)*GUS4
      GUS4=GUS4*DELTA2
      CK1=CK(IT1)
      A=CK1+GUS4
      CK(IT1)=A+GUS4
      IF(ABS(A).LT.TOPEN(K))GOTO 1002
      IF(ABS(A).LT.ABS(CK1))GOTO 1002
      D2=X(IIT1)/DELTA2
      C11=CIK(IT1+1)+D2*GUS4
      CK1=X(IIT1)/(DELTA2*D2)
      D3=A*(CK1+1.0)-C11/D2
      CRIT(K)=ABS(D3/CK1)
      GOTO 2102
2101 CK1=E(NN1)-E(N2)
      IF(ABS(CK1).LT.TOPEN(K1))GOTO 1002
      ADELAY(K)=T-FKPOS1(K)
      IF(K.NE.IABS(IJUMPS))GOTO 55450
      IF(IJUMPS.LT.0)GOTO 55436
      E(N2)=E(NN1)
      GOTO 55450
55436 E(NN1)=E(N2)
55450 GOTO 2102
2102 I=1
2105 TCLOSE(K)=0
      IF(TCL.GE.0)TCL=T
      IF(ISWRRR.EQ.0)WRITE(LO,21991)TCL
21991 FORMAT(101X,'CLOSED AFTER',E12.5,'SEC')
620 IF(N4.LT.0)GOTO 625
      IF(N3.LT.0)GOTO 623
      KCTRL(N2)=0
      GOTO 625
623 KCTRL(N2)=-N2
625 IF(KCTRL(N1).GT.0)KCTRL(N1)=N4
      GOTO 2115
2108 IF(T.LT.TCL)GOTO 1002
      I=2
      IF(TOPEN(K).EQ.1.0E+20)GOTO 2105
      IF(II.GT.0.AND.TOPEN(K).GT.TMAX)I=0

```

```

      GOTO 2105
2110  L=N2
      IF(N4.LT.0)L=N1
      A=-FSWTCH(K)
      IT2=KKS(L)
2111  IT2=IT2+1
      J=KMS(IT2)
      IF(J.LT.0)GOTO 2116
      A=A+YS(IT2)*E(J)
      GOTO 2111
2116  J=IABS(J)
      A=A+YS(IT2)*E(J)
      GUS1=A*TCLOSE(K)
      TCLOSE(K)=A
      IF(N4.LT.0)A=-1
      IF(II.LE.0)VSL=A
      IF(I.EQ.3)GUS1=-A
2112  IF(I.GT.1.AND.T.LT.TOPEN(K))GOTO 1002
      IF(ABS(A).LT.CRIT(K))GUS1=-1.0
C     IF(T.LT.ADELAY(K))GUS1=1.0
2117  IF(GUS1.GE.0)GOTO 1002
2113  KCTRL(N1)=N3
      KCTRL(N2)=N4
      I=I+1
      IF(I.NE.2)GOTO 2118
      I=10
2118  TCLOSE(K)=0.0
      IF(ISWRRR.EQ.0)WRITE(LO,2114)T
2114  FORMAT(101X,' OPEN AFTER',E12.5,'SEC')
2115  IALTER=I
      CONTINUE
      J=IABS(I)
      IF(II.LT.0)J=-J
      II=J
      KPOS(K)=II
1002  IF(II.GE.0)GOTO 1003
      KCOUNT=KCOUNT+1
      BVALUE(KCOUNT)=VSL
1003  CONTINUE
C
C
1009  IF(ISTEP.EQ.0)GOTO 800
C
C -----
800   CONTINUE
      IF(INONL.EQ.0)GOTO 73991
      DO 73990 I=1,INONL
      K=NONLK(I)
      M=IABS(NONLM(I))
      VSL=E(K)-E(M)
C -----IPR-----
      IF(NLTYPE(I).LT.0)GOTO 83903
      GUS2=CURR(I)
      IF(NONLE(I).GT.0)GOTO 73960
      IF(KUPL(I).NE.0)GOTO 73960

```

```

      VNONL(I)=VNONL(I)+DELTAT*VSL
      GOTO 73960
83903 K9899=1
      J=0
      GOTO 7642
7642  IT1=ABS(CURR(I))
      IT2=KENT99(I)-1
      BUS2=0.0
      IF(IT1.NE.0)GOTO 73910
73900 IF(ABS(VSL).LE.VF99(I))GOTO 73960
      J=1
      NN1=ANONL(I)
      IT2=IT2+NN1
      CURR(I)=NN1
      IF(VSL.LT.0.0)CURR(I)=-CURR(I)
      VECNL2(I)=T+VECNL1(I)
      A=GSLOPE(IT2)
      GUS1=CURINT(IT2)
73905 IF(VSL.LT.0.0)GUS1=-GUS1
      GOTO 73950
73910 IT2=IT2+IT1
      GUS2=VSL*GSLOPE(IT2)
      IF(NLTYPE(I).NE.-98)GOTO 43420
      K9899=0
      BUS2=BUS2+ANONL(I)
      GUS4=DELTA2*VSL
      VSL=VNONL(I)+GUS4
      GOTO 73908
43420 GUS2=ABS(GUS2)+CURINT(IT2)
      IF(VSL.LT.0.0)GUS2=-GUS2
      IF(T.LT.VECNL2(I))GOTO 73908
      IF(ABS(VSL).LT.VZERO(I))GOTO 73942
73908 A=CURR(I)*VSL
      IF(A.GT.0.0)GOTO 73913
      IF(IT1.EQ.1)GOTO 73937
73913 IF(IT2.GE.KFIN99(I))GOTO 73930
      IF(ABS(VSL).LE.VBREAK(IT2))GOTO 73930
      NN1=IT2-1
      IF(CURR(I).LT.0.0)GOTO 73935
73915 CURR(I)=CURR(I)+1.0
73925 A=GSLOPE(NN1)-GSLOPE(IT2)
      IF(K9899.GT.0)GOTO 73927
      GUS3=CURINT(NN1)
      IF(VSL.LT.0.0)GUS3=-GUS3
      D1=(VSL+GUS4-GUS3)*GSLOPE(NN1)/DELTA2
      GUS2=(ANONL(I)+D1)/2.0
      GOTO 73950
73927 GUS1=CURINT(NN1)-CURINT(IT2)
      GOTO 73905
73930 IF(IT1.GT.1)GOTO 73932
73931 IF(K9899.GT.0)GOTO 73960
      GOTO 73973
73932 NN1=IT2-1
      IF(ABS(VSL).GE.VBREAK(NN1))GOTO 73931
      IF(CURR(I).LT.0.0)GOTO 73915

```

