NASA Communications Network
Design and Costing Model
A Communications Network Design and Costing (CNDC) computer model provides the capability for analyzing long-haul communication networks. The model is well-suited for market assessment studies involving the integration of developing satellite communication systems into the environment of existing service offerings of licensed domestic and specialized interstate communications common carriers. A capability is provided for analyzing long-haul trunking networks comprising a set of user-defined cities, traffic conditions, and tariff rates. Networks may consist of all terrestrial connectivity, all satellite connectivity, or a combination of terrestrial and satellite connectivity. Network solutions provide the least-cost routes between all cities, the least-cost network routing configuration, and terrestrial and satellite service cost totals. The CNDC model allows analyses involving three specific FCC-approved tariffs, which are uniquely structured and representative of most existing service connectivity and pricing philosophies. User-defined tariffs that can be variations of these three tariffs are accepted as input to the model and allow considerable flexibility in network problem specification.

The resulting model extends the domain of network analysis from traditional fixed link cost (distance-sensitive) problems to more complex problems involving combinations of distance- and traffic-sensitive tariffs. A heuristic algorithm is developed to determine minimum cost network routing solutions specifying the location of satellite access cities and their hubbing terrestrial extensions for typical customer premise services.

This manual presents a detailed description of the CNDC computer model from the programmer's point of view.
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0 Introduction</td>
<td>1-1</td>
</tr>
<tr>
<td>2.0 Optimization Algorithms</td>
<td>2-1</td>
</tr>
<tr>
<td>2.1 Network Optimization Techniques</td>
<td>2-1</td>
</tr>
<tr>
<td>2.1.1 Introduction</td>
<td>2-1</td>
</tr>
<tr>
<td>2.1.2 Network Terminology</td>
<td>2-1</td>
</tr>
<tr>
<td>2.1.3 Networks in the CNDC Model.</td>
<td>2-2</td>
</tr>
<tr>
<td>2.1.4 The Least Cost Route Problem.</td>
<td>2-3</td>
</tr>
<tr>
<td>2.1.5 The Least Cost Network Problem.</td>
<td>2-4</td>
</tr>
<tr>
<td>2.1.6 Network Optimization Algorithm</td>
<td>2-4</td>
</tr>
<tr>
<td>2.1.7 Determination of Least Cost Routes.</td>
<td>2-8</td>
</tr>
<tr>
<td>2.1.8 Determination of Least Cost Network</td>
<td>2-10</td>
</tr>
<tr>
<td>2.2 Optimization Techniques Used for Each Problem Type</td>
<td>2-12</td>
</tr>
<tr>
<td>2.3 Mixed SBS/Terrestrial Network Optimization.</td>
<td>2-13</td>
</tr>
<tr>
<td>2.3.1 Statement of the Problem.</td>
<td>2-13</td>
</tr>
<tr>
<td>2.3.2 Methodology</td>
<td>2-14</td>
</tr>
<tr>
<td>2.3.3 Detailed Discussion of Algorithm</td>
<td>2-16</td>
</tr>
<tr>
<td>2.3.3.1 NAC City Selection</td>
<td>2-16</td>
</tr>
<tr>
<td>2.3.3.2 Network Partitioning</td>
<td>2-21</td>
</tr>
<tr>
<td>2.3.3.3 Service Decision Making</td>
<td>2-22</td>
</tr>
<tr>
<td>2.3.3.4 Cost Response Tracking</td>
<td>2-27</td>
</tr>
<tr>
<td>3.0 CNDC Program Modules.</td>
<td>3-1</td>
</tr>
<tr>
<td>3.1.1 Executive Program</td>
<td>3-1</td>
</tr>
<tr>
<td>3.1.2 General Description</td>
<td>3-1</td>
</tr>
<tr>
<td>3.1.3 CNDC Module</td>
<td>3-5</td>
</tr>
<tr>
<td>3.1.4 COSTNG Modules.</td>
<td>3-7</td>
</tr>
<tr>
<td>3.1.5 NETWRK Modules.</td>
<td>3-9</td>
</tr>
<tr>
<td>3.2 SBSMIX Description.</td>
<td>3-11</td>
</tr>
<tr>
<td>3.2.1 SBSMIX Module</td>
<td>3-14</td>
</tr>
<tr>
<td>3.2.2 DEFTER Module</td>
<td>3-18</td>
</tr>
<tr>
<td>3.2.3 SRTNAC Module</td>
<td>3-20</td>
</tr>
<tr>
<td>3.2.4 PRTITN Module</td>
<td>3-23</td>
</tr>
<tr>
<td>3.2.5 SBSC Module</td>
<td>3-25</td>
</tr>
<tr>
<td>3.2.6 DECIDE Module</td>
<td>3-27</td>
</tr>
<tr>
<td>3.2.7 TOTCST Module</td>
<td>3-30</td>
</tr>
<tr>
<td>3.2.8 TRSTOT Module</td>
<td>3-34</td>
</tr>
<tr>
<td>3.2.9 ALDONE Module</td>
<td>3-36</td>
</tr>
<tr>
<td>3.2.10 ISNAC Module</td>
<td>3-38</td>
</tr>
<tr>
<td>3.2.11 SORTR Module</td>
<td>3-40</td>
</tr>
<tr>
<td>3.2.12 INDX Module</td>
<td>3-41</td>
</tr>
<tr>
<td>Section</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
</tr>
<tr>
<td>3.3</td>
<td>INPUT Description</td>
</tr>
<tr>
<td>3.3.1</td>
<td>INPUT Module</td>
</tr>
<tr>
<td>3.3.2</td>
<td>BSTRCT Module</td>
</tr>
<tr>
<td>3.3.3</td>
<td>EXCON Module</td>
</tr>
<tr>
<td>3.3.4</td>
<td>XCREATE Module</td>
</tr>
<tr>
<td>3.3.5</td>
<td>EXCREA Module</td>
</tr>
<tr>
<td>3.3.6</td>
<td>XLIST Module</td>
</tr>
<tr>
<td>3.4</td>
<td>NETCON Description</td>
</tr>
<tr>
<td>3.4.1</td>
<td>NETCON Module</td>
</tr>
<tr>
<td>3.4.2</td>
<td>NINIT Module</td>
</tr>
<tr>
<td>3.4.3</td>
<td>CREATE Module</td>
</tr>
<tr>
<td>3.4.4</td>
<td>GETTFC Module</td>
</tr>
<tr>
<td>3.4.5</td>
<td>GETPER Module</td>
</tr>
<tr>
<td>3.4.6</td>
<td>DEFINOD Module</td>
</tr>
<tr>
<td>3.4.7</td>
<td>NCREATE Module</td>
</tr>
<tr>
<td>3.4.8</td>
<td>DELETE Module</td>
</tr>
<tr>
<td>3.4.9</td>
<td>NDELETE Module</td>
</tr>
<tr>
<td>3.4.10</td>
<td>LIST Module</td>
</tr>
<tr>
<td>3.4.11</td>
<td>NLSTOR Module</td>
</tr>
<tr>
<td>3.4.12</td>
<td>NLSTFL Module</td>
</tr>
<tr>
<td>3.4.13</td>
<td>NCLOSE Module</td>
</tr>
<tr>
<td>3.5</td>
<td>TARCON Description</td>
</tr>
<tr>
<td>3.5.1</td>
<td>TARCON Module</td>
</tr>
<tr>
<td>3.5.2</td>
<td>TINIT Module</td>
</tr>
<tr>
<td>3.5.3</td>
<td>CREATATT Module</td>
</tr>
<tr>
<td>3.5.4</td>
<td>CRATT Module</td>
</tr>
<tr>
<td>3.5.5</td>
<td>CRWU Module</td>
</tr>
<tr>
<td>3.5.6</td>
<td>CRSBS Module</td>
</tr>
<tr>
<td>3.5.7</td>
<td>TCREAT Module</td>
</tr>
<tr>
<td>3.5.8</td>
<td>TCRATT Module</td>
</tr>
<tr>
<td>3.5.9</td>
<td>TCRWU Module</td>
</tr>
<tr>
<td>3.5.10</td>
<td>TCRSBS Module</td>
</tr>
<tr>
<td>3.5.11</td>
<td>DELETT Module</td>
</tr>
<tr>
<td>3.5.12</td>
<td>TDELET Module</td>
</tr>
<tr>
<td>3.5.13</td>
<td>LISTTT Module</td>
</tr>
<tr>
<td>3.5.14</td>
<td>TLSTDRT Module</td>
</tr>
<tr>
<td>3.5.15</td>
<td>TLSTFL Module</td>
</tr>
<tr>
<td>3.5.16</td>
<td>TLSTAT Module</td>
</tr>
<tr>
<td>3.5.17</td>
<td>TLSTWU Module</td>
</tr>
<tr>
<td>3.5.18</td>
<td>TLSTSB Module</td>
</tr>
<tr>
<td>3.5.19</td>
<td>TCLOSE Module</td>
</tr>
<tr>
<td>3.6</td>
<td>INIT Description</td>
</tr>
<tr>
<td>3.6.1</td>
<td>INIT Module</td>
</tr>
<tr>
<td>3.6.2</td>
<td>IBLNET Module</td>
</tr>
<tr>
<td>3.6.3</td>
<td>IBLNOD Module</td>
</tr>
<tr>
<td>3.6.4</td>
<td>IBLWHO Module</td>
</tr>
<tr>
<td>3.6.5</td>
<td>IBLLLST Module</td>
</tr>
<tr>
<td>Section</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>---------------------</td>
</tr>
<tr>
<td>3.6.6</td>
<td>IBLTRF Module</td>
</tr>
<tr>
<td>3.6.7</td>
<td>IBLTAR Module</td>
</tr>
<tr>
<td>3.6.8</td>
<td>IBLATT Module</td>
</tr>
<tr>
<td>3.6.9</td>
<td>IBLWU1 Module</td>
</tr>
<tr>
<td>3.6.10</td>
<td>IBLWU2 Module</td>
</tr>
<tr>
<td>3.6.11</td>
<td>IBLWU3 Module</td>
</tr>
<tr>
<td>3.6.12</td>
<td>IBLSB1 Module</td>
</tr>
<tr>
<td>3.6.13</td>
<td>IBLSBS Module</td>
</tr>
<tr>
<td>3.7</td>
<td>OUTPUT Description</td>
</tr>
<tr>
<td>3.7.1</td>
<td>OUTPUT Module</td>
</tr>
<tr>
<td>3.7.2</td>
<td>OUT1 Module</td>
</tr>
<tr>
<td>3.7.3</td>
<td>OUTATT Module</td>
</tr>
<tr>
<td>3.7.4</td>
<td>OUTMU Module</td>
</tr>
<tr>
<td>3.7.5</td>
<td>OUTSBS Module</td>
</tr>
<tr>
<td>3.7.6</td>
<td>OUT2 Module</td>
</tr>
<tr>
<td>3.7.7</td>
<td>OUT2SB Module</td>
</tr>
<tr>
<td>3.7.8</td>
<td>PRTSBS Module</td>
</tr>
<tr>
<td>3.7.9</td>
<td>GRETRE Module</td>
</tr>
<tr>
<td>3.7.10</td>
<td>OUT3 Module</td>
</tr>
<tr>
<td>3.7.11</td>
<td>OUT3SB Module</td>
</tr>
<tr>
<td>3.7.12</td>
<td>OUT4 Module</td>
</tr>
<tr>
<td>3.7.13</td>
<td>SUMEQP Module</td>
</tr>
<tr>
<td>3.8</td>
<td>Utility Functions</td>
</tr>
<tr>
<td>3.8.1</td>
<td>ACC Module</td>
</tr>
<tr>
<td>3.8.2</td>
<td>ATCOST Module</td>
</tr>
<tr>
<td>3.8.3</td>
<td>ATTST Module</td>
</tr>
<tr>
<td>3.8.4</td>
<td>CATEG Module</td>
</tr>
<tr>
<td>3.8.5</td>
<td>DEFPRE Module</td>
</tr>
<tr>
<td>3.8.6</td>
<td>DIRGEN Module</td>
</tr>
<tr>
<td>3.8.7</td>
<td>FACSIIZ Module</td>
</tr>
<tr>
<td>3.8.8</td>
<td>FILDEF Module</td>
</tr>
<tr>
<td>3.8.9</td>
<td>INTHCR Module</td>
</tr>
<tr>
<td>3.8.10</td>
<td>LISTED Module</td>
</tr>
<tr>
<td>3.8.11</td>
<td>MILES Module</td>
</tr>
<tr>
<td>3.8.12</td>
<td>MNMPTR Module</td>
</tr>
<tr>
<td>3.8.13</td>
<td>MTPLY Module</td>
</tr>
<tr>
<td>3.8.14</td>
<td>MTPY Module</td>
</tr>
<tr>
<td>3.8.15</td>
<td>PTRPRS Module</td>
</tr>
<tr>
<td>3.8.16</td>
<td>ROWNOT Module</td>
</tr>
<tr>
<td>3.8.17</td>
<td>RVCF Module</td>
</tr>
<tr>
<td>3.8.18</td>
<td>SBCOST Module</td>
</tr>
<tr>
<td>3.8.19</td>
<td>SMSA Module</td>
</tr>
<tr>
<td>3.8.20</td>
<td>SORT1 Module</td>
</tr>
<tr>
<td>3.8.21</td>
<td>SORT1D Module</td>
</tr>
<tr>
<td>3.8.22</td>
<td>SORT2 Module</td>
</tr>
<tr>
<td>3.8.23</td>
<td>TARTYP Module</td>
</tr>
<tr>
<td>3.8.24</td>
<td>TRFSKP Module</td>
</tr>
<tr>
<td>3.8.25</td>
<td>USRNET Module</td>
</tr>
<tr>
<td>3.8.26</td>
<td>WRTTRF Module</td>
</tr>
<tr>
<td>Section</td>
<td>Page</td>
</tr>
<tr>
<td>-------------</td>
<td>------</td>
</tr>
<tr>
<td>3.8.27</td>
<td></td>
</tr>
<tr>
<td>WUCOST Module</td>
<td>3-239</td>
</tr>
<tr>
<td>3.8.28</td>
<td></td>
</tr>
<tr>
<td>WUPAIR Module</td>
<td>3-240</td>
</tr>
<tr>
<td>3.9</td>
<td></td>
</tr>
<tr>
<td>BATCH Description</td>
<td>3-241</td>
</tr>
<tr>
<td>3.9.1</td>
<td></td>
</tr>
<tr>
<td>CNDCB Module</td>
<td>3-244</td>
</tr>
<tr>
<td>3.9.2</td>
<td></td>
</tr>
<tr>
<td>INPUTB Module</td>
<td>3-246</td>
</tr>
<tr>
<td>3.9.3</td>
<td></td>
</tr>
<tr>
<td>NETB Module</td>
<td>3-248</td>
</tr>
<tr>
<td>3.9.4</td>
<td></td>
</tr>
<tr>
<td>TARB Module</td>
<td>3-250</td>
</tr>
<tr>
<td>3.9.5</td>
<td></td>
</tr>
<tr>
<td>CRATTB Module</td>
<td>3-251</td>
</tr>
<tr>
<td>3.9.6</td>
<td></td>
</tr>
<tr>
<td>CRWUB Module</td>
<td>3-252</td>
</tr>
<tr>
<td>3.9.7</td>
<td></td>
</tr>
<tr>
<td>CRSBSB Module</td>
<td>3-254</td>
</tr>
<tr>
<td>3.9.8</td>
<td></td>
</tr>
<tr>
<td>EXCONB Module</td>
<td>3-255</td>
</tr>
<tr>
<td>4.0</td>
<td></td>
</tr>
<tr>
<td>CNDC Program Data Structures</td>
<td>4-1</td>
</tr>
<tr>
<td>5.0</td>
<td></td>
</tr>
<tr>
<td>CNDC Program Data Base</td>
<td>5-1</td>
</tr>
<tr>
<td>5.1</td>
<td></td>
</tr>
<tr>
<td>Introduction</td>
<td>5-1</td>
</tr>
<tr>
<td>5.2</td>
<td></td>
</tr>
<tr>
<td>Prestored Data Base</td>
<td>5-3</td>
</tr>
<tr>
<td>5.2.1</td>
<td></td>
</tr>
<tr>
<td>Network</td>
<td>5-3</td>
</tr>
<tr>
<td>5.2.2</td>
<td></td>
</tr>
<tr>
<td>Traffic Table</td>
<td>5-4</td>
</tr>
<tr>
<td>5.2.3</td>
<td></td>
</tr>
<tr>
<td>Traffic by Volume</td>
<td>5-5</td>
</tr>
<tr>
<td>5.2.4</td>
<td></td>
</tr>
<tr>
<td>ATT Tariff.</td>
<td>5-5</td>
</tr>
<tr>
<td>5.2.5</td>
<td></td>
</tr>
<tr>
<td>WU Tariff.</td>
<td>5-7</td>
</tr>
<tr>
<td>5.2.6</td>
<td></td>
</tr>
<tr>
<td>SBS Tariff.</td>
<td>5-8</td>
</tr>
<tr>
<td>5.2.7</td>
<td></td>
</tr>
<tr>
<td>User Defined Tariff Directory</td>
<td>5-9</td>
</tr>
<tr>
<td>5.2.8</td>
<td></td>
</tr>
<tr>
<td>User Defined Network Directory</td>
<td>5-10</td>
</tr>
<tr>
<td>5.3</td>
<td></td>
</tr>
<tr>
<td>User Defined Data Base</td>
<td>5-11</td>
</tr>
<tr>
<td>5.3.1</td>
<td></td>
</tr>
<tr>
<td>User Defined Networks</td>
<td>5-11</td>
</tr>
<tr>
<td>5.3.2</td>
<td></td>
</tr>
<tr>
<td>User Defined Tariffs</td>
<td>5-12</td>
</tr>
<tr>
<td>5.3.2.1</td>
<td></td>
</tr>
<tr>
<td>NASA Defined ATT Tariff</td>
<td>5-12</td>
</tr>
<tr>
<td>5.3.2.2</td>
<td></td>
</tr>
<tr>
<td>NASA Defined Western Union Tariff</td>
<td>5-14</td>
</tr>
<tr>
<td>5.3.2.3</td>
<td></td>
</tr>
<tr>
<td>NASA Defined SBS Tariff</td>
<td>5-15</td>
</tr>
<tr>
<td>5.3.3</td>
<td></td>
</tr>
<tr>
<td>Execution Control File</td>
<td>5-17</td>
</tr>
<tr>
<td>Appendix</td>
<td></td>
</tr>
<tr>
<td>A</td>
<td></td>
</tr>
<tr>
<td>Module Cross Reference</td>
<td>A-1</td>
</tr>
<tr>
<td>B</td>
<td></td>
</tr>
<tr>
<td>Error Numbers and Messages</td>
<td>B-1</td>
</tr>
<tr>
<td>C</td>
<td></td>
</tr>
<tr>
<td>Prestored ATT Tariff</td>
<td>C-1</td>
</tr>
<tr>
<td>D</td>
<td></td>
</tr>
<tr>
<td>Prestored WU Tariff</td>
<td>D-1</td>
</tr>
<tr>
<td>E</td>
<td></td>
</tr>
<tr>
<td>Prestored SBS Tariff</td>
<td>E-1</td>
</tr>
<tr>
<td>Appendix</td>
<td>Page</td>
</tr>
<tr>
<td>----------</td>
<td>------</td>
</tr>
<tr>
<td>F</td>
<td>F-1</td>
</tr>
<tr>
<td>G</td>
<td>G-1</td>
</tr>
<tr>
<td>H</td>
<td>H-1</td>
</tr>
<tr>
<td>I</td>
<td>I-1</td>
</tr>
<tr>
<td>J</td>
<td>J-1</td>
</tr>
<tr>
<td>K</td>
<td>K-1</td>
</tr>
</tbody>
</table>

Addendum
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2-1</td>
<td>A Tree with Root at Node 1.</td>
<td>2-3</td>
</tr>
<tr>
<td>2-2</td>
<td>A Sample Network.</td>
<td>2-6</td>
</tr>
<tr>
<td>2-3</td>
<td>$M^0$ and $O^0$ Matrices of Sample Network.</td>
<td>2-6</td>
</tr>
<tr>
<td>2-4</td>
<td>Least Cost Network.</td>
<td>2-7</td>
</tr>
<tr>
<td>2-5</td>
<td>$M^N$ and $O^N$ Matrices of Sample Network.</td>
<td>2-9</td>
</tr>
<tr>
<td>2-6</td>
<td>Traffic Volumes Along Least Cost Route Contributing to Link Within Route.</td>
<td>2-10</td>
</tr>
<tr>
<td>2-7</td>
<td>A Least Cost Network.</td>
<td>2-11</td>
</tr>
<tr>
<td>2-8</td>
<td>Baseline SBS Cost Function (Actual Rates)</td>
<td>2-17</td>
</tr>
<tr>
<td>2-9</td>
<td>Sample Network, Traffic and Cost Tables.</td>
<td>2-20</td>
</tr>
<tr>
<td>2-10</td>
<td>Example of a Crossover Cost</td>
<td>2-23</td>
</tr>
<tr>
<td>2-11</td>
<td>An Ordering of Crossover Costs For Any Partition.</td>
<td>2-24</td>
</tr>
<tr>
<td>2-12</td>
<td>Grouping of Partition Links with Initial Satellite Cost.</td>
<td>2-25</td>
</tr>
<tr>
<td>2-13</td>
<td>Expected Total Network Cost Function.</td>
<td>2-28</td>
</tr>
<tr>
<td>3-1</td>
<td>CNDC Hierarchy Chart.</td>
<td>3-2</td>
</tr>
<tr>
<td>3-2</td>
<td>NASA CNDC Model Program Flow.</td>
<td>3-6</td>
</tr>
<tr>
<td>3-3</td>
<td>COSTNG Logic Flow.</td>
<td>3-8</td>
</tr>
<tr>
<td>3-4</td>
<td>NETWRK Logic Flow.</td>
<td>3-10</td>
</tr>
<tr>
<td>3-5</td>
<td>SBSMIX Hierarchy Chart.</td>
<td>3-13</td>
</tr>
<tr>
<td>3-6</td>
<td>SBSMIX Logic Flow.</td>
<td>3-15</td>
</tr>
<tr>
<td>3-7</td>
<td>DEFTER Logic Flow.</td>
<td>3-19</td>
</tr>
<tr>
<td>3-8</td>
<td>SRTNAC Logic Flow.</td>
<td>3-22</td>
</tr>
<tr>
<td>3-9</td>
<td>PRTITN Logic Flow.</td>
<td>3-24</td>
</tr>
<tr>
<td>3-10</td>
<td>SBSC Logic Flow.</td>
<td>3-26</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>-----------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>3-11</td>
<td>DECIDE Logic Flow</td>
<td>3-28</td>
</tr>
<tr>
<td>3-12</td>
<td>TOTCST Logic Flow</td>
<td>3-32</td>
</tr>
<tr>
<td>3-13</td>
<td>TRSTOT Logic Flow</td>
<td>3-35</td>
</tr>
<tr>
<td>3-14</td>
<td>ALDONE Logic Flow</td>
<td>3-37</td>
</tr>
<tr>
<td>3-15</td>
<td>ISNAC Logic Flow</td>
<td>3-39</td>
</tr>
<tr>
<td>3-16</td>
<td>INPUT Hierarchy Chart</td>
<td>3-44</td>
</tr>
<tr>
<td>3-17</td>
<td>INPUT Logic Flow</td>
<td>3-46</td>
</tr>
<tr>
<td>3-18</td>
<td>BSTRCT Logic Flow</td>
<td>3-49</td>
</tr>
<tr>
<td>3-19</td>
<td>EXCON Logic Flow</td>
<td>3-51</td>
</tr>
<tr>
<td>3-20</td>
<td>XCREAT Logic Flow</td>
<td>3-54</td>
</tr>
<tr>
<td>3-21</td>
<td>EXCREA Logic Flow</td>
<td>3-57</td>
</tr>
<tr>
<td>3-22</td>
<td>XLIST Logic Flow</td>
<td>3-59</td>
</tr>
<tr>
<td>3-23</td>
<td>NETCON Hierarchy Chart</td>
<td>3-62</td>
</tr>
<tr>
<td>3-24</td>
<td>NETCON Logic Flow</td>
<td>3-64</td>
</tr>
<tr>
<td>3-25</td>
<td>NINIT Logic Flow</td>
<td>3-66</td>
</tr>
<tr>
<td>3-26</td>
<td>CREATE Logic Flow</td>
<td>3-69</td>
</tr>
<tr>
<td>3-27</td>
<td>GETTFC Logic Flow</td>
<td>3-73</td>
</tr>
<tr>
<td>3-28</td>
<td>GETPER Logic Flow</td>
<td>3-75</td>
</tr>
<tr>
<td>3-29</td>
<td>DEFNOD Logic Flow</td>
<td>3-78</td>
</tr>
<tr>
<td>3-30</td>
<td>NCREAT Logic Flow</td>
<td>3-81</td>
</tr>
<tr>
<td>3-31</td>
<td>DELETE Logic Flow</td>
<td>3-83</td>
</tr>
<tr>
<td>3-32</td>
<td>NDELETE Logic Flow</td>
<td>3-85</td>
</tr>
<tr>
<td>3-33</td>
<td>LIST Logic Flow</td>
<td>3-87</td>
</tr>
<tr>
<td>3-34</td>
<td>NLSTDR Logic Flow</td>
<td>3-89</td>
</tr>
<tr>
<td>3-35</td>
<td>NLSTFL Logic Flow</td>
<td>3-91</td>
</tr>
<tr>
<td>3-36</td>
<td>NCLOSE Logic Flow</td>
<td>3-93</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>--------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>3-37</td>
<td>TARCON Hierarchy Chart</td>
<td>3-95</td>
</tr>
<tr>
<td>3-38</td>
<td>TARCON Logic Flow</td>
<td>3-97</td>
</tr>
<tr>
<td>3-39</td>
<td>TINIT Logic Flow</td>
<td>3-99</td>
</tr>
<tr>
<td>3-40</td>
<td>CREAT Logic Flow</td>
<td>3-101</td>
</tr>
<tr>
<td>3-41</td>
<td>CRATT Logic Flow</td>
<td>3-103</td>
</tr>
<tr>
<td>3-42</td>
<td>CRWU Logic Flow</td>
<td>3-105</td>
</tr>
<tr>
<td>3-43</td>
<td>CRSBS Logic Flow</td>
<td>3-107</td>
</tr>
<tr>
<td>3-44</td>
<td>TCREAT Logic Flow</td>
<td>3-109</td>
</tr>
<tr>
<td>3-45</td>
<td>TCRATT Logic Flow</td>
<td>3-111</td>
</tr>
<tr>
<td>3-46</td>
<td>TCRWU Logic Flow</td>
<td>3-113</td>
</tr>
<tr>
<td>3-47</td>
<td>TCRSBS Logic Flow</td>
<td>3-115</td>
</tr>
<tr>
<td>3-48</td>
<td>DELETT Logic Flow</td>
<td>3-117</td>
</tr>
<tr>
<td>3-49</td>
<td>TDELET Logic Flow</td>
<td>3-119</td>
</tr>
<tr>
<td>3-50</td>
<td>LISTT Logic Flow</td>
<td>3-121</td>
</tr>
<tr>
<td>3-51</td>
<td>TLSTDR Logic Flow</td>
<td>3-123</td>
</tr>
<tr>
<td>3-52</td>
<td>TLSTFL Logic Flow</td>
<td>3-125</td>
</tr>
<tr>
<td>3-53</td>
<td>TLSTAT Logic Flow</td>
<td>3-127</td>
</tr>
<tr>
<td>3-54</td>
<td>TLSTWU Logic Flow</td>
<td>3-129</td>
</tr>
<tr>
<td>3-55</td>
<td>TLSTSB Logic Flow</td>
<td>3-131</td>
</tr>
<tr>
<td>3-56</td>
<td>TCLOSE Logic Flow</td>
<td>3-133</td>
</tr>
<tr>
<td>3-57</td>
<td>INIT Hierarchy Chart</td>
<td>3-135</td>
</tr>
<tr>
<td>3-58</td>
<td>INIT Logic Flow</td>
<td>3-138</td>
</tr>
<tr>
<td>3-59</td>
<td>INBLNET Logic Flow</td>
<td>3-141</td>
</tr>
<tr>
<td>3-60</td>
<td>IBLNOD Logic Flow</td>
<td>3-144</td>
</tr>
<tr>
<td>3-61</td>
<td>IBLWHO Logic Flow</td>
<td>3-146</td>
</tr>
<tr>
<td>3-62</td>
<td>IBLLST Logic Flow</td>
<td>3-149</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>------------------------------------</td>
<td>--------</td>
</tr>
<tr>
<td>3-63</td>
<td>IBLTRF Logic Flow</td>
<td>3-152</td>
</tr>
<tr>
<td>3-64</td>
<td>IBLTAR Logic Flow</td>
<td>3-155</td>
</tr>
<tr>
<td>3-65</td>
<td>IBLATT Logic Flow</td>
<td>3-159</td>
</tr>
<tr>
<td>3-66</td>
<td>IBLWU1 Logic Flow</td>
<td>3-161</td>
</tr>
<tr>
<td>3-67</td>
<td>IBLWU2 Logic Flow</td>
<td>3-163</td>
</tr>
<tr>
<td>3-68</td>
<td>IBLWU3 Logic Flow</td>
<td>3-166</td>
</tr>
<tr>
<td>3-69</td>
<td>IBLSB1 Logic Flow</td>
<td>3-168</td>
</tr>
<tr>
<td>3-70</td>
<td>IBLSBS Logic Flow</td>
<td>3-171</td>
</tr>
<tr>
<td>3-71</td>
<td>OUTPUT Hierarchy Chart.</td>
<td>3-176</td>
</tr>
<tr>
<td>3-72</td>
<td>OUTPUT Logic Flow</td>
<td>3-178</td>
</tr>
<tr>
<td>3-73</td>
<td>OUT1 Logic Flow</td>
<td>3-180</td>
</tr>
<tr>
<td>3-74</td>
<td>OUTATT Logic Flow</td>
<td>3-182</td>
</tr>
<tr>
<td>3-75</td>
<td>OUTWU Logic Flow</td>
<td>3-184</td>
</tr>
<tr>
<td>3-76</td>
<td>OUTSBS Logic Flow</td>
<td>3-186</td>
</tr>
<tr>
<td>3-77</td>
<td>OUT2 Logic Flow</td>
<td>3-189</td>
</tr>
<tr>
<td>3-78</td>
<td>OUT2SB Logic Flow</td>
<td>3-193</td>
</tr>
<tr>
<td>3-79</td>
<td>PRTSBS Logic Flow</td>
<td>3-197</td>
</tr>
<tr>
<td>3-80</td>
<td>GETRTE Logic Flow</td>
<td>3-199</td>
</tr>
<tr>
<td>3-81</td>
<td>OUT3 Logic Flow</td>
<td>3-202</td>
</tr>
<tr>
<td>3-82</td>
<td>OUT3SB Logic Flow</td>
<td>3-205</td>
</tr>
<tr>
<td>3-83</td>
<td>OUT4 Logic Flow</td>
<td>3-208</td>
</tr>
<tr>
<td>3-84</td>
<td>SUMEQP Logic Flow</td>
<td>3-210</td>
</tr>
<tr>
<td>3-85</td>
<td>BATCH Hierarchy Chart</td>
<td>3-243</td>
</tr>
<tr>
<td>3-86</td>
<td>CNDCB Logic Flow</td>
<td>3-245</td>
</tr>
<tr>
<td>3-87</td>
<td>INPUTB Logic Flow</td>
<td>3-247</td>
</tr>
<tr>
<td>5-1</td>
<td>Communications Network Design and Costing Model</td>
<td>5-2</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>--------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>I-1</td>
<td>Network Description</td>
<td>I-2</td>
</tr>
<tr>
<td>I-2</td>
<td>ATT Tariff Description</td>
<td>I-3</td>
</tr>
<tr>
<td>I-3</td>
<td>WU Tariff Description</td>
<td>I-4</td>
</tr>
<tr>
<td>I-4</td>
<td>SBS Tariff Description</td>
<td>I-5</td>
</tr>
<tr>
<td>I-5</td>
<td>SBS Least Cost Route</td>
<td>I-6</td>
</tr>
<tr>
<td>I-6</td>
<td>Least Cost Route</td>
<td>I-7</td>
</tr>
<tr>
<td>I-7</td>
<td>Least Cost Network</td>
<td>I-8</td>
</tr>
<tr>
<td>I-8</td>
<td>Output Network Totals</td>
<td>I-9</td>
</tr>
<tr>
<td>I-9</td>
<td>Tariff Summary</td>
<td>I-10</td>
</tr>
<tr>
<td>I-10</td>
<td>Traffic Table</td>
<td>I-11</td>
</tr>
</tbody>
</table>
1.0 INTRODUCTION