```

73935  CURR(I)=CURR(I)-1.0
      GOTO 73925
73937  IF(K9899.GT.0)GOTO 73940
      CURR(I)=-CURR(I)
      GOTO 73973
73940  IF(T.GE.VECNL2(I))GOTO 73942
      CURR(I)=-CURR(I)
      GOTO 73978
73942  GUS1=-CURINT(IT2)
      IF(CURR(I);LT.0.0)GUS1=-GUS1
      J=1
      A=-GSLOPE(IT2)
      CURR(I)=0.0
      D1=VBREAK(IT2)-SGLFIR
      IF(ABS(D1).LT.1.E-12)V99(I)=1.0E+20
73950  IF(K.LT.M)GOTO 4113
      N8=M
      N9=N
      GOTO 4114
4113   N8=K
      N9=M
4114   IF(N8.EQ.1)GOTO 73965
      IF(N8.GT.KPARTB)GOTO 73978
      N3=KKS(N8)
73963  N3=N3+1
      N4=IABS(KMS(N3))
      IF(N4.NE.N8)GOTO 73964
      YS(N3)=YS(N3)+A
      GOTO 73963
73964  IF(N4.NE.N9)GOTO 73963
      YS(N3)=YS(N3)-A
      GOTO 73967
73965  IF(N9.GT.KPARTB)GOTO 73974
73967  IF(N9.GT.KPARTB)GOTO 73972
      N1=KKS(N9)
73970  N1=N1+1
      N2=IABS(KMS(N1))
      IF(N2.NE.N9)GOTO 73971
      YS(N1)=YS(N1)+A
      GOTO 73972
73971  IF(N2.NE.N8)GOTO 73970
      YS(N1)=YS(N1)-A
      GOTO 73970
73972  IALTER=1
73974  IF(K9899.GT.0)GOTO 73975
73973  GUS1=2.0*(GUS2-ANONL(I))
      VNONL(I)=VSL+GUS4
      ANONL(I)=ANONL(I)+GUS1
      IF(IPR.GT.0)
        WRITE(LO,63974)I,K,M,CURR(I),GUS2,GUS4,VSL,GUS1,VNONL(I),
        2ANONL(I),A
63974  FORMAT(/,' 98 ELEMENT ',3I10,F10.0,/, (1X,7E25.15))
73975  FINIT(K)=FINIT(K)-GUS1
      FINIT(M)=FINIT(M)+GUS1
73978  IT1=CURR(I)

```

```

73960 IF(NONLM(I).GT.0)GOTO 73990
      KCOUNT=KCOUNT+1
      BVALUE(KCOUNT)=GUS2
73990 CONTINUE
C
C
73991 CONTINUE
C
      IF(IALTER.NE.1)GOTO 1004
C
C CHANGE SWITCH BUS PART OF TRIANGULAR MATRIX
C
      II=IUPPER
      L=KPARTA
      GOTO 2290
2205 L=L+1
      IT2=KCTRL(L)
      IF(IT2.LE.0)GOTO 2290
      CALL MOVERO(F,NTOT)
      J=KKS(L)
2220 J=J+1
      I=KMS(J)
      IF(I.LT.0)GOTO 2230
      F(I)=F(I)+YS(J)
      GOTO 2220
2230 I=IABS(I)
      F(I)=F(I)+YS(J)
      IF(IT2.EQ.L)GOTO 2240
C
C ADDITION OF LOWER NUMBERED ROW WHEN SWITCH IS CLOSED
C
      J=KKS(IT2)
      IT2=L
      GOTO 2220
C
C ADDITION ELEMENT FOR CLOSED SWITCH
C
2240 IF(KSWTCH.EQ.0)GOTO 2248
      DO 2245 K=1,KSWTCH
      IF(IABS(KPOS(K)).GT.3)GOTO 2245
      N1=IABS(KHIGH(K))
      N2=IABS(KLOW(K))
      IF(KCTRL(N1).LT.0)GOTO 2245
      F(N1)=F(N1)+F(N2)
2245 CONTINUE
C *****
C
2248 K=KPARTA
      ACHECK=ABS(F(L))*1.E-20
      IF(ACHECK.NE.0.0)GOTO 2260
      GOTO 9200
2260 K=K+1
      IF(K.EQ.L)GOTO 2280
      IF(KCTRL(K).LE.0)GOTO 2260
      A=F(K)

```

```

73960 IF (NONLM(1).GT.0)GOTO 73990
      KCOUNT=KCOUNT+1
      BVALUE(KCOUNT)=GUS2
73990 CONTINUE
C
C
73991 CONTINUE
C
      IF (IALTER.NE.1)GOTO 1004
C
C CHANGE SWITCH BUS PART OF TRIANGULAR MATRIX
C
      II=IUPPER
      L=KPARTA
      GOTO 2290
2205 L=L+1
      IT2=KCTRL(L)
      IF (IT2.LE.0)GOTO 2290
      CALL MOVERO(F,NTOT)
      J=KKS(L)
2220 J=J+1
      I=KMS(J)
      IF (I.LT.0)GOTO 2230
      F(I)=F(I)+YS(J)
      GOTO 2220
2230 I=IABS(I)
      F(I)=F(I)+YS(J)
      IF (IT2.EQ.L)GOTO 2240
C
C ADDITION OF LOWER NUMBERED ROW WHEN SWITCH IS CLOSED
C
      J=KKS(IT2)
      IT2=L
      GOTO 2220
C
C ADDITION ELEMENT FOR CLOSED SWITCH
C
2240 IF (KSWTCH.EQ.0)GOTO 2248
      DO 2245 K=1,KSWTCH
      IF (IABS(KPOS(K)).GT.3)GOTO 2245
      N1=IABS(KHIGH(K))
      N2=IABS(KLOW(K))
      IF (KCTRL(N1).LT.0)GOTO 2245
      F(N1)=F(N1)+F(N2)
2245 CONTINUE
C *****
C
2248 K=KPARTA
      ACHECK=ABS(F(L))+1.E-20
      IF (ACHECK.NE.0.0)GOTO 2260
      GOTO 9200
2260 K=K+1
      IF (K.EQ.L)GOTO 2280
      IF (KCTRL(K).LE.0)GOTO 2260
      A=F(K)

```

```

IF(A.EQ.0.)GOTO 2260
I=KK(K)
2270 N1=KM(I)
IF(N1.LT.0)GOTO 2260
F(N1)=F(N1)-A*YKM(I)
I=I-1
GOTO 2270
2280 A=F(L)
IF(ABS(A).GE.ACHECK)GOTO 4312
GOTO 9200
4312 A=1.0/A
II=II+1
KM(II)=-L
YKM(II)=A
K=KPARTA
GOTO 2285
2283 K=K+1
IF(F(K).EQ.0.0)GOTO 2285
IF(KCTRL(K).EQ.0.0)GOTO 2285
IF(K.GT.L)GOTO 2284
IF(KCTRL(K).GT.0.0)GOTO 2285
2284 II=II+1
YKM(II)=F(K)*A
KM(II)=K
2285 IF(K.LT.NTOT)GOTO 2283
KK(L)=II
IF(II.LE.LYMAT)GOTO 2290
GOTO 9200
C *****
2290 IF(L.LT.KPARTB)GOTO 2205
GOTO 2300
1004 IF(ISTEP.NE.0)GOTO 2616
C -----
C START FIND DIFFERENCE OF COLUMNS OF INVERSE FOR NONLINEARITY
C NONLINEAR ELEMENT ; FIND VECTOR
C -----
2300 IF(NCOMP.LE.0)GOTO 2616
CALL MOVERO(KKNONL,NTOT)
N1=NTOT*NCOMP
CALL MOVERO(ZNONL,NTOT)
IF(INONL.EQ.NUM)GOTO 2321
DO 2320 I=1,INONL
IF(NLTYPE(I).LT.0)GOTO 2320
IF(KUPL(I).NE.0)GOTO 2320
VZERO(I)=-1.0E+20
N1=NONLAD(I)
ILAST(I)=IABS(N1)
IF(N1.LT.0)VZERO(I)=0.
2320 .CONTINUE
2321 L=1
DO 2322 N2=1,NUMSUB
N1=0
N7=ISUBEG(N2+1)
2306 D8=-1.0
J=KSUB(L)

```



```

DO 4323 N10=1,2
IF (J.GT.KPARTB)GOTO 2315
IF (J.EQ.1)GOTO 2315
N5=J+N1
ZNONL(N5)=D8
IF (KKNONL(J).EQ.0)GOTO 4322
IF (KKNONL(J).EQ.N2)GOTO 4322
4321 GOTO 9200
4322 KKNONL(J)=N2
2315 D8=1.0
4323 J=MSUB(L)
L=L+1
IF (L.GE.N7)GOTO 2322
N1=N1+NTOT
GOTO 2306
2322 CONTINUE
II=1
KLIMIT=IUPPER
ICHECK=0
N4=-KPARTB
2410 IF (II.GT.KLIMIT)GOTO 2450
L=IABS(KM(II))
N8=L
DO 2413 M=1,NCOMP
VOLTBC(M)=ZNONL(N8)
ZNONL(N8)=ZNONL(N8)*YKM(II)
2413 N8=N8+NTOT
N2=KKNONL(L)
J=IABS(KK(L))
2420 II=II+1
IF (II.GT.J)GOTO 2410
K=KM(II)
IF (KCTRL(K).LT.N4)GOTO 2420
N8=K
DO 2423 M=1,NCOMP
ZNONL(N8)=ZNONL(N8)-VOLTBC(M)*YKM(II)
2423 N8=N8+NTOT
N1=KKNONL(K)
IF (N2.EQ.0)GOTO 2420
IF (N1.EQ.N2)GOTO 2420
IF (N1.LE.0)GOTO 4334

```

```

C -----
4333 J=ISUBEG(N1)
K=KSUB(J)
M=MSUB(J)
BUS3=BUS(K)
BUS4=BUS(M)
J=ISUBEG(N2)
K=KBUS(J)
M=MBUS(J)
BUS5=BUS(K)
BUS6=BUS(M)
GOTO 9200
4334 KKNONL(K)=N2
GOTO 2420

```

```

2450 N13=NTOT*NCOMP
      IF (ICHECK.EQ.1)GOTO 2500
      ICHECK=1
      IF (KSWTCH.LE.0)GOTO 2483
      N7=1
      DO 2460 K=1,KSWTCH
      IF (IABS(KPOS(K)).GT.3)GOTO 2460
      I=KLOW(K)
      J=KHIGH(K)
      N1=IABS(J)
      N2=IABS(J)
      N3=N1
      IF (I.LE.0)GOTO 2453
      IF (J.LE.0)GOTO 2452
      I=KKNONL(N1)
      IF (I.EQ.0)GOTO 2451
      J=KKNONL(N2)
      KKNONL(N2)=I
      IF (J.EQ.0)GOTO 2451
      IF (J.EQ.1)GOTO 2451
      GOTO 4333
2451 DO 2455 M=1,NCOMP
      ZNONL(N2)=ZNONL(N2)+ZNONL(N1)
      N1=N1+NTOT
2455 N2=N2+NTOT
      KPOS2(K)=KKNONL(N3)
      GOTO 2454
2452 KPOS2(K)=KKNONL(N3)
      KKNONL(N1)=0
      GOTO 2454
2453 N3=N2
      KPOS2(K)=KKNONL(N3)
      KKNONL(N2)=0
2454 N8=N7+NCOMP-1
      DO 2456 M=N7,N8
      FSWABC(M)=ZNONL(N3)
      ZNONL(N3)=0.0
2456 N3=N3+NTOT
2460 CONTINUE
C -----
      N14=KSWTCH*NCOMP
      IF (IPR.GT.0)
        IWRITE(LO,2341) (ZNONL(K),K=1,N13)
2341 FORMAT(5X,'ZNONL',2X,5E20.10)
2483 IF (IEND.EQ.1UPPER)GOTO 2500
      KLIMIT=IEND
      N4=0
      GOTO 2410
2500 IF (II.EQ.1)GOTO 2550
      CALL MOVERO(VOLTBC,NCOMP)
      N2=0
2510 II=II-1
      K=KM(II)
      IF (KKNONL(K).GT.0)N2=KKNONL(K)
      N8=K

```

```

DO 2513 M=1,NCOMP
VOLTBC(M)=VOLTBC(M)-ZNONL(NB)*YKM(II)
2513 NB=NB+NTOT
GOTO 2510
2520 L=IABS(K)
IF(KKNONL(L).EQ.0)KKNONL(L)=N2
NB=L
DO 2526 M=1,NCOMP
ZNONL(NB)=ZNONL(NB)+VOLTBC(M)
2526 NB=NB+NTOT
IF(KCTRL(L).LE.0)GOTO 2500
J=KCTRL(L)
KKNONL(J)=KKNONL(L)
NB=J
N9=L
DO 2531 M=1,NCOMP
ZNONL(NB)=ZNONL(N9)
NB=NB+NTOT
2531 N9=N9+NTOT
GOTO 2500
C *****
2550 IF(INONL.EQ.NUM)GOTO 2616
DO 2600 I=1,INONL
IF(NLTYPE(I).LT.0)GOTO 2600
IF(KUPL(I).NE.0)GOTO 2600
N1=NONLK(I)
N2=IABS(NONLM(I))
A=ZNONL(N1)-ZNONL(N2)
IF(NLTYPE(I).LT.0)GOTO 2600
IF(KUPL(I).NE.0)GOTO 2600
N1=NONLK(I)
N2=IABS(NONLM(I))
A=ZNONL(N1)-ZNONL(N2)
IF(NLTYPE(I).EQ.94)GOTO 2561
IF(NONLE(I).GT.0)GOTO 2561
2557 A=A*DELTA2
2561 ANONL(I)=A
2600 CONTINUE
C
C *****
C
2616 CALL MOVER(FINIT,F,NTOT)
IF(IPR.GT.0)WRITE(LO,29919)(E(I),F(I),I=1,NTOT)
29919 FORMAT(5X,10(E25.15))
IPOINT=IPOINT+1
IF(IPOINT.EQ.LPAST)IPOINT=0
GOTO 9999
9200 CONTINUE
NCH=1
WRITE(LO,28881)NCH
28881 FORMAT(' <ERROR TS1> NCH',16)
9999 CONTINUE
RETURN
END

```

SUBROUTINE TS2

```

C
C TS2 CALLED BY SUBROUTINE TSTEP
C STEP 3 IN TIME STEP LOOP PROCESS
C
$INCLUDE: 'COMMON.FOR'
      DIMENSION XK(100),XM(100)
      EQUIVALENCE (H1,SK),(D2,SM)
      K=1
1100  IT2=LENGTH(K)
      N1=KBUS(K)
      IF(N1.LT.0)GOTO 1180
      IT2=IABS(IT2)
      IF(IT2.EQ.1)GOTO 1150
1105  IT1=K
      IF(LENGTH(K+1).EQ.-555)GOTO 3535
      DO 1110 I=1,IT2
      N1=KBUS(IT1)
      N2=MBUS(IT1)
      VOLTI(I)=-E(N1)
      VOLTK(I)=-E(N2)
      VOLT(I)=VOLTK(I)-VOLTI(I)
1110  IT1=IT1+2
      N3=NR(K)
C IF T>TMAX STOP THE PROGRAM
C
      IF(T.GT.TMAX)GOTO 1190
      CALL MULT(C,VOLTI,CI,IT2,-1)
      CALL MULT(C,VOLTK,CK,IT2,-1)
      CALL MOVER(CIK,VOLTI,IT2)
      CALL MULT(X,VOLT,VOLTI,IT2,1)
      CALL MULT(R,VOLT,VOLTI,CIK,IT2,-1)
      GOTO 3535
3530  CALL MULT(X,CIK,VOLT,IT2,1)
      CALL MULT(R,VOLT,CIK,IT2,-1)
3535  IT1=K+IT2-1
      DO 1120 I=K,IT1
      N1=KBUS(I)
      N2=MBUS(I)
      F(N1)=F(N1)-CIK(I)-CI(I)
1120  F(N2)=F(N2)+CIK(I)-CK(I)
      GOTO 1200
1150  I=NR(K)
      MSIGN =MBUS(K)
      N2 =IABS(MSIGN)
      GUS1=E(N1)
      GUS2=E(N2)
      GUS3=GUS1-GUS2
      A=CIK(K)
      CI1=CI(K)
      CK1=CK(K)
      IT1=IABS(I)
      CAP=C(IT1)
      IF(I.LT.0)GOTO 1170
      IF(T.GT.TMAX)GOTO 1105

```