This report contains a detailed description of the NASA Communications Network Design and Costing (CNDC) Model from the programmer's point of view. As such, it contains descriptions of all system program modules, the data structures within the model, and the files which make up the data base.

The CNDC Model is a computer program which may be used to model and analyze communication trunking networks. The model provides full connectivity among as many as 600 service nodes using either terrestrial systems, satellite systems or a combination of terrestrial and satellite systems for voice services. The program determines the traffic routing and associated costs which provide the least cost over the tariffs and network specified by the user. The user may specify for the terrestrial tariff the American Telephone and Telegraph Tariff (as described in FCC #260, #264) or for the satellite tariff either that of the Western Union Telegraph Company (as described in FCC #261) or the Satellite Business System Incorporated (as described in FCC #2). In addition, he may specify tariffs postulated by NASA which have the same connectivity philosophy of the above mentioned FCC approved tariffs. The output from the model enables a direct determination of the least cost route network based upon the lowest tariffed costs per set of input traffic assumptions.

The CNDC program has two main functions:

1) allow the user to define networks and tariffs

2) determine the traffic routing and costs which provide the least cost over the network and tariffs specified.

The optimization algorithm selected (described in Section 2) solves the least cost route problem for terrestrial only, satellite only, and mixed terrestrial and Western Union like tariff cases. However, the mixed terrestrial-SBS like tariff problem cannot be solved with this algorithm since the two tariff types involved are defined by very different philosophies. The terrestrial like tariff is distance sensitive while the SBS like tariff is
volume sensitive. As a result, a separate optimization algorithm was defined for the terrestrial-SBS mixed problem. (Section 2.3)

The CNDC program was implemented on the Lewis Research Center IBM 3033AP Computer system running under the TSS/370 operating system. CNDC was written in TSS/370 FORTRAN IV. The program requires computer operator involvement only if the user chooses to operate in batch mode using a card deck. However, since batch mode can be invoked from the terminal or the user may operate in interactive mode, operator involvement can be avoided.

Any changes to the tariffs are set forth in the Addendum.
2.0 OPTIMIZATION ALGORITHMS

2.1 NETWORK OPTIMIZATION TECHNIQUES

2.1.1 Introduction

The CNDC model provides solutions to two different network problems. The first problem deals with the least-cost routes connecting unique pairs of network nodes exclusive of other network traffic. The second problem considers the entire network to find the set of links connecting the nodes so that the overall network cost is a minimum. This section provides definitions of the two network optimization problems, discusses the optimization algorithm implemented in the CNDC, and explains the method used to provide solutions to each of the two problems.

2.1.2 Network Terminology

A network (or graph) consists of a finite set of nodes (or vertices) and a finite set of links (or arcs) connecting pairs of nodes. The network nodes are assigned numbers from 1 to N where N is the number of nodes. The arcs of the network are described by ordered pairs of vertices. The first element in the ordered pair is the number of the originating node for the link and the second element of the ordered pair indicates the node at which the link terminates. The ordered pair (i, j) denotes the arc that connects node i to node j (1 ≤ i, j ≤ N). Values of a measure such as cost, distance or energy level are generally assigned to arcs of the network. Network optimization techniques seek to maximize or minimize with respect to the measure used.

A network is said to be directed if the value associated with arc (i, j) is in general not equal to the value associated with arc (j, i). A path is a finite sequence of arcs connecting two nodes. The terminating vertex of each arc in the path except the last is the originating vertex of the next arc in the path. For example, the set of arcs (x, y), (y, z), (z, w), (w, t) is a path...
from node x to node t. A path can be described by listing the nodes it includes. The above example is then determined by the sequence of nodes x,y, z,w,t. The path is denoted x-y-z-w-t.

Network optimization algorithms generally assume that link values are additive. This assumption is necessary for the comparison of alternative paths. The value of the link from node x to node y can be compared to the sum of values of the links through a third node. The shortest-path connecting two nodes, then, is a path between the two nodes such that the sum of the values of the links comprising the path is a minimum. The algorithms thus assume that the addition of link values is meaningful. If link values represent distance, for example, the total distance covered in traversing a path is equal to the sum of the lengths of the links. The measure of interest must be transformed so as to be additive or otherwise expressed in a way that satisfies this assumption.

2.1.3 Networks in the CNDC Model

In the communications networks modeled for the CNDC program, up to 600 rate centers define the network nodes. Directed links are assumed to connect all ordered pairs of nodes. The links represent single voice circuits providing voice communications between rate centers. The criteria being optimized is cost. In order to satisfy the additivity assumption of the network algorithm, link costs must be expressed as costs per circuit. The cost to run a single voice circuit from node x to node y to node z is then equal to the sum of the cost of a circuit connecting node x to node y plus the cost of a circuit from node y to node z. When the link costs are a function of traffic volumes as in the case of the SBS tariff, the cost is calculated based on the appropriate traffic volume and then expressed as a cost per circuit. After the execution of the network algorithm, the number of voice circuits required on each link is multiplied by the cost of a single voice circuit on the link to determine total link costs.