```

A=(GUS3+A*X(I))*R(I)-A
GUS4=-CAP*GUS1-CI1
CK1=-CAP*GUS2-CK1
F(N1)=F(N1)-GUS4
F(N2)=F(N2)-CK1
1160 F(N1)=F(N1)-A
      F(N2)=F(N2)+A
      CI(K)=A
      CI(K)=GUS4
      CK(K)=CK1
      GOTO 1200
1170 GUS3=X(IT1)*GUS3
      GUS4=GUS3+A
      IF(CAP.EQ.0.)GOTO 1175
      CK1=CK1+CAP*(GUS4+CI1)
      GUS3=GUS3-2.0*X(IT)*CK1
1175 A=GUS3-R(IT1)*GUS4
      IF(MSIGN.GT.0)GOTO 1160
      GUS3=GUS4
      D1=N1
      IF(CK1.EQ.D1)GUS3=GUS4+CI(K-1)
      KCOUNT=KCOUNT+1
      BVALUE(KCOUNT)=GUS3
      GOTO 1160
1180 CONTINUE
      N1=-N1
      N2=MBUS(K)
      IF(IT2.GT.0)IT2=1

C
C --- TRANSFORMATION TO MODE VOLTAGE
C
1185 IT2=IABS(IT2)
      H1=E(N1)
      H2=E(N2)
      CI1=H1
      CK1=H2
      IT1=K+1
      II=LENGTH(IT1)
      IF(LENGTH(K).GT.0)II=LENGTH(K)
      GUS1=IT2
      N3=LTYPE(K)

C
C -----
C
1194 IT1=K
      DO 1195 I=1,IT2
      N1=-KBUS(IT1)
      N2=MBUS(IT1)
      VOLT(I)=E(N1)
      NDX1=LS6+I
      VOLT(NDX1)=E(N2)
1195 IT1=IT1+1
      DO 1197 I=1,IT2
      CI1=0.0
      CK1=0.0

```

```

NDX1=LS6
DO 1196 J=1,IT2
YX=QFD(N3)
CI1=CI1+YX*VOLT(J)
NDX1=NDX1+1
CK1=CK1+YX*VOLT(NDX1)
1196 N3=N3+1
VOLT(I)=CI1*GUS1
1197 VOLT(K(I))=CK1*GUS1
IF(IPR.GT.0)
1WRITE(LO,21197)(VOLT(I),VOLT(I),VOLT(K(I)),I=1,IT2)
21197 FORMAT(' VOLT(I) VOLT(I) VOLT(K(I) ARE',/, (1X,9E14.6))
NDX1=LS6+1
NDX2=LS6+IT2
GUS1=1.0

C
C UPDATE NODE QUANTITIES AND FIND PHASE VECTORS I
C
CI1=0
IF(MODOUT.LE.0)GOTO 4641
D2=IT2
D2=1.0/D2
DO 4633 I=1,MODOUT
N7=NV+1
BVALUE(N7)=VOLT(I)*D2
N8=NV+MODOUT
4633 BVALUE(N8)=VOLT(K(I))*D2
-----
C
4641 CK1=0.
MSIGN=K
41189 I=I+1
N3=NR(MSIGN)
IF(N3.LT.0)GOTO 1187
IT1=I+IPPOINT
N4=IT1+N3
IF(IT1.GT.LPAST)IT1=IT1-LPAST
IF(N4.GT.LPAST)N4=N4-LPAST
IT21=IT1+1
N41=N4+1
IF(IT21.GT.LPAST)IT21=1
IF(N41.GT.LPAST)N41=1
XMN4=XM(N4)
A=CIK(MSIGN)
D2=1.0-A
GUS2=CI(MSIGN)
H1=ABS(GUS2)
H2=ABS(CK(MSIGN))
XM(N4)=H1*VOLT(I)-H2*XK(N4)
XK(N4)=H1*VOLT(K(I))-H2*XMN4
IF(CK(MSIGN).LT.0.0)THEN
NCH=21
GOTO 9200
ENDIF
H1=XK(IT1)*A+XK(IT21)*D2
D2=XM(IT1)*A+XM(IT21)*D2

```

```

IF (GUS2.LT.0.0)GOTO 1192
A=H1+D2
D2=(H1-D2)*H2
H1=(A+D2)/2.0
D2=(A-D2)/2.0
1192  XK(N41)=H1
      XM(N41)=D2
1188  II=II+N3+2
C -----
      GOTO 10130
C
C CALCULATE EQUIVALENT SOURCE
C
10130  VOLTI(I)=H1*GUS1
      VOLTK(I)=D2*GUS1
      CI1=CI1+H1
      CK1=CK1+D2
      MSIGN=MSIGN+1
      GOTO 1189
1187  H1=0.
      D2=0.0
      GOTO 1188
1189  IF(I.LT.IT2)GOTO 41189
      J=LITYPE(K)
C
C          IPHASE = II*IMODE
C          OUTPUT DESIRED CURRENT
C
N3=J-1
DO 1203 J=1,IT2
H1=0
H2=0
N3=N3+1
N4=N3
DO 1199 I=1,IT2
YX=OFD(N4)
H1=H1+YX*VOLTI(I)
H2=H2+YX*VOLTK(I)
1199  N4=N4+IT2
      N1=-KBUS(K)
      N2=MBUS(K)
      F(N1)=F(N1)+H1
      F(N2)=F(N2)+H2
      IF(IPR.GT.0)
1WRITE(LO,2199)J,IT2,N1,N2,H1,H2,F(N1),F(N2)
2199  FORMAT(' J IT2 N1 N2 H2 F(N1) F(N2)',/,
14I8,4(1X,1E15.6))
1203  K=K+1
C
C END OF CONVERSION TO IPHASE FOR UNTRANSPOSED OPTION
C
1204  IT2=0
C
1190  CONTINUE
1200  K=K+IT2
      IF(IPR.GT.0)

```



```
1WRITE(LO,7342)K,IBR,(F(I),I=1,NTOT)
7342  FORMAT(/, ' K  IBR =',2I6,10X, ' F',/, (1X,8E16.7))
      IF(K.LE.IBR)GOTO 1100
C --- IF > IBR GOTO 018
9200  CONTINUE
      NCH=2
      WRITE(LO,14414)NCH
14414  FORMAT(' <ERROR TS2> NCH',I6)
9999  GOTO 99999
99999  RETURN
      END
```



```

SUBROUTINE TS3
C
C   TS3 CALLED BY SUBROUTINE TSTEP
C   STEP 4-11 IN TIME STEP LOOP
C-----
C   SOLVE THE SOLUTION
C-----
C
$INCLUDE: 'COMMON.FOR'
      DIMENSION PEAKND(3), ILAST(30)
      IF(PEAKND(1).EQ.0.0)GOTO 1003
      D1=ABS(PEAKND(1))
      DO 1002 J=2,NTOT
      IF(ABS(E(J)).LE.D1)GOTO 1002
      PEAKND(1)=E(J)
      D1 =ABS(E(J))
      PEAKND(2)=T
      PEAKND(3)=J
1002  CONTINUE
1003  IF(ISTEP.LE.9999)GOTO 12
      KILL=1
      NCH=31
      GOTO 9200
12    CONTINUE
13    IF(NV.EQ.0)GOTO 1202
      DO 1201 K=1,NV
      N1=IBRNCH(K)
      N2=JBRNCH(K)
1201  BVALUE(K)=E(N1)-E(N2)
1202  VOLTI(1)=T
      K=1
1600  IF(K.GT.NUMV)GOTO 1630
      I=IBSOUT(K)
      K=K+1
      VOLTI(K)=E(I)
      GOTO 1600
1610  K=NTOT
      CALL MOVER(E(2),VOLTI(2),NTOT1)
1630  IF(NC.EQ.0)GOTO 1642
C   NODEV :- NUMBER OF VOLTAGE NODE OUTPUT
C
      NODEV=K
      DO 1640 I=1,NC
      K=K+1
1640  VOLTI(K)=BVALUE(I)
1642  CONTINUE
C   IF(IAVERG.EQ.0)GOTO 1652
      IF(ISTEP.EQ.0)GOTO 1657
      GOTO 1652
1657  DO 1658 I=2,K
      NDX1=LS6+I
1658  VOLTI(NDX1)=VOLTI(I)
1652  CONTINUE
54284 IF(IOUTCS.EQ.0)GOTO 1650
      NDX1=KKOUT+1

```

```

CALL MOVER(OUT(NDX1),VOLT1(K+1),IOUTCS)
K=K+IOUTCS
1650 CONTINUE
      NI=K
      IF(NI.GT.10)NI=10
      IF(T.GE.1.)GOTO 48
      IF(DELTA.T.LT.1E-6)GOTO 48
      GOTO 56
48    IF(ISTEP.LE.9000)GOTO 39
      GOTO 56
C    PRINT OUTPUT RESULTS
C
39    WRITE(LO,40)ISTEP,(VOLT(I),I=1,NI)
40    FORMAT(1X,I4,E10.3,9E13.6)
56    IF(K.GT.NI)WRITE(LO,87400)(VOLT1(I),I=11,K)
87400 FORMAT(15X,9E13.6)
C
C    UPDATE XMAX,TMAX,XMIN,TMIN
C    STORE IN XMAX AT 4 PARTITIONS
C
      DO 1655 I=2,K
      L=I-1
      NDX1=2*LPEAK+L
      IF(VOLT1(I).GE.XMAX(NDX1))GOTO 4655
      XMAX(NDX1)=VOLT1(I)
      NDX1=NDX1+LPEAK
      XMAX(NDX1)=T
4655  IF(VOLT1(I).LE.XMAX(L))GOTO 1655
C
C    XMAX-TMAX
C
      XMAX(L)=VOLT1(I)
      NDX1=LPEAK+L
      XMAX(NDX1)=T
1655  CONTINUE
C -----
1661  ISTEP=ISTEP+1
      T=T+DELTA.T
      IF(T.GT.TMAX)THEN
      NCH=9
      GOTO 9999
      ENDIF
      IF(IPR.GT.0)THEN
      WRITE(LO,73140)K,IBR
      WRITE(LO,73141)(F(I),I=1,NTOT)
73140  FORMAT(' K IBR=',2I6)
73141  FORMAT(10X,'F(1)-1,NTOT',/, (1X,8E16.7))
      ENDIF
      ICYCLE=ICYCLE+1
      CALL MOVER(VOLTBC,10)
C
C    READ OR COMPUTE SPECIFY VOLTAGE
C
1249  I=I+1
      IF(I.GT.KCONST)GOTO 1300

```

```

N1=NODE(I)
N2=IABS(N1)
N3=IFORM(I)
1251 YX=0.
    TS=T-TSTART(I)
    IF(TS.LT.0.)GOTO 1250
    IF(T.GE.TSTOP(I))GOTO 1250
    K=IABS(N3)
    IF(K.EQ.14)GOTO 1270
    IF(K.EQ.15)GOTO 1286
1387 YX=CREST(I)
    IF(K.GE.12)GOTO 1280
1250 IF(N1.GT.0)GOTO 1260
    F(N2)=F(N2)+YX
    GOTO 1249
1260 IF(N3.LT.0)YX=YX+E(N2)
    E(N2)=YX
    GOTO 1249
1270 YX=CREST(I)*COS(SFREQ(I)*TS+TIME1(I))
    GOTO 1250
1280 IF(TS.GE.TIME1(I))GOTO 1285
    YX=TS/TIME1(I)*YX
    GOTO 1250
1285 IF(K.EQ.12)GOTO 1250
    YX=YX+SFREQ(I)*(TS-TIME1(I))
    GOTO 1250

C
C STANDARD FUNCTION
C
1286 YX=CREST(I)*(EXP(SFREQ(I)*TS)-EXP(TIME1(I)*TS))
    GOTO 1250

C
C ----- ELIMINATION PROCESS -----
C FORWARD SOLUTION
C
1300 II=1
    KLIMIT=IUPPER
    IF(IPR.GT.0)WRITE(LO,41300)(F(L),L=1,NTOT)
41300 FORMAT(/,' F(L),L=1,NTOT AT B.R. SOLUTION',/,(1X,SE2S.15))
    IF(IPR.GT.0)WRITE(LO,41320)(L,KM(L),YKM(L),L=1,IEND)
41320 FORMAT(/,' TABLE OF FACTOR L-KM-YKM',/,(2I10,E15.5,5X,2I10,E15.5))
    ICHECK=0
1410 IF(II.GT.KLIMIT)GOTO 1450
    L=IABS(KM(II))
    A=F(L)
    F(L)=A*YKM(II)
    J=KK(L)
1420 II=II+1
    IF(II.GT.J)GOTO 1410
    K=KM(II)
    F(K)=F(K)-A*YKM(II)
    GOTO 1420
1450 IF(ICHECK.EQ.1)GOTO 1500
    ICHECK=1
    KLIMIT=IEND
    IF(KSWTCH.LE.0)GOTO 1410

```

```

C -----
C REFLECT SWITCH POSITION IN [Itotal],e(t)
C -----
DO 1460 K=1,KSWTCH
IF (IABS(KPOS(K)).GT.3)GOTO 1460
I=KLOW(K)
J=KHIGH(K)
N1=IABS(I)
N2=IABS(J)
FSWTCH(K)=F(N1)
IF (I.GT.0)GOTO 1455
FSWTCH(K)=F(N2)
E(N2)=E(N1)
GOTO 1460
1455 F(N2)=F(N2)+F(N1)
IF (J.GT.0)GOTO 1460
E(N1)=E(N2)
1460 CONTINUE
GOTO 1410

C
C BACKSUBSTITUTION TO SOLVE THE SOLUTION
C
1500 IF (II.EQ.1)GOTO 1550
A=0
1510 II=II-1
K=KM(II)
IF (K.LT.0)GOTO 1520
A=A-E(K)*YKM(II)
GOTO 1510
1520 L=IABS(K)
E(L)=F(L)+A
IF (KCTRL(L).LE.0)GOTO 1500
J=KCTRL(L)
E(J)=E(L)
GOTO 1500