The cost per circuit associated with each link represents a minimum over all tariffs in a given run. This fact is transparent to the optimization
algorithm. Except for the SBS mixed problem, costs are minimized over all tariffs prior to network optimization. (The SBS mixed problem will be discussed in section 2.3.) The tariffs supplying the minimum costs are stored separately. Although the overall CNDC Model may consider multiple services on all links, the network that is input to the optimization algorithm has at most a single direct link between each ordered pair of nodes.

2.1.4 The Least Cost Routing Problem

Research in the field of shortest-path methods has resulted in the development of several efficient computer codes (see references 5, 7 in Appendix G). These codes solve for the shortest path tree originating from a given root node. The following is an example of a tree rooted at node 1.

![Figure 2-1. A Tree with Root at Node 1](image)

All of the paths in the tree are directed outward from the root. All nodes in the network are included in the tree. Note also that each node has a unique predecessor node in the tree. Each node is the terminating node of exactly one arc in the tree. The corresponding predecessor node is the originating node of that arc. The least-cost tree for a given root defines the least-cost paths from the root to all other nodes in the network.

The CNDC Model does not implement any of the above-mentioned codes. If these codes are repeated N times with each node as the root, the union of the resulting least-cost trees is the least-cost network. The CNDC Model implements a network algorithm that maintains information about the least-cost
routes while it solves the least-cost network problem. The results of the network algorithm are used by submodules of the OUTPUT module to generate information about both (1) the least-cost routes between all pairs of nodes in the network and (2) the least-cost network.

2.1.5 The Least-Cost Network Problem

In contrast to the activity surrounding current research to improve on the implementation of shortest-path methods, the problem of finding the least-cost network has received very little attention. The solutions that have been found are definitive and there is good reason to believe that the number of computer operations that must be performed cannot be reduced further (see reference 4 in Appendix G). The implementation used in the CNDC Model is based on an algorithm of Floyd (see reference 12 in Appendix G). The code has been expanded to maintain information about the least-cost routes as the model executes.

The least-cost network problem involves finding the set of links that provides connectivity between all pairs of nodes such that the total network cost is a minimum. The least-cost network includes the least-cost routes connecting all pairs of nodes in the network.

2.1.6 The Network Optimization Algorithm

The algorithm used by the CNDC model to find the least-cost network (and least-cost routes) is based on an algorithm of Floyd (see reference 12 in Appendix G). The algorithm defines paths between all pairs of nodes. The paths are initialized to single direct links between the nodes. If a given link does not exist, it is assigned an infinite cost. The algorithm considers all network nodes for inclusion in all paths.Whenever the inclusion of a node reduces the cost of a path, the path is rerouted through that node. If no combination of links results in a lower cost than the direct link, it will never be replaced.
The network nodes are numbered from 1 to N. The minimum cost of links between each pair of nodes are stored in an array, call it $M^0$. The $i,j^{th}$ element of the array represents the cost associated with the optimal link from node $i$ to node $j$. If no link exists between any pair of nodes, the corresponding cost is infinite. The diagonal elements in the array are all zero since there is no cost to connect a node to itself. The optimization algorithm constructs $N$ additional arrays where each array is determined from the entries in the preceding array. Let $C_{ij}$ represent the entry in row $i$ and column $j$ of the array $M^n$ constructed at the $n^{th}$ iteration ($1 \leq i, j \leq N$, $2 \leq n < N$). Then the recursion equation used to determine the entry in cell $i, j$ of the $(n+1)^{st}$ array, $M^{n+1}$ is

$$C_{ij} = \min \{ C_{in} + C_{nj}, C_{ij} \}.$$ 

This equation is applied for all pairs $i, j$ where neither $i$ nor $j$ is equal to $n$. It can be interpreted as testing whether routing the traffic from node 1 to node $j$ via node $n$ reduces the associated cost per circuit. If so, node $n$ is included in the path from node $i$ to node $j$. A second array, $O^n$ is constructed at each iteration to record the second node in each path.

An intermediate node in a path is any node in the path except the initial or terminal node. The $n^{th}$ array, $M^n$, contains costs to connect all pairs of nodes where only the first $n$ nodes appear in any path. The $(n+1)^{st}$ array $M^{n+1}$ is constructed by considering the $(n+1)^{st}$ node for inclusion in each path. Upon termination, the entry in row $i$ and column $j$ of the final array $M^n$ represents the cost associated with the least-cost path from node $i$ to node $j$. The assurance that the final cost is a minimum follows inductively from the fact that at the $n^{th}$ iteration, the least-cost path from node $i$ to node $j$ that includes only intermediate nodes with numbers from 1 to $n$ must be the smaller of (1) the cost of the least-cost path from node $i$ to node $j$ that includes only intermediate nodes with numbers from 1 to $n-1$ and (2) the cost of the least-cost path that includes only the intermediate nodes with numbers from 1 to $n$ with node $n$ appearing once as an intermediate vertex.
Consider the four node directed network shown in figure 2-2.

![Figure 2-2. A Sample Network](image)

The costs of direct links are indicated by the numbers attached to the links. The network contains direct links between all pairs of nodes. The initial arrays are:

\[
\begin{array}{cccc}
1 & 2 & 3 & 4 \\
1 & 0 & 4 & 2 & 8 \\
2 & 3 & 0 & 1 & 4 \\
3 & 8 & 9 & 0 & 4 \\
4 & 2 & 6 & 9 & 0 \\
\end{array}
\]

\[
\begin{array}{cccc}
1 & 2 & 3 & 4 \\
1 & 1 & 2 & 3 & 4 \\
2 & 1 & 2 & 3 & 4 \\
3 & 1 & 2 & 3 & 4 \\
4 & 1 & 2 & 3 & 4 \\
\end{array}
\]

![Figure 2-3. M° and 0° Matrices of Sample Network](image)
Row indices correspond to the numbers of originating nodes and column indices correspond to the numbers of terminating nodes. Cell i,j of the $M^n$ array indicates the cost per circuit of traffic from node i to node j. Cell i,j of the $O^n$ array indicates the second node in the path from node 1 to node j.

$M^1$ is constructed from $M^n$ as follows:

$M^1_{(2,3)} = C_{123} = \min \{ C_{21} + C_{13}, C_{23} \} = \min \{ 3+2, 1 \} = 1 \quad \text{Path Selected} \quad \text{2-3}$

$M^1_{(2,4)} = C_{124} = \min \{ C_{21} + C_{14}, C_{24} \} = \min \{ 3+8, 4 \} = 4 \quad \text{2-4}$

$M^1_{(3,2)} = C_{132} = \min \{ C_{31} + C_{12}, C_{32} \} = \min \{ 8+4, 9 \} = 9 \quad \text{3-2}$

$M^1_{(3,4)} = C_{134} = \min \{ C_{31} + C_{14}, C_{34} \} = \min \{ 8+8, 4 \} = 4 \quad \text{3-4}$

$M^1_{(4,2)} = C_{142} = \min \{ C_{41} + C_{12}, C_{42} \} = \min \{ 2+4, 6 \} = 6 \quad \text{4-2}$

$M^1_{(4,3)} = C_{143} = \min \{ C_{41} + C_{13}, C_{43} \} = \min \{ 2+2, 9 \} = 4 \quad \text{4-1-3}$

$C_{11} = C_{22} = C_{33} = C_{44} = 0$ on all iterations and $C_{n in} = C_{n-1 in}$ and $C_{n-1 ni} = C_{n-1 ni}$ on all iterations.

Entries in row and column 1 retain their previous values. The originating and terminating nodes of a path need not be considered for inclusion as intermediate nodes in the path. $M^2$, $M^3$, $M^4$, $O^2$, $O^3$, and $O^4$ are calculated in a similar manner. Figure 2.5 contains the resulting tables. The least cost network defined by $M^4$ and $O^4$ is represented in Figure 2-4.

![Figure 2-4. Least Cost Network](image-url)
The paths connecting pairs of nodes in the least-cost network are traced from the 0 array. The second node on the path from node 1 to node 4 is node 3 since \(0(1,4) = 3\). The second node on the path from node 3 to node 4 is node 4 since \(0(3,4) = 4\). The path from node 1 to node 4 is 1-3-4.

The paths connecting all pairs of network nodes and the associated costs are as follows:

<table>
<thead>
<tr>
<th>Least-Cost Path</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>node 1 to node 2</td>
<td>1-2</td>
</tr>
<tr>
<td>node 1 to node 3</td>
<td>1-3</td>
</tr>
<tr>
<td>node 1 to node 4</td>
<td>1-3-4</td>
</tr>
<tr>
<td>node 2 to node 1</td>
<td>2-1</td>
</tr>
<tr>
<td>node 2 to node 3</td>
<td>2-3</td>
</tr>
<tr>
<td>node 2 to node 4</td>
<td>2-4</td>
</tr>
<tr>
<td>node 3 to node 1</td>
<td>3-4-1</td>
</tr>
<tr>
<td>node 3 to node 2</td>
<td>3-2</td>
</tr>
<tr>
<td>node 3 to node 4</td>
<td>3-4</td>
</tr>
<tr>
<td>node 4 to node 1</td>
<td>4-1</td>
</tr>
<tr>
<td>node 4 to node 2</td>
<td>4-2</td>
</tr>
<tr>
<td>node 4 to node 3</td>
<td>4-1-3</td>
</tr>
</tbody>
</table>

2.1.7 The Determination of Least-Cost Routes

The output of the network optimization algorithm includes a data structure that contains linked lists defining the least-cost paths between all pairs of nodes in the network (the 0 array in the example). In order to describe a given least-cost route, the traffic between all pairs of nodes in the route is added to the appropriate link. Consider for example, a least-cost route from node x to node y via nodes v and w (see figure 2-6).
Figure 2-5. $M^N$ and $O^N$ Matrices of Sample Network
The link between node v and w must be sized to accommodate the sum of the following traffic volumes:

- traffic from node x to node y
- traffic from node x to node w
- traffic from node v to node w
- traffic from node v to node y

The results of the network optimization algorithm are used by the CNDC Model to describe each least-cost route in the network. The appropriate traffic volumes traversing each link in the route are summed. The corresponding costs per circuit on each link are used to calculate the cost of each link. The link costs are summed to determine the total cost of circuits on the route.

2.1.8 Determination of the Least-Cost Network

The output of the network optimization algorithm defines a path in the least-cost network connecting each pair of network nodes. In order to describe the least-cost network, it is necessary to determine the total
network traffic traversing each link in the least-cost network. Consider the simple example described in figure 2-7. Suppose the least-cost network

![Figure 2-7. A Least Cost Network](image)

for the three nodes consists of the links as shown. To determine the total traffic on each link, the CNDC model traverses the least-cost path connecting each pair of network nodes, adding the traffic volume for the pair to the total volume for each link in the path. While all pairs of nodes have been considered, the link totals represent the total network traffic on the links. The pairs of nodes in the least-cost network in figure 2-7 are (1,2), (1,3), (2,1), (2,3), (3,1), and (3,2).

Denote the traffic volume from node x to node y by $T_{xy}$. Denote the total network traffic volume on link $(ij)$ by $T(i,j)$. Traversing the paths connecting each pair of nodes yields the link volumes

\[
\begin{align*}
T(1,2) &= T_{12} + T_{32} \\
T(1,3) &= T_{13} \\
T(2,3) &= T_{21} + T_{23} \\
T(3,1) &= T_{21} + T_{31} + T_{32}
\end{align*}
\]

The paths from node 1 to node 2 and node 1 to node 3 are direct so $T_{12}$ is added to $T(1,2)$ and $T_{13}$ is added to $T(1,3)$. The path from node 2 to node 1 is via node 3 so $T_{21}$ is added to $T(2,3)$ and $T(3,1)$. The path from node 2 to node 3 is direct so $T_{23}$ is only added to $T(2,3)$. The path from node 3 to node 1 is direct so $T_{31}$ is added to $T(3,1)$. The path from node 3 to node 2 is via node 1 so $T_{32}$ is added to $T(3,1)$ and $T(1,2)$.
The procedure outlined in the above example is executed by the CNDC for the least-cost network based on the outputs of the network optimization algorithm. The link volumes thus obtained are multiplied by the minimum cost per circuit associated with each link to determine the total cost of the link. The sum of the costs for each link is the overall network cost.