C
C NONLINEAR ELEMENT : CORRECT VOLTAGE
C
1550 IF (INONL.EQ.NUM)GOTO 9999
DO 1559 I=1,INONL
IF (NLTYPE(I).LT.0)GOTO 1559
IF (KUPL(I).EQ.0)GOTO 3534
IF (KUPL(I).GT.1)GOTO 1559
3534 I1=NONLAD(I)
I2=NONLE(I)
K=NONLK(I)
L=IABS(NONLM(I))
A2=ANONL(I)
N1=ILAST(I)
VZERO2=E(K)-E(L)
IF (I1.LT.0)GOTO 1557
IF (I2.GT.0)GOTO 1556
I2=IABS(I2)
VZERO2=VZERO2+DELTA2+VNONL(I)
1551 VD1FF=VZERO2-VZERO(I)

```

```

VZERO(I)=VZERO2
IF(VDIFF.LT.0.0)GOTO 3400
IF(VDIFF.EQ.0.0)GOTO 1555
VSL=VZERO2+A2*CCHAR(N1)
H1=VSL-VCHAR(N1)
GOTO 3800
3100 N1=N1+1
IF(N1.LT.I2)GOTO 3200
KILL=1
NCH=32
GOTO 9200
3200 H1=H2
3800 VSR=VZERO2+A2*CCHAR(N1+1)
H2=VSR-VCHAR(N1+1)
IF(H1*H2.GT.0.)GOTO 3100
3300 DELTI=ABS(H1)*(CCHAR(N1+1)-CCHAR(N1))/(ABS(H1)+ABS(H2))
CURR(I)=CCHAR(N1)+DELTI
GOTO 1555
3500 N1=N1-1
IF(N1.GE.I1)GOTO 3475
NCH=33
GOTO 9200
3400 VSR=VZERO2+A2*CCHAR(N1+1)
H2=VSR-VCHAR(N1+1)
GOTO 3450
3475 H2=H1
3450 VSL=VZERO2+A2*CCHAR(N1)
H1=VSL-VCHAR(N1)
IF(H1*H2)3300,3300,3500
1555 ILAST(I)=N1
GOTO 8259
1556 IF(VNONL(I).EQ.0)GOTO 1551
IF(ABS(VZERO2).LT.VNONL(I))GOTO 8259
WRITE(LO,15569)BUS(K),BUS(L),T
15569 FORMAT(' FIRES AT TIME',E14.6,'SEC')
VNONL(I)=0.0
GOTO 1551
1557 IF(VNONL(I).EQ.0.)GOTO 1558
IF(ABS(VZERO2).LT.VNONL(I))GOTO 8259
C
C FIRING TIME TYPE 92
C -----
VNONL(I)=0.
VZERO(I)=T
1558 A=T-VZERO(I)
DO 15581 K=N1,I2
IF(A.LT.CCHAR(K))GOTO 15582
15581 CONTINUE
KILL=2
NCH=34
GOTO 9200.
15582 N1=K-1
H1=CCHAR(N1)
VSL=VCHAR(N1)
VDIFF=(A-H1)*(VCHAR(K)-VSL)/(CCHAR(K)-H1)+VSL

```

คู่มือการใช้โปรแกรม

โปรแกรมการวิเคราะห์ทรานเซียนต์แม่เหล็กไฟฟ้าในระบบไฟฟ้ากำลัง (Electromagnetic Transient Analysis in Electric Power System)

การวิเคราะห์ทรานเซียนต์แม่เหล็กไฟฟ้าประกอบด้วยโปรแกรมที่ใช้ในการวิเคราะห์ 2 โปรแกรม ซึ่งอิสระต่อกันคือ โปรแกรมคำนวณพารามิเตอร์ของสายส่งและโปรแกรมวิเคราะห์ทรานเซียนต์แม่เหล็กไฟฟ้า

โปรแกรมเขียนด้วยภาษาฟอร์แทรน (Fortran 77) สำหรับไมโครคอมพิวเตอร์ของ ไอบีเอ็ม (IBM) ขนาด 16 บิต คอมไพเลอร์ (Compile) ด้วยฟอร์แทรนคอมไพเลอร์ของบริษัท ไมโครซอฟท์ (Microsoft) สามารถเรียกใช้โดยตรงภายใต้ดอส (DOS) โปรแกรมที่ผู้ใช้ (User) สามารถเรียกใช้งานได้คือ

1. LCON.EXE เป็นโปรแกรมใช้คำนวณพารามิเตอร์ของสายส่ง รายละเอียดของ สับรูกินย่อยแสดงในภาคผนวก ข. การป้อนข้อมูลแสดงในบทที่ 5
2. EMTA.EXE เป็นโปรแกรมใช้วิเคราะห์ทรานเซียนต์แม่เหล็กไฟฟ้า รายละเอียด ของสับรูกินย่อยอธิบายในบทที่ 4

โปรแกรมทั้งสองสามารถแยกอยู่ในดิสก์คนละแผ่นได้ เนื่องจากโปรแกรมทั้งสองแยก อิสระต่อกัน

ไฟล์ต่าง ๆ ที่เกี่ยวกับข้อมูลจะต้องอยู่ในดิสก์เดียวกันขณะใช้โปรแกรม แต่อาจจะอยู่ คนละดิสก์กับโปรแกรมได้ ลักษณะของไฟล์ข้อมูลแบ่งออกเป็น 3 ชนิดคือ

1. ไฟล์ข้อมูลอินพุตจะใส่ชื่ออะไรก็ได้ แต่เพื่อสะดวกในการใช้งานอาจใช้นามสกุล (Extension) ของชื่อไฟล์นั้นเป็น INP เช่น DATA1.INP เป็นต้น
2. ไฟล์ข้อมูลเอาท์พุตจะใส่ชื่ออะไรก็ได้ แต่เพื่อความสะดวกในการใช้งานอาจใช้นามสกุลของชื่อไฟล์เป็น OUT
3. ไฟล์ข้อมูลสำหรับพลอตเช่นเดียวกับไฟล์สองแบบแรก อาจใช้นามสกุลของชื่อไฟล์ เป็น PLT

ขนาดของไมโครคอมพิวเตอร์และส่วนประกอบควรมีดังนี้

1. ไมโครคอมพิวเตอร์ของ IBM ขนาด 16 บิต หรือไมโครคอมพิวเตอร์ที่มีความสามารถเทียบเท่าซึ่งมีหน่วยความจำอย่างน้อย 256 กิโลไบต์
2. ฟลอปปีดิสก์ไดรฟ์ 2 ตัว หรือฟลอปปีดิสก์ไดรฟ์ 1 ตัวและฮาร์ดดิสก์ไดรฟ์ 1 ตัว
3. MS-DOS Operating System
4. คัสเคอรักรายฟิคส์สไลด์อะแดปเตอร์สำหรับการแสดงภาพ
5. คอสมเมตริกซ์พริ้นเตอร์ของ EPSON (Dot Matrix Printer)

การใช้งานเริ่มต้นด้วยพิมพ์คำว่า EMTA จากนั้นโปรแกรมจะทำการถามชื่อของ
ฮาร์ดไดรฟ์เอาท์พุทไฟล์และพลดไฟล์ตามลำดับ ในขั้นตอนการอ่านข้อมูลแสดงรายละเอียดของ
การถามตอบของโปรแกรมดังตาราง จ.1 แสดง Help File ดังตาราง จ.2 และเอาท์พุทไฟล์
ของตัวอย่างในตาราง จ.3

TO MEASURE THE STEP RESPONSE
 INPUT VOLTAGE STEP FUNCTION 1 PU.

 5.E-9,4.E-7,,,1,1,1
 BUS1 BUS2 0
 500.0,,,
 BUS3 0
 0,,5.E-6
 BUS3 BUS4 0
 232.0,,,
 BUS4 BUS3 0
 BUS4 BUS5 BUS3 BUS4 0
 BUS5 BUS3 0
 BUS5 BUS6 BUS3 BUS4 0
 BUS6 BUS3 0
 BUS6 BUS7 BUS3 BUS4 0
 BUS7 BUS3 0
 BUS7 BUS8 0
 208.8,,,
 BUS8 0
 23.2,,,
 -1BUS2 BUS3 0
 0.0,272.0,20.,10.,1
 88
 88
 11BUS1
 1,20.0,0.,0.,0.,0.,0.,0
 88
 BUS 3
 999999

67573 11.1