2.2 OPTIMIZATION TECHNIQUE USED FOR EACH PROBLEM TYPE

There are five problem types for which CNDC must determine the optimal solution:

1. All terrestrial tariffs
2. All Western Union like tariffs
3. All SBS like tariffs
4. Mixture of one terrestrial tariff and one Western Union like tariff
5. Mixture of one terrestrial tariff and one SBS like tariff

In problem types 1 through 4, the tariff providing the least cost service for each link is determined. For satellite only problems (problem types 2 and 3) optimization is complete at this point since all satellite routes must be direct, that is, double hopping is not permitted. When the problem is of type 1 or 4, there is an additional step of determining the least cost path between each pair of nodes in the network. The network optimization technique described in section 2.1 performs this function.

As indicated earlier, the SBS Mixed Problem cannot be solved using the optimization technique described above because the two tariffs are defined by very different philosophies. A discussion of the mixed SBS/terrestrial network optimization technique is included in section 2.3.
2.3 MIXED SBS/TERRESTRIAL NETWORK OPTIMIZATION

This section discusses the algorithm implemented in CNDC to determine the least-cost network solution for a mixed SBS/Terrestrial type network. While the proposed algorithm cannot guarantee an optimal solution, the assumptions made with the algorithm should produce a near minimum cost network. The determination of an optimal solution is hindered by a usage sensitive nonlinear SBS cost function with many local extreme points. A heuristic approach is developed to:

1) Place SBS facilities in locations with sufficient traffic to yield the lowest possible satellite costs per circuit within the network,

2) Partition the network around SBS satellite cities such that the intrapartition network is optimally configured, and

3) Determine which services (terrestrial or satellite) yield the least-cost interpartition links between all network cities.

2.3.1 Statement of the Problem

The solution of the general least-cost network problem has received substantial discussion in the literature. Algorithms have been presented which are efficient and fairly straightforward. All of these algorithms require a set of known and fixed link costs for their solution. Networks involving tariffs which are distance sensitive, such as AT&T and WU, can be analyzed quite readily with these general algorithms. This results from the fact that for a given set of cities specified as a network, the distance between any two cities is known and the associated link cost can be determined from a table look up. For other types of networks involving usage sensitive tariffs, such as SBS, the link costs are a variable function of the traffic volume transmitted over the link. In general, the cost per circuit is quantity discounted and decreases as the traffic volume increases. In order to determine the link costs for these types of tariffs, the traffic volume over the link must be known. For any pair of cities in the network, the two-way
traffic between the cities is supplied by the CNDC Model user. However, the actual traffic over these links will depend on the number of cities which send their traffic indirectly over the links, instead of transmitting on some other links. The capability to share satellite facilities among several cities is permitted by the SBS tariff and the usage costs are prorated among the users on the basis of their proportion of the total traffic volume over the link. In order to calculate the cost per circuit for SBS satellite links (or CPS tariffs in general), the total traffic volume over the link must be known. Based on this traffic level, the facility can be sized and costed. In a mixed terrestrial/SBS type of network, two alternatives exist for communicating voice traffic:

1) terrestrial links, and
2) satellite links.

For each city in the network, the costs of each alternative must be compared to determine the least cost link for the distance and traffic volume under consideration. The basic problem in performing such a comparison is in fixing the satellite link costs. A method is required to determine which cities will share a single satellite facility and how much traffic will be transmitted over the satellite link.

2.3.2 Methodology

The general methodology incorporated into the algorithm is as follows:

1) Select locations for SBS satellite facilities.

2) Partition the network around the individual satellite facility locations.

3) Perform cost decisions regarding terrestrial and satellite services between partitions.

4) Determine the number of satellite cities and the network configuration which yields the minimum cost network.
The selection of cities in which SBS satellite facilities will be installed is based on the determination of those cities with the heaviest traffic requirements. SBS facilities include Network Access Centers (NACs) and cities having these will be denoted as NAC cities.

Once NAC cities have been selected, the entire network will be divided into partitions, with each partition containing exactly one NAC city. Each partition has two categories of traffic:

1) intrapartition traffic, and
2) interpartition traffic.

Intrapartition traffic is defined as the traffic between any two cities within the same partition. The actual link between these two cities may include segments inside and/or outside of the partition. It will be assumed that all intrapartition traffic will use terrestrial links. Interpartition traffic is defined as the traffic between any two cities not members of the same partition. The choice of service (satellite or terrestrial) for interpartition links will be decided upon within the algorithm on the basis of cost comparisons. An initial assumption is made that all interpartition traffic will be routed through the NAC cities in the respective partitions. This will have the effect of producing the lowest possible cost per circuit for satellite links, since these are volume-sensitive costs. A determination is then made as to what proportion of the initially assumed traffic volume through the NAC can cost justify the use of the satellite link versus the terrestrial alternative. The algorithm will iteratively calculate the SBS cost per circuit based on total traffic through the NAC cities. During each iteration, it will perform cost comparisons of satellite versus terrestrial costs for each city pair. The total traffic volume of those cities favoring terrestrial links will be removed from the assumed total, routed through the NAC city, and a new satellite cost per circuit will be calculated. This process is repeated until no further cities change service. At this point, a network solution is obtained.
2.3.3 Detailed Discussion of Algorithm

There are four main subprograms contained in the algorithm:

1) NAC city selection,
2) Network partitioning,
3) Service decisionmaking, and
4) Cost response tracking.

Each of these will be described in detail. The least-cost network solution for an all terrestrial network is used as an input by the algorithm.

2.3.3.1 NAC City Selection

The basic assumption of the NAC city selection scheme is that the SBS service is more cost effective for high volume users than for low volume users. This is shown, in fact, by plotting SBS costs per circuit versus traffic volume (figure 2-8). Figure 2-8 shows that although there are spikes in the SBS cost function at specific volume levels, the general trend is for lower per circuit costs as volume increases. Based on the underlying structure of the SBS tariff, this trend will always be the case, regardless of the rates of the individual SBS cost components. Given this general characteristic, the NAC city selection scheme places NACs in those cities with the highest traffic volumes on their communications links. This should ensure the lowest possible costs for satellite links with the network. Note that no mention has yet been made of whether the satellite link is more cost effective than an alternate terrestrial link.

In this NAC city selection scheme, called the cluster algorithm, it is initially assumed that there are satellite facilities at all NNODES nodes in the network. Using the terminology of this algorithm, there are initially NNODES clusters. Links in the network are then investigated one at a time in ascending order of terrestrial cost. The first link investigated would then be that which has the least terrestrial cost. This link is defined by two nodes. The node with the least total traffic, A, is merged into the node...
Figure 2-8. Baseline SBS Cost Function (Actual Rates)
with the most traffic, B, to form a cluster in which node B would have
satellite facilities and node A could send any of its traffic via satellite
through B. At this point there are NNODES-1 clusters since A has now lost
its satellite facilities. This procedure of investigating links is repeated
until there are only two clusters. As each link is investigated (in ascen­
ding order of terrestrial cost) its two defining nodes are identified. If
both nodes still possess satellite facilities, the less busy node (with all
nodes in its cluster) will be merged into the more busy node (with all the
other nodes in its cluster) to form one larger cluster from two smaller clus­
ters. The list of nodes resulting from this scheme is arranged in such a
fashion that the first N names in the list represent the best placement of
satellite facilities if only N such facilities were to exist in the network.
This list is then used to iteratively assign NACs to those cities which
represent the best location for satellite facilities, starting with three NAC
locations. (A minimum configuration of three NAC locations is required by
the SBS tariff.) Within this first iteration, the algorithm will then par­
tition the network around these three NAC cities, determine the least-cost
service for each interpartition link in the network, and calculate the total
network cost. A second iteration will be performed in which the next city in
the list will be assigned NACs. The algorithm will then repartition the
entire network around each of the NAC cities and calculate a new network cost
solution.

In general, if the new solution is less than that resulting from the
previous iteration, the process is repeated by assigning NACs to that city
next in the list. When the new solutions are consistently greater than the
previous solutions, then the process is terminated and some previous solution
provides the minimum cost network solution found by the algorithm. Due to
the spikes in the SBS cost function, it is possible that total network cost
solution resulting from any particular iteration may be higher than that of
the previous iteration, yet the previous solution does not represent the true
minimum cost network. With a one-step ahead comparison (i.e., the (I+1)th
solution is compared to the Ith solution), there is a danger in terminating
the algorithm prematurely. In order to avoid this, the algorithm must pro­
ceed through several iterations in order to determine the minimum network
solution. For clarity, this process will be discussed under cost response
tracking.
An example will help to illustrate the NAC city selection scheme. Consider the network shown in figure 2-9. Its traffic and cost tables are also shown. The network consists of six cities labeled A, B, C, D, E, F. The sorted list of link costs and list of total traffic for each node are shown below.

<table>
<thead>
<tr>
<th>LINK</th>
<th>COST</th>
<th>NODE</th>
<th>VOLUME</th>
</tr>
</thead>
<tbody>
<tr>
<td>EF</td>
<td>19</td>
<td>A</td>
<td>1500</td>
</tr>
<tr>
<td>AB</td>
<td>25</td>
<td>B</td>
<td>1160</td>
</tr>
<tr>
<td>AE</td>
<td>28</td>
<td>C</td>
<td>1550</td>
</tr>
<tr>
<td>CF</td>
<td>32</td>
<td>D</td>
<td>1960</td>
</tr>
<tr>
<td>BE</td>
<td>43</td>
<td>E</td>
<td>1310</td>
</tr>
<tr>
<td>CE</td>
<td>47</td>
<td>F</td>
<td>1140</td>
</tr>
<tr>
<td>AD</td>
<td>51</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BF</td>
<td>53</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BD</td>
<td>55</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DF</td>
<td>58</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DE</td>
<td>61</td>
<td></td>
<td></td>
</tr>
<tr>
<td>BC</td>
<td>63</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AF</td>
<td>76</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AC</td>
<td>88</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CD</td>
<td>96</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Initially, it is assumed there are satellite facilities at all six nodes. Link EF is investigated first. Since both have satellite facilities and E has more traffic than F, F is merged into E and there are now satellite facilities at the 5 nodes A, B, C, D, E. With link AB, both nodes have satellite facilities. B is merged into A leaving the 4 nodes A, B, D, E with satellite facilities. Both nodes in link AE, the next link in the list, have satellite facilities. E is merged into A leaving satellite facilities at the three nodes A, C, D. The next three links CF, BE, CE are skipped because not both nodes in these links have satellite facilities. Link AD is investigated next. A is merged into D leaving two nodes with satellite facilities C and D. The list generated by the cluster algorithm would be D, C, A, E, B, F. The first three in the list represent NAC placement with three NAC locations, the first four the NAC placement with four nodes.
Figure 2-9. Sample Network, Traffic and Cost Tables
The algorithm will enable the user to specify the placement of NACS. In this case the algorithm will not sort the nodes by traffic volume but will use the list of NACs specified and perform the partitioning around this set.

It is conceivable that the all terrestrial solution will be optimal. If this is the case, the solution will indicate that no NACs were placed and the all terrestrial solution has been selected.

2.3.3.2 Network Partitioning

The network partitioning component of the algorithm requires a set of NAC cities as its input. The total network is divided into non-overlapping partitions, with the number of partitions equal to the number of NAC cities. A partition is defined as a NAC city and an associated set of non-NAC cities within the network having the following characteristics:

For each non-NAC city within the network, its nearest (in a cost-wide sense) NAC neighbor is identified and the non-NAC city is assigned to that NAC city's partition. Only terrestrial costs are considered since it has been assumed that all intrapartition traffic will use terrestrial services. The intent in constructing the network partitions is to configure each intrapartition network such that its traffic costs are a minimum. Since terrestrial services are being considered, there are three factors which will determine the link costs. These are:

1) Distance of link,

2) Listed/unlisted status of cities on link, and

3) Tariff rate structure.

The minimum cost links for intrapartition traffic will be identical to those resulting from a solution of the terrestrial-only least cost network problem. The network partitioning component of the algorithm will require this solution as an input. This solution will also be used in the service decision-making component.
2.3.3.3 Service Decisionmaking

The service decision making component of the algorithm will process through each partition to determine the service to be used (terrestrial or satellite) for its traffic links to all other partitions. The specific decision criteria for choosing a service for any particular interpartition link will be the crossover cost associated with the link. The crossover cost is defined as that total cost per circuit associated with terrestrial services between two nodes which will determine whether satellite or terrestrial services are employed. If the associated satellite cost per circuit between two nodes is less than the corresponding crossover cost, then satellite service will be used to link the nodes. An example of a crossover cost is illustrated in figure 2-10. In this example, the service decision for the link between city A in partition I and city D in partition J will be made. There are two alternatives to consider:

1) Via the terrestrial link AD (assumed least-cost terrestrial path), or

2) Via terrestrial link AB to satellite link BC to terrestrial link CD.

In order for the satellite alternative to be cost effective, the following relationship must hold:

\[ C_{AB} + C_{BC} + C_{CD} < C_{AD}, \]  

or equivalently

\[ C_{BC} < C_{AD} - (C_{AB} + C_{CD}) \]  

In words, the satellite cost must be less than the difference between the least-cost terrestrial link and the sum of the tail costs in each partition \((C_{AB} + C_{CD})\). The expression on the right side of the inequality (2) is defined as the crossover cost. Others studying similar problems have defined and used crossover distances. They have assumed some constant satellite cost in doing so. Since we are dealing with usage sensitive satellite tariffs with a variable cost per circuit function, cost per circuit must be used as the service decision criterion.
Figure 2-10. Example of a Crossover Cost

\[ \text{CROSSOVER COST}_{AD} = C_{AD} - C_{AB} - C_{CD} \]
For any partition, the crossover costs associated with links to all other partitions can be calculated. Each partition will have its own set of crossover costs, with each crossover cost having an associated traffic volume. The set of crossover costs for any partition can be ordered from some minimum value to some maximum value. In making the service decision, there are three possibilities regarding the value of the satellite cost. (See figure 2-11).

Case 1 - the satellite cost is less than the minimum crossover cost.

Case 2 - the satellite cost is greater than the maximum crossover cost.

Case 3 - the satellite cost is between the minimum and maximum crossover costs.

Figure 2-11. An Ordering of Crossover Costs for Any Partition

For cases 1 and 2, the decisions are clearcut. In case 1, since the cost of the satellite link is less than the costs of all competing terrestrial links, all interpartition traffic originating in the partition under consideration
will be routed via satellite. In case 2, the exact opposite is true and all traffic is routed terrestrially. Case 3 involves a situation in which satellite routes are cost effective for some links and not for others.

Before proceeding with a discussion of case 3, the assumptions regarding the calculation of the satellite cost per circuit should be stated. As previously mentioned, each partition has associated with it a set of ordered pairs of crossover costs and traffic volumes corresponding to directional links to other partitions. To calculate the satellite cost per circuit, it will be initially assumed that all nodes within any partition will use the satellite link. The total volume of originating interpartition traffic for all nodes within a partition is used to determine satellite cost per circuit. This cost may produce one of the three possibilities previously discussed. If case 3 occurs, the initial satellite cost will need to be adjusted in some iterative manner to complete the service decisionmaking process.

When the initial satellite cost ($SC_0$) falls somewhere in between the minimum and maximum crossover costs, two groups of links result: one terrestrial and one satellite (see figure below).

Figure 2-12. Grouping of Partition Links with Initial Satellite Cost

2-25
Since the value of $SC_0$ was determined on the basis of the traffic from both groups, the total traffic associated with the links in the terrestrial group must be removed from the total originating partition traffic in order to calculate a new satellite cost ($SC_1$). The new cost, $SC_1$, should be higher than the previous cost and may result in terrestrial service becoming more cost effective for some links. If this be the case, the volume associated with those links is removed from the remaining satellite total and a new satellite cost is again calculated. The process is repeated until either:

1) the new satellite cost results in no further service changes, or

2) the new satellite cost exceeds the maximum crossover cost, in which case all links are terrestrial service.

In general, the following will be true of the calculated satellite costs:

$$SC_i < SC_{i+1}$$

Since the process is started with the minimum possible satellite cost per circuit, this cost can only increase. It is possible however, due to the nature of the SBS cost function, that particularly if the satellite traffic volume associated with $SC_1$ was incrementally enough to exceed the existing SCU or NAC capacities. The difference will increase as volume decreases toward the stable (high volume) portion of the SBS cost curve. In order to avoid instability in the algorithm, an assumption is made that once links have decided upon terrestrial service, this decision is final. There will be no re-evaluating procedure for these links if the satellite cost comes down below the link crossover cost. This assumption is reasonable since:

- SBS cost function generally decreases as volume increases.

- For high volume links, the magnitude of increases in SBS cost associated with decreases in volume will be small.

- The effect of the difference between the new satellite cost and the alternate terrestrial cost on the total network cost will be small for any one link.
2.3.3.4 Cost Response Tracking

Since the objective of the algorithm is to determine the least-cost network, some method of examining the total network cost solutions is needed. The functions of the cost response tracking component of the algorithm are:

- to monitor the total network cost for the various solutions as additional satellite cities are included in the network,
- to determine when the sequence of total network costs begins to diverge,
- to terminate the algorithm's solution iteration process, and
- to identify the minimum solution in the sequence.

Figure 2-13 illustrates the expected behavior of total network costs as the number of NAC cities within the network increases. The figure is a plot of total network costs versus the number of NAC cities for an N node network. The solution space ranges from an all terrestrial network with no satellite links to an all satellite network with no terrestrial links. Although the behavior of a total network cost function will depend on the topology of the network under consideration and the tariffs used, this concave cost curve is realistic and characteristic of the SBS and AT&T tariffs. Note that the all terrestrial case will be the least-cost network solution for a "terrestrial only" network.
The general methodology of the cost tracking component will be to first store off the total network cost solution resulting from each iteration of the NAC selection, partitioning, and service decisionmaking components as well as that for the all terrestrial solution. As each new network solution is generated, it will be compared to the stored file of previous solutions. The algorithm will be terminated when the cost tracking component becomes "reasonably confident" that the total network cost solutions are diverging (i.e., the minimum solution has been reached). At this point, stored solutions can be examined and the minimum solution identified. The solution will correspond to a network with a specific number of NAC nodes (for partitions) which were selected from the highest volume nodes in the stored node list or the all terrestrial situation. The number of NAC nodes is effectively a pointer into this sorted list and will allow the minimum cost network to be easily reconstructed without iterative processing through the list.
The objective of the termination scheme will be to detect a divergent trend in the sequence of total network costs and to signal the algorithm to halt its iterative generation of solutions. It is appropriate to terminate the algorithm when it can be reasonably assured that further partitioning of the network will only result in higher cost network solutions. There are many possible schemes which can be developed to terminate the algorithm, ranging from the very simple to the complex.

The termination scheme selected is very simple. The iterative generation of solutions will terminate when the most recent solution generated is a factor larger than the best solution. This factor may be adjusted to meet the user's needs as one studies the cost response behavior of the system's output.
3.0 CNDC PROGRAM MODULES

3.1 Executive Program

3.1.1 General Description

The Executive Program (CNDC) is the main program which controls the execution of the Communications Network Design and Costing Model. Program execution is initiated and terminated from CNDC which performs a logical sequence of program module calls. Called program modules include the following:

INPUT
INIT
COSTNG
NETWRK
SBSMIX
OUTPUT

A hierarchy chart of the CNDC program is illustrated in Figure 3-1. The overall program flow is illustrated in Figure 3-2. A brief description of each of the called modules is provided in the following paragraphs.

INPUT MODULE

All user input necessary for execution of the CNDC Model is controlled through the INPUT module. Specifically, the user has the capability to construct one or more networks for analysis, select the desired common carrier tariffs to be used or select and define his own, and set parameters controlling program execution. The INPUT module is described in more detail in sections 3.3 to 3.5.
Figure 3-1. CNDC Hierarchy Chart
INIT MODULE

This module performs all the initialization of data structures required for program execution. The INIT module is described completely in section 3.6.

COSTNG MODULE

This module uses the network and tariff data specified as input to determine link costs between all nodes of the specified network for all tariffs. It then determines the service which provides the least cost for every link in the network. The COSTNG module is described in this section.

NETWRK MODULE

This module contains the algorithm which solves the least cost routing and least cost network problems for terrestrial only or mixed terrestrial-Western Union satellite cases. It generates the information needed by the OUTPUT module. NETWRK is described in this section.

SBSMIX MODULE

This module performs the optimization for the problem which contains one terrestrial and one SBS tariff. The SBSMIX module is described in section 3.2.

OUTPUT MODULE

The OUTPUT module controls all report generation from an execution of the program. The user is provided the capability to select various output reports. The OUTPUT module is described in detail in section 3.7.
During the input session, the user is allowed to specify multiple problem input sets. This capability allows several program runs to be batched together to be run in succession without operator/user intervention. At the end of each problem set, the CNDC determines if there are any additional problems to be run. If there are, CNDC branches back to the INIT module and performs the next run. This process continues until there are no further runs to be made. At this time the program terminates.

Following is a description of the CNDC, COSTNG, and NETWRK modules. Each description contains a list of all the data structures referenced or changed within the specific module. With each structure name is the common block to which it belongs. Error conditions encountered within the module are also described. Each error is assigned a number. A further listing of errors, by error numbers is contained in appendix B.

A fatal error describes a condition which would prevent the model from completing its task. These errors should not occur. If a fatal error occurs, there is a problem with the model or a file in the data base. For this reason an error message is printed and the model is terminated.

A non-fatal error is one from which the program can recover and continue to run. When a non-fatal error occurs, an error message is written and the program continues.
3.1.3 **CNDC Module**

NASA COMMUNICATIONS NETWORK DESIGN AND COSTING MODEL
MAIN PROGRAM -- CNDC -- 22 NOVEMBER 1982

**PURPOSE**
DIRECT OVERALL PROCESSING OF CNDC MODEL

**INPUT ARGUMENTS**
NONE

**OUTPUT ARGUMENTS**
NONE

**DATA STRUCTURES**
(U - USED, S - DEFINED, R - READ)
NCASE /RUN/ S
OPTION /RUN/ U

**LOCAL VARIABLES**
FLAG LOGICAL
FLAG THAT INDICATES WHETHER TO CONTINUE PROCESSING OR TERMINATE
TRUE CONTINUE
FALSE TERMINATE

**SUBPROGRAMS CALLED**
INPUT
INIT
COSTNG
NETWRK
OUTPUT
SBSMIX

**ERRORS**
FATAL
NONE
NONFATAL
NONE

3-5
Figure 3-2. Nasa CNDC Model Program Flow
3.1.4 COSTNG Module

SUBROUTINE COSTNG

PURPOSE
CALCULATE THE MINIMUM COST OF A DIRECT CIRCUIT BETWEEN
ALL PAIRS OF NODES IN THE NETWORK ACROSS TARIFFS

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
(COST /CSTOUT/ S
NNODES /RUN/ U
NTARIF /RUN/ U
OPTION /RUN/ U
SVC /CSTOUT/ S
DEBUG /LUN/ U
SATNVC /SBSMX2/ S

LOCAL VARIABLES
BIGNUM R*4 CONSTANT /16.0**62/ LARGE REAL
NUMBER TO REPRESENT INFINITY
CPC R*4 TEMPORARY COST FOR SBS-TYPE TARIFF
CANDIDATE FOR MINIMUM
I I*4 LOOP INDEX - FROM CITIES
J I*4 LOOP INDEX - TO CITIES
K I*4 LOOP INDEX - TARIFFS
KTEMP I*2 TEMPORARY TARIFF INDEX
TEMP R*4 TEMPORARY COST - CANDIDATE FOR MINIMUM
DIST I*4 DISTANCE BETWEEN TWO NODES

SUBPROGRAMS CALLED
ATCOST
CATEG
LISTED
MILES
ROWTOT
SBCOST
WUCOST

ERRORS
FATAL
NONE
NONFATAL
NONE
COSTING

INITIALIZE SERVICE FOR EACH LINK TO NONE

OPTION SBS ONLY?

ALL NODES CONSIDERED?

DETERMINE ORIGINATING TRAFFIC

ALL TARIFFS CONSIDERED?

DETERMINE NODE COST

DETERMINE IF MINIMUM COST

RETURN

DETERMINE LINK COST FOR TARIFF

COMPUTE LINK DISTANCE

DETERMINE IF MINIMUM COST

ALL LINKS CONSIDERED?

YES

NO

YES

NO

Figure 3-3 Costing Logic Flow

3-8
3.1.5 NETWRK Module

SUBROUTINE NETWRK

PURPOSE
DETERMINE THE COMBINATION OF LINKS AND SERVICES THAT WILL PROVIDE
VOICE COMMUNICATION BETWEEN ALL CITY PAIRS AT MINIMUM COST
FOR ALL BUT SBS MIXED OPTION

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
   COST    /CSTOUT/    U
   MINCPC  /CSTOUT/    S
   NNODES  /RUN/       U
   OROUTE  /CSTOUT/    S
   DEBUG   /LUN/       U
   OPTION  /RUN/       U

LOCAL VARIABLES
   BIGNUM     R*4   CONSTANT /16.0**62/ LARGE REAL NUMBER TO
                REPRESENT IN
   I         I*4   LOOP INDEX FOR INTERMEDIATE NODE
   J         I*4   LOOP INDEX FOR FROM CITIES
   K         I*4   LOOP INDEX FROM TO CITIES
   S         R*4   TEMPORARY COST - CANDIDATE FROM MINIMUM
   NUMNDS   I*4   NUMBER OF NODES
   MINIK     R*4   MINIMUM COST PER CIRCUIT FROM I TO K
   MINJI     R*4   MINIMUM COST PER CIRCUIT FROM J TO I

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
NONE
NONFATAL
NONE
Figure 3-4. Network Logic Flow
3.2 SBSMIX DESCRIPTION

The user has the option within the SBS mixed problem to specify the placement of NACs or to have the model, through an iterative process (starting with three NAC locations) determine the optimal location of NACs. In either case, if the all terrestrial solution turns out to be optimal, no NACs will be placed.