```

-----*
*
*   HELP PROCEDURE TO USE THE PROGRAM   *
*
*   GROUP OF DATA IS REQUIRED           *
*
*-----*

```

NAME OF INPUT FILE

NAME OF OUTPUT FILE

NAME OF PLOT FILE

GROUP 1 CASE TITLE 3 LINES, 80 COLUMN/LINE

INPUT CASE TITLE

GROUP 2 CONTROL DATA

DELTA : TIME STEP SIZE IN SEC
TMAX : TERMINATED TIME IN SEC
XOPT : 0 FOR OH
 f FOR OHM
COPT : 0 FOR MICROFARAD
 f FOR MHO
IPR : 0 NO PRINT DIAGNOSTIC VARIABLE
 1 PRINT
IPRNT : NUMBER OF STEP TO PRINT IN O/P FILE
 NUMBER OF STEP TO PLOT IN PLOT FILE

INPUT TIME STEP SIZE, TERMINATION TIME XOPT, COPT

GROUP 3 BRANCH DATA FOR LUMPED PARAMETERS IN SERIES

3.1 ITYPE : CODE OF DATA TYPE
 BUS1 : BUS NAME K
 BUS2 : BUS NAME M
 BUS3 : BUS NAME K
 BUS4 : BUS NAME M
 IOUT : OUTPUT I SELECTION

3.2 R : RESISTANCE IN OHM

L : INDUCTANCE OR REACTANCE DEPEND
XOPT

C : CAPACITANCE OR SUSCEPTANCE DEPEND
COPT

GROUP 4 BRANCH DATA FOR PI-EQUIVALENT

4.1 SAME AS 3.1

4.2 TR : RESISTANCE

TX : REACTANCE

C : SUSCEPTANCE

FOR 3 PHASES :-

TR TX C
TR TX C TR TX C
TR TX C TR TX C TR TX C

GROUP 5 BRANCH DATA FOR DISTRIBUTED PARAMETERS

5.1 SAME AS 3.1

5.2 H1 : RESISTANCE/UNIT

AA : IF ILINE =0 MEAN INDUCTANCE/UNIT
1 MEAN SURGE IMPEDANCE

X3 : IF ILINE =0 MEAN CAPACITANCE/UNIT
=1 MEAN TRAVEL TIME

XLONG : LENGTH OF LINE

ILINE : SPECIFY FIELD AA,X3

GROUP 6 BRANCH DATA FOR NONLINEAR PARAMETERS

6.1 SAME AS 3.1

6.2 A1 & A2 RELATION

IF TYPE = 98 OR 99 : A1 MEAN CURRENT IN AMPS

IF TYPE =98 A2 MEAN FLUX
(NONLINEAR INDUCTANCE)

=99 A2 MEAN VOLTAGE
(NONLINEAR RESISTANCE)

GROUP 7 SWITCH DATA

222

IT22 : 20 FOR SWITCH DATA
BUS1 : BUS K
BUS2 : BUS M
GUS3 : TIME TO CLOSE
GUS4 : TIME TO OPEN

GROUP 8 SOURCE DATA

N2 : SOURCE TYPE
11 - STEP FN.
12 - RAMP FN.
13 - STANDARD FN.
14 - AC VOLTAGE SOURCE
BUS1 : SOURCE BUS NAME
A : AMPLITUDE FOR TYPE 13,14
AMPLITUDE CORRESPONDING TO
FOR TYPE =11,12
D1 : FREQUENCY IN HZ FOR TYPE=14,
ALPHA PARAMETER FOR TYPE=13
GUS2 : TO CORRESPONDING (A) FOR N=11,12
EQUAL BETA FOR TYPE=13
H1 : AMPLITUDE CORRESPOND TO TIME OF H2
H2 : TIME CORRESPOND TO H1
GUS3 : TSTART IN SEC
GUS4 : TSTOP IN SEC

GROUP 9 NODE SELECT VOLTAGE OUTPUT

9 NODES VOLTAGE SELECTION (MAX.)
6 CHARACTERS FOR EACH NODE NAME

GROUP 10 END INPUT 999999

OUTPUT RESULT
=====

```

*****
*
*   A COMPUTER METHOD OF ELECTROMAGNETIC TRANSIENT   STUDIES *
*   IN ELECTRIC POWER SYSTEM                       *
*
*****

```

=====

CASE STUDY TITLE

=====

LINE # 1 TO MEASURE THE STEP RESPONSE
 LINE # 2 INPUT VOLTAGE STEP FUNCTION 1 PU.
 LINE # 3 -----

CONTROL CODE DATA

=====

<DELTAT> <TMAX> <XOPT> <COPT> <IPR> <IPRNT> <IPLT>
 .50000000E-08 .40000000E-06 .00000000E+00 .00000000E+00 1 1 1

BRANCH DATA

=====

OBUS1	BUS2			0
SERIES R=L=C	1	.50000E+03	.00000E+00	.00000E+00
OBUS3				0
SERIES R=L=C	2	.00000E+00	.00000E+00	.50000E-05
OBUS3	BUS4			0
SERIES R=L=C	3	.23200E+03	.00000E+00	.00000E+00
OBUS4	BUS3			0
OBUS4	BUS5	BUS3	BUS4	0
OBUS5	BUS3			0
OBUS5	BUS6	BUS3	BUS4	0
OBUS6	BUS3			0
OBUS6	BUS7	BUS3	BUS4	0
OBUS7	BUS3			0
OBUS7	BUS8			0
SERIES R=L=C	4	.20880E+03	.00000E+00	.00000E+00
OBUS8				0
SERIES R=L=C	5	.23200E+02	.00000E+00	.00000E+00
-1BUS2	BUS3			0
< R>	<Z-L>	<T-C>	<IL>	.000 272.000 20.000 10.000 1
88				0

SWITCH DATA

=====

TYPE	BUS1	BUS2	T-CLOSE	T-OPEN
88			.0000E+00	.0000E+00

SOURCE DATA

=====

TYPE	BUS1	11BUS1		
SOURCE	.2000E+02	.0000E+00	.0000E+00	.0000E+00

TYPE BUS1 88

NODE VOLTAGE SELECTION BUS3

LUMPED PARAMETER STORAGE TR, TX, C

1	.500000E+03	.000000E+00	.000000E+00
2	.000000E+00	.000000E+00	.500000E-05
3	.232000E+03	.000000E+00	.000000E+00
4	.208800E+03	.000000E+00	.000000E+00
5	.232000E+02	.000000E+00	.000000E+00

SOURCE TABLE VECTOR

ROW-IFORM-NODE-CREST-TIME-TSTART-SFREQ

1	11	2	.2000E+02	.0000E+00	.0000E+00	.0000E+00
---	----	---	-----------	-----------	-----------	-----------

LIST OF INPUT ELEMENTS CONNECTED TO EACH BUS.

- 1) ONLY THE PHYSICAL CONNECTIONS OF MULTIPHASE LINES ARE SHOWN (CAPACITIVE AND INDUCTIVE COUPLING IGN)
- 2) REPEATED ENTRIES IMPLY PARALLEL CONNECTIONS
- 3) SOURCES ARE OMITTED, ALTHOUGH SWITCHES ARE INCLUDED

FROM BUS NAME 1 NAMES OF ALL ADJACENT BUSESSES