If the user selects to have the model select the optimal NAC location, there must be a scheme to choose where to place the NACs. All nodes in the network are arranged in a list such that the first \( N \) names in the list would represent the optimal placement of satellite facilities if only \( N \) such facilities were to exist in the network. NACs are initially placed at the first three nodes in the list and a system cost is determined. NACs are then placed at the first four nodes in the list, and again a system cost is computed. This process is continued, adding one node at a time, until it appears that the solutions are no longer improving. At this point the iterative process is terminated and the NAC arrangement which yielded the lowest system cost is selected as optimal.

Within the iterative procedure once a trial NAC configuration is known, the entire network is partitioned so that each non-NAC node is associated with that NAC to which the node can communicate at the least terrestrial cost. Once the network is partitioned, the initial service configuration is established where by each node sends all intrapartition traffic via terrestrial links and all interpartion traffic via satellite links. This defines the maximum use of each NAC location. The cost associated with this configuration is computed.

Each node is then given the opportunity to choose to send portions of its traffic by terrestrial links if these prove to be of less cost. Once traffic has been removed, the cost for each NAC location is again computed. Again the nodes may remove traffic from the satellite based on the new cost. This process is repeated until the system stabilizes and no more traffic is removed from satellite service. A more complete description of this algorithm is contained in section 2.3. A description of the SBSMIX module
follows in the next section. The logic flow diagram is Figure 3-6. The hierarchy chart for SBSMIX is contained in Figure 3-5.
Figure 3-5. SBSMIX Hierarchy Chart
3.2.1 SBSMIX Module

SUBROUTINE SBSMIX

PURPOSE
DETERMINE OPTIMAL SBSMIX SOLUTION FOR THE CURRENT NETWORK

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)

SBMX /SBSMXD/ U
NACMB /SBSMXD/ U
SYSTBL /SBSMX2/ U
SVC /CSTOUT/ S
NNODES /RUN/ U
TOTSAT /SBSMX2/ S
DEBUG /LUN/ U
NODPTR /SBSMXD/ U
NACLOC /SBSMXD/ U
SATNVC /SBSMX2/ U

LOCAL VARIABLES
USRDEF I*4 CONSTANT /1/ USER HAS SPECIFIED NAC PLACEMENT
FINISH L PROGRAM NAC PLACEMENT COMPLETE
SYSCST R*4 COST OF SYSTEM WITH NACNMB NACS
I,J I*4 LOOP INDICES
PRT1 I*4 PARTITION TO WHICH FROM NODE BELONGS
PRT2 I*4 PARTITION TO WHICH TO NODE BELONGS
BSTSOL I*4 NUMBER OF NACS IN OPTIMAL SOLUTION
CODE I*4 FOUR CHARACTER CODE OF USER SPECIFIED NAC
SKIP L BSTSOL WAS THE LAST SOLUTION COMPUTED
ALTERNL I*4 NO NEED TO REPEAT COMPUTATIONS
ALTERNL I*4 CONSTANT /2/ ALL TERRESTRIAL SOLUTION

SUBPROGRAMS CALLED
DEFTER
SRNAC
PRTITN
SBSC
DECIDE
TOTCST
ALDON
MNMPTR

ERRORS
FATAL
NONE
NONFATAL
NONE
DETERMINE ALL TERRESTRIAL SOLUTION

USER SPECIFY NAC PLACEMENT?

INITIALIZE NUMBER OF NAC LOCATIONS TO 3

PARTITION NETWORK

INITIALIZE SERVICE FOR THIS PARTITION

COMPUTE COST FOR EACH NAC

ALLOW EACH LINK TO MAKE TERRESTRIAL SATELLITE DECISION

COMPUTE COST FOR EACH NAC

SYSTEM STABLE?

Figure 3-6. SBSMIX Logic Flow (Sheet 1 of 2)
Figure 3-6. SBSMIX Logic Flow (Sheet 2 of 2)

3-15
D

COMPUTE TOTAL SYSTEM COST

DETERMINE IF ALL TERRESTRIAL OPTIMAL

E

ALL TERRESTRIAL OPTIMAL?

NO

YES

SET SERVICE TO ALL TERRESTRIAL

CALCULATE TOTAL SYSTEM COST

RETURN

Figure 3-6. SBSMIX Logic Flow (Sheet 3 of 3)
3.2.2 DEFTER Module

SUBROUTINE DEFTER

PURPOSE
DETERMINE LINKS THAT WILL PROVIDE TERRESTRIAL COMMUNICATION BETWEEN ALL CITY PAIRS AT MINIMUM COST AND TOTAL SYSTEM COST OF ALL TERRESTRIAL SOLUTION

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)

- NNODES /RUN/ U
- COST /CSTOUT/ S
- SVC /CSTOUT/ S
- TRAFFIC /TRFC/ U
- NETOTL /CSTOUT/ U
- MINCPC /CSTOUT/ U
- TOTSAT /SBSMX2/ S
- DEBUG /LUN/ U

LOCAL VARIABLES
FROM I*4 NODE FROM WHICH TRAFFIC ORIGINATES
TO   I*4 NODE AT WHICH TRAFFIC TERMINATES
DIST I*4 AIRLINE MILEAGE BETWEEN TWO NODES
ISCHED I*4 TERRESTRIAL TARIFF SCHEDULE BETWEEN TWO NODES
TARIF I*4 NUMBER OF TERRESTRIAL TARIFF
TRAFIC I*4 ORIGINAL TRAFFIC BETWEEN NODAL PAIR
TRRCST R*4 TOTAL TERRESTRIAL COST
NETFT I*4 TOTAL TRAFFIC ACROSS LINK
MINFT R*4 MINIMUM COST FOR EACH LINK
LSTDTO I*4 CATEGORY A STATUS OF TO NODE

SUBPROGRAMS CALLED
MILES
LISTED
ATOST
NETWRK
MTPY
TRSTOT

ERRORS
FATAL NONE
NONFATAL NONE
DEFTER

ALL LINKS CONSIDERED?

DETERMINE TERRESTRIAL COST

DETERMINE LEAST COST ROUTES

COMPUTE TOTAL SYSTEM COST

RETURN

Figure 3-7. DEFTER Logic Flow
3.2.3 SRTNAC Module

SUBROUTINE SRTNAC

PURPOSE
DETERMINE THE ORDER OF THE NODES IN WHICH NACs WILL BE PLACED IN THE NETWORK

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)

S
U
NNODES /RUN/
TRAFIC /TRFC/
DEUG /LUN/
MINCPC /CSTOUT/

LOCAL VARIABLES

I,J,K I*4 LOOP INDICES
NVAL I*4 NUMBER OF ITEMS IN LIST OF DISTANCES BETWEEN PAIRS OF NODES
NACPTR I*4 POSITION OF EACH NODE IN NAC LIST
NCLUST I*4 NUMBER OF CLUSTERS
NLINK I*4 ITEM IN DISTANCE ARRAY BEING REFERENCED
POS . I*4 POSITION IN ORDER ARRAY
POS A I*4 POSITION OF FROM NODE IN NAC LOCATION LIST
POS B I*4 POSITION OF TO NODE IN NAC LOCATION LIST
LSTTOP I*4 LAST ITEM TO BE SHIFTED DOWN IN NACLOC LIST
FSTTOP I*4 FIRST ITEM TO BE SHIFTED UP IN NACLOC LIST
MCST R*4 ARRAY OF LINK COSTS
ORDER I*4 POSITIONS OF LINK DISTANCES SORTED IN ASCENDING ORDER
END I*4 END INDEX OF LOOP
A I*4 FROM NODE
B I*4 TO NODE
NAC I*4 NUMBER OF NAC
SHIFT L INDICATES IF ORDER OF NACs ACTUALLY CHANGED BY MERGING A NODE INTO A PARTITION
TOTFC I*4 TOTAL TRAFFIC AT NODE
MAX R*4 MAXIMUM COST IN CURRENT SET OF LINKS
MIN R*4 MINIMUM COST IN CURRENT SET OF LINKS
KK,II I*4 LOOP INDICES
NEL I*4 NUMBER OF LINK COSTS IN SET
LOCATION IN CURRENT LINK SET OF MAX COST
NUMBER OF ITEMS IS LIST TO BE SORTED

SUBPROGRAMS CALLED
SORTR
INDX

ERRORS
FATAL
NONE
NONFATAL
NONE
Figure 3-8. SRTNAC Logic Flow
3.2.4 PRTTN Module

SUBROUTINE PRTTN

PURPOSE
PARTITION NETWORK AROUND NAC CITIES

INPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
NACLOC /SBXMXD/ U
NACNMB /SBXMXD/ U
NNODES /SBXMXD/ U
NODPTR /SBXMXD/ S
MINCPC /CSTOUT/ U
DEBUG /LUNOUT/ U

LOCAL VARIABLES
MINCST I*4 MINIMUM TERRESTRIAL COST REQUIRED TO GET
FROM CURRENT NODE TO A NAC LOCATION
MINPNT R I*4 NUMBER OF NAC THAT IS CHEAPEST FOR NODE
TO GET TO
NAC I*4 NUMBER OF CURRENT NAC UNDER CONSIDERATION
NODE I*4 NUMBER OF CURRENT NODE UNDER CONSIDERATION
NACPTR I*4 NODE PRINTER OF CURRENT NAC
MAXINT I*4 MAXIMUM I*4 VALUE
I I*4 LOOP INDEX

SUBPROGRAMS CALLED
ISNAC

ERRORS
FATAL
NONE
NONFATAL
NONE
Figure 3-9. PRTITN Logic Flow
3.2.5 SBSC Module

SUBROUTINE SBSC

PURPOSE
COMPUTE COST PER CIRCUIT FOR EACH NAC LOCATION

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
NACNMB /SBSMXD/ U
NODPTR /SBSMXD/ U
NNODES /RUN/ U
TRAFIC /TRFC/ U
SATNVC /SBSMX2/ S
SATCST /SBSMX2/ S
SVC /CSTOUT/ U
DEBUG /LUN/ U

LOCAL VARIABLES
SBS I*4 TARIFF NUMBER OF SBS TARIFF /2/ CONSTANT
PART1 I*4 NUMBER OF FROM PARTITION
NVC I*4 NUMBER OF VOICE CIRCUIT USING NAC
FROM I*4 FROM NODE
TO I*4 TO NODE
PRIFRM I*4 PARTITION TO WHICH FROM NODE BELONGS
PRITTO I*4 PARTITION TO WHICH TO NODE BELONGS
BIGNUM R*4 CONSTANT /16.0**62/LARGE REAL NUMBER TO REPRESENT INFINITE COST

SUBPROGRAMS CALLED
SBCOST

ERRORS
FATAL
NONE
NONFATAL
NONE
Figure 3-10. SBSC Logic Flow
3.2.6 DECIDE Module

SUBROUTINE DECIDE

PURPOSE
ALLOW EACH LINK TO MAKE TERRESTRIAL VS SATELLITE DECISION

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
NACNMB /SBSMXD/ U
NNODES /RUN/ U
NODPTR /SBSMXD/ U
NACLOC /SBSMXD/ U
MINCPC /CSTOUT/ U
SVC /CSTOUT/ U
SATCST /SBSMX2/ U
SYSTBL /SBSMX2/ S
DEBUG /LUN/ U

LOCAL VARIABLES
I I*4 LOOP INDEX
FROM I*4 PARTITION OF FROM NODE
PART1 I*4 PARTITION OF FROM NODE
NOD1 I*4 NODE WHICH HAS NAC IN FROM PARTITION
TAIL1 R*4 COST OF TAIL TO NAC
NOD2 I*4 NODE WHICH HAS NAC IN TO PARTITION
TO I*4 TO NODE
PART2 I*4 PARTITION OF TO NODE
FULL R*4 MINIMUM TERRESTRIAL COST OF LINK
TAIL2 R*4 MINIMUM COST OF TRAIL FROM TO NAC
CRS0VR R*4 CROSSOVER COST FOR THIS LINK
TRSTRL I*4 CONSTANT /1/ TERRESTRIAL SERVICE
CHANGE I*4 INDICATES OF NAC LOCATION MADE A CHANGE
THIS ITERATION
BIGNUM R*4 CONSTANT /16.0**52) LARGE REAL TO
REPRESENT INFINITE COST

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
NONE
NONFATAL
NONE
Figure 3-11. DECIDE Logic Flow (Sheet 1 of 2)
DETERMINE TAIL TO SENDING SATELLITE

DETERMINE CROSSOVER COST

CHEAPER TERRESTRIAL?