```

+          1
BUS1 1BUS2 *
BUS2 1BUS1 *BUS3 *
BUS3 1TERRA *BUS2 *BUS4 *
BUS4 1TERRA *BUS3 *BUS5 *
BUS5 1TERRA *BUS4 *BUS6 *
BUS6 1TERRA *BUS5 *BUS7 *
BUS7 1TERRA *BUS6 *BUS8 *
BUS8 1TERRA *BUS7 *
TERRA 1BUS3 *BUS4 *BUS5 *BUS6 *BUS7 *BUS8 *

```

```

+          1
BUS NAME          BUS1          BUS2          BUS3          BUS4          BUS5          BUS6          BUS7          BUS8

```

<u>step</u>	<u>time</u>	<u>(ns)</u>	<u>bus1</u>	<u>bus2</u>	<u>bus3</u>	<u>bus7</u>	<u>bus8</u>
0	0	0	0	0	0	0	0
1	5	1	0.352332	0	0	0	0
2	10	1	0.352332	0	0	0	0
3	15	1	0.352332	0	0	0	0
4	20	1	0.352332	0	0	0	0
5	25	1	0.352332	0.332737	0.016936	0.001693	
6	30	1	0.352332	0.562768	0.064686	0.006468	
7	35	1	0.352332	0.539382	0.104257	0.010425	
8	40	1	0.352332	0.576875	0.111255	0.011125	
9	45	1	0.32695	0.563874	0.112491	0.011249	
10	50	1	0.624918	0.574122	0.114383	0.011438	
11	55	1	0.594625	0.568362	0.113786	0.011378	
12	60	1	0.643191	0.572364	0.114283	0.011428	
13	65	1	0.62635	0.564272	0.113823	0.011382	
14	70	1	0.639624	0.626468	0.116372	0.011637	
15	75	1	0.632163	0.653431	0.124605	0.012460	
16	80	1	0.637347	0.665133	0.131681	0.013168	
17	85	1	0.634362	0.676017	0.133595	0.013359	
18	90	1	0.626926	0.672665	0.134492	0.013449	
19	95	1	0.683752	0.676477	0.134994	0.013499	
20	100	1	0.671614	0.674316	0.134935	0.013493	
21	105	1	0.690696	0.676105	0.135109	0.013510	
22	110	1	0.682424	0.672902	0.134945	0.013494	
23	115	1	0.699564	0.694875	0.135344	0.013534	
24	120	1	0.685234	0.690265	0.136736	0.013673	
25	125	1	0.688433	0.691888	0.138003	0.013800	
26	130	1	0.68648	0.694324	0.138439	0.013843	
27	135	1	0.685206	0.693691	0.138727	0.013872	
28	140	1	0.695774	0.694798	0.138845	0.013884	
29	145	1	0.692243	0.694145	0.13886	0.013886	
30	150	1	0.697839	0.694781	0.138907	0.013890	
31	155	1	0.69491	0.693782	0.138866	0.013886	
32	160	1	0.697622	0.696202	0.13893	0.013893	
33	165	1	0.695832	0.696808	0.139162	0.013916	
34	170	1	0.697233	0.697435	0.139389	0.013938	
35	175	1	0.696315	0.697864	0.13948	0.013948	
36	180	1	0.696329	0.697807	0.139555	0.013955	
37	185	1	0.698157	0.698066	0.13958	0.013958	
38	190	1	0.697315	0.697907	0.139591	0.013959	
39	195	1	0.698736	0.698094	0.139602	0.013960	
40	200	1	0.697862	0.697819	0.139594	0.013959	
41	205	1	0.698726	0.698329	0.139604	0.013960	
42	210	1	0.698106	0.698341	0.139643	0.013964	
43	215	1	0.698619	0.698541	0.139683	0.013968	
44	220	1	0.698259	0.69859	0.139701	0.013970	
45	225	1	0.69838	0.698609	0.139718	0.013971	
46	230	1	0.698644	0.698657	0.139724	0.013972	
47	235	1	0.698483	0.698626	0.139727	0.013972	
48	240	1	0.698805	0.698673	0.13973	0.013973	
49	245	1	0.698574	0.698604	0.139728	0.013972	
50	250	1	0.69882	0.698715	0.13973	0.013973	
51	255	1	0.698628	0.698693	0.139736	0.013973	
52	260	1	0.698796	0.698751	0.139744	0.013974	
53	265	1	0.69867	0.698748	0.139747	0.013974	

54	270	!	0.698736	0.698762	0.139751	0.013975
55	275	!	0.698755	0.698767	0.139752	0.013975
56	280	!	0.698735	0.698764	0.139753	0.013975
57	285	!	0.698799	0.698774	0.139754	0.013975
58	290	!	0.698745	0.698758	0.139753	0.013975
59	295	!	0.698808	0.698782	0.139754	0.013975
60	300	!	0.698754	0.698772	0.139755	0.013975
61	305	!	0.698804	0.698789	0.139756	0.013975
62	310	!	0.698764	0.698784	0.139757	0.013975
63	315	!	0.69879	0.69879	0.139757	0.013975
64	320	!	0.698783	0.698789	0.139758	0.013975
65	325	!	0.698785	0.69879	0.139758	0.013975
66	330	!	0.698795	0.698791	0.139758	0.013975
67	335	!	0.698785	0.698789	0.139758	0.013975
68	340	!	0.698799	0.698793	0.139758	0.013975
69	345	!	0.698785	0.69879	0.139758	0.013975
70	350	!	0.698799	0.698794	0.139758	0.013975
71	355	!	0.698787	0.698792	0.139759	0.013975
72	360	!	0.698796	0.698795	0.139759	0.013975
73	365	!	0.698792	0.698794	0.139759	0.013975
74	370	!	0.698794	0.698794	0.139759	0.013975
75	375	!	0.698795	0.698794	0.139759	0.013975
76	380	!	0.698793	0.698794	0.139759	0.013975
77	385	!	0.698796	0.698795	0.139759	0.013975
78	390	!	0.698793	0.698794	0.139759	0.013975
79	395	!	0.698796	0.698795	0.139759	0.013975
80	400	!	0.698793	0.698794	0.139759	0.013975

ตาราง 2.3





๑๑๗ ✓

ประวัติผู้เขียน

นายวิชัย ลีวรรณันตกุล เกิดเมื่อวันที่ 8 ตุลาคม พ.ศ. 2501 ณ จังหวัดกรุงเทพฯ
สำเร็จการศึกษาวិชากรรมศาสตร์บัณฑิตจากคณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
เมื่อปี พ.ศ. 2524 หลังจากสำเร็จการศึกษาแล้วได้เข้าทำงานที่การไฟฟ้าฝ่ายผลิตฯ ฝ่ายบำรุง
รักษาอุปกรณ์ระบบส่ง ทำหน้าที่วิเคราะห์ระบบไฟฟ้าเป็นเวลาประมาณ 6 ปี ปัจจุบันดำรง
ตำแหน่งวิศวกรระดับ 5 แผนกวิเคราะห์ระบบ กองระบบรีเลย์ ฝ่ายบำรุงรักษาอุปกรณ์ระบบส่ง
การไฟฟ้าฝ่ายผลิตแห่งประเทศไทย