SWITCH TO TERRESTRIAL SERVICE

Figure 3-11. DECIDE Logic Flow (Sheet 2 of 2)
SUBROUTINE TOTCST(SYSCST)

PURPOSE
   COMPUTE TOTAL SYSTEM COST

INPUT ARGUMENTS
   NONE

OUTPUT ARGUMENTS
   SYSCST  R*8  TOTAL SYSTEM COST

DATA STRUCTURES
   (U - USED, S - DEFINED, R - READ)
      NNODES  /RUN/  U
      NETOTL  /CSTOUT/  S
      SATOTL  /SATMXD/  S
      SUC  /CSTOUT/  U
      NODPTR  /SBSMXD/  U
      NACLOC  /SBSMXD/  U
      TRRCST  /CSTMXD/  S
      TRRMLG  /CSTMXD/  S
      TRRCKT  /CSTMXD/  S
      MINCPC  /CSTMXD/  U
      STLMLG  /CSTMXD/  S
      STLCKT  /CSTMXD/  S
      STLCSST  /CSTMXD/  S
      SATNVC  /SBSMX2/  U
      SATCST  /SBSMX2/  U
      NACNMB  /SBSMX2/  U
      TRAFIC  /TRFC/  U
      STABLE  /SBSMX2/  S
      DEBUG  /LUN/  U

LOCAL VARIABLES
   FROM  I*4  NUMBER OF FROM NODE FOR TERRESTRIAL SERVICE
   TO  I*4  NUMBER OF TO NODE FOR TERRESTRIAL SERVICE
   TRRSTR  I*4  CONSTANT /1/ TERRESTRIAL TARIFF
   PART1  I*4  NUMBER OF FROM PARTITION
   NAC1  I*4  NODE WHICH IS NAC LOCATION IN FROM PARTITION
   PART2  I*4  NUMBER OF TO PARTITION
   NAC2  I*4  NODE WHICH IS NAC LOCATION IN TO PARTITION
   NACFRM  I*4  NODE WHICH IS NAC LOCATION IN FROM PARTITION
   NACTO  I*4  NODE WHICH IS NAC LOCATION IN TO PARTITION
   I  I*4  LOOP INDEX
   TRAFC  I*4  NUMBER OF VOICE CIRCUITS BETWEEN TWO NODES
   NETFT  I*4  NUMBER OF VOICE CIRCUITS - FROM TO
   MINET  R*4  MINIMUM COST PER CIRCUIT - FROM TO
   PROD  R*4  TOTAL CIRCUIT COST

SUBPROGRAMS CALLED
   TRSTOT
   MILES
   MTPY
ERRORS
FATAL
NONE
NONFATAL
NONE
Figure 3-12. TOTCST Logic Flow (Sheet 1 of 2)
Figure 3-12. TOTCST Logic Flow (Sheet 2 of 2)
3.2.8 TRSTOT Module

SUBROUTINE TRSTOT (FROM, TO, TRAFC)

PURPOSE

ADD SPECIFIED TRAFFIC TO LINKS THAT MAKE UP
THE LEAST COST NETWORK BETWEEN SPECIFIED NODES

INPUT ARGUMENTS

FROM I*4 NUMBER OF FROM NODE
TO I*4 NUMBER OF TO NODE
TRAFC I*4 NUMBER OF VOICE CIRCUITS TO ADD TO
EACH LINK OF PATH BETWEEN FROM AND TO

OUTPUT ARGUMENTS

NONE

DATA STRUCTURES

(U - USED, S - DEFINED, R - READ)

ROUTE /CSTOUT/ U
NETOTL /CSTOUT/ S
DEBUG /LUN/ U

LOCAL VARIABLES

CURFRM I*4 CURRENT FROM NODE ALONG LEAST COST ROUTE
CURTO I*4 CURRENT TO NODE ALONG LEAST COST ROUTE

SUBPROGRAMS CALLED

NONE

ERRORS

FATAL
NONE
NONFATAL
NONE
DETERMINE START NODE

DETERMINE NEXT NODE IN PATH

ADD TRAFFIC TO LINK

END OF PATH ?

RETURN

Figure 3-13. TRSTOT Logic Flow
3.2.9 ALDONE Module

SUBROUTINE ALDONE (FINISH,BSTSOL)

PURPOSE
DETERMINE IF OPTIMAL SOLUTION HAS BEEN FOUND

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
FINISH 
BSTSOL 
DATA 
(l - USED, S - DEFINED, R - READ)
TOTSAT /SBSMX2/ U
NACNMB /SBSMXD/ S
NNODES /RUN/ U
STABLE /SBSMX2/ U
DEBUG /LUN/ U

LOCAL VARIABLES
BSTCST R*4 SYSTEM COST OF OPTIMAL SOLUTION
I I*4 LOOP INDEX
FINFAC R*4 FACTOR BY WHICH CURRENT SOLUTION MUST EXCEED BEST SOLUTION TO TERMINATE NAC PLACEMENT

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
NONE
NONFATAL
NONE
Figure 3-14. ALDONE Logic Flow
3.2.10 ISNAC Module

LOGICAL FUNCTION ISNAC(NODE)

PURPOSE
DETERMINE IF SPECIFIED NODE IS ALREADY IN LIST OF
NETWORK NAC LOCATIONS

INPUT ARGUMENTS
NODE I*4 INTEGER POSITION OF NODE WITHIN NETWORK

OUTPUT ARGUMENTS
ISNAC L FUNCTION VALUE OF .TRUE. IF NODE IS IN
LIST OF NETWORK LINKS, .FALSE. OTHERWISE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
NACLOC /SBSMXD/ U
NACNMB /SBSMXD/ U
DEBUG /LUN/ U

LOCAL VARIABLES
FOUND L INDICATES IF NODE FOUND IN NACLOC LIST
I I*4 LOOP INDEX

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
NONE
NONFATAL
NONE
Figure 3-15. ISNAC Logic Flow
3.2.11 SORTR Module

SUBROUTINE SORTR (CODE, ORDER, NEL)

PURPOSE

SORTS THE NEL ITEMS IN CODE ARRAY IN ASCENDING ORDER

INPUT ARGUMENTS
CODE R*4 REAL ARRAY TO BE SORTED
NEL I*4 NUMBER OF ELEMENTS TO BE SORTED

OUTPUT ARGUMENTS
ORDER I*2 POSITION OF ITEMS IN CODE ARRAY IN ASCENDING ORDER. IT IS ASSUMED THAT ORDER HAS BEEN INITIALIZED BY CALLING ROUTINE

DATA STRUCTURES
NONE

LOCAL VARIABLES
INDX1 I*4 POINTER TO FIRST OF TWO ITEMS IN CODE TO BE COMPARED
INDX2 I*4 POINTER TO SECOND OF TWO ITEMS IN CODE TO BE COMPARED
IEND I*4 NUMBER OF COMPARISONS
SORTED L INDICATES IF LIST FULLY SORTED
I I*4 LOOP INDEX
POS I*2 POSITIONS OF CODE
HI I*2 TEMPORARY LOCATION FOR ORDER VALUE TO BE SHIFTED
H2 I*2 TEMPORARY LOCATION FOR ORDER VALUE TO BE SHIFTED

SUBPROGRAMS CALLED
NONE

ERRORS
NONE
3.2.12 INDX Module

INDX Module

SUBROUTINE INDX (ORDR, FROM, TO)

PURPOSE

DETERMINE THE FROM AND TO COORDINATES OF A POSITION WHICH CORRESPONDS TO A SLOT IN THE UPPER RIGHT TRIANGLE OF A SQUARE MATRIX

INPUT ARGUMENTS

ORDR  I*4  POSITION IN ONE DIMENSIONAL ARRAY WHICH REPRESENTS UPPER RIGHT TRIANGLE OF A SQUARE MATRIX WHERE ELEMENTS ARE ARRANGED IN ORDER BY COLUMN

OUTPUT ARGUMENTS

FROM  I*4  ROW IN SQUARE MATRIX ASSOCIATED WITH ORDR
TO    I*4  COLUMN IS SQUARE MATRIX ASSOCIATED WITH ORDR

DATA STRUCTURES

(U - USED, S - DEFINED, R - READ)
DEBUG /LUN/  U

LOCAL VARIABLES

COL    I*4  COLUMN NUMBER
ENDCOL I*4  POINTER TO END OF COLUMN
DISP   I*4  DISPLACEMENT BETWEEN ORDR AND END OF COLUMN

SUBPROGRAMS CALLED

NONE

ERRORS

NONE
3.3 INPUT DESCRIPTION

All user specified input necessary for execution of the CNDC Model is controlled through the INPUT module. During any input session, the user may specify problem data sets for one or more runs. The input session may be operated in batch mode. The batch processing input modules are described in section 3.9.

The INPUT module consists of three submodules:

1. NETCON - network construction
2. TARCON - tariff construction
3. EXCON - execution control

A brief description of each submodule is contained in the following paragraphs.

NETCON

This submodule allows the user to specify the network to be analyzed by the CNDC Model. During a network construction session, the user may create a network file to identify the nodes of the network. Nodes may be selected from a prestored list of SMSA cities or specified as any city that the user wishes to include in the network. The user may create several network files. The program will organize these files and maintain a directory of current network files. The user is also provided the capability to list the directory and the contents of any file in the directory. He is also provided the capability to delete any existing network files. In this case the program will automatically remove the file name from the network directory making it available for a new definition. The NETCON submodule is described in detail in section 3.4.
TARCON

This submodule allows the user to specify tariffs to be used as input to the CNDC model. During a tariff construction session, the user may create up to twelve different tariff files. These files are stored and can be specified as input to any particular run during program execution. TARCON provides the user with the capability to manage the tariff files. The program organizes existing tariff files in a directory. Options are available to create, delete, and list tariff files. The tariff directory may also be listed to examine which files are currently stored. The creation of a tariff file includes the use of any one of the three tariffs (AT&T, Western Union, or SBS) as a starting point. The user defined tariff follows the same structure and connectivity philosophy of the modeled tariff. TARCON is described in section 3.5.

EXCON

This submodule allows the user to specify input parameters which define the problem being solved and control the execution of the program. The user may specify multiple problem sets for batching program runs. For each run the user specifies the network file to use, the tariffs to be considered, and the output reports desired. The EXCON submodule is described in this section.

A hierarchy chart of the INPUT module is in Figure 3-16.
Figure 3-16. INPUT Hierarchy Chart
3.3.1 INPUT Module

SUBROUTINE INPUT (CONT)

PURPOSE
QUERY USER AND BUILD EXECUTION FILE OR TARIFF OR NETWORK FILES SPECIFIED

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
CONT LOGICAL

DATA STRUCTURE
(U - USED, S - DEFINED, R - READ)
NONE

LOCAL VARIABLES
IOPT I*4 MENU OPTION SELECTED
IANS I*4 RESPONSE OF YES OR NO
YES I*4 CONSTANT '/Y'/ ANSWER OF YES
NO I*4 CONSTANT '/N'/ ANSWER OF NO

SUBPROGRAMS CALLED
BSTRCT
NETCON
TARCON
EXCON

ERRORS
FATAL NONE
NONFATAL NONE
Figure 3-17. INPUT Logic Flow
### 3.3.2 BSTRCT Module

**SUBROUTINE BSTRCT**

**PURPOSE**

BUILD DATA STRUCTURES REQUIRED BY INPUT

**INPUT ARGUMENTS**

NONE

**OUTPUT ARGUMENTS**

NONE

**DATA STRUCTURES**

(U - USED, S - DEFINED, R - READ)

- TRFND S /LUN/ U
- NPRNDS /PRSNDS/ S
- PPRNDS /PRSNDS/ S
- TRFATT /LUN/ U
- HPATT /HEADRS/ U
- HNOD /HEADRS/ U
- NCATA /INPATT/ S
- PRCATA /INPATT/ S
- NLEVEL /ATT/ U
- NTLEVEL /INPATT/ S
- TATMIL /INPATT/ S
- TRFSBS /LUN/ U
- HPSBS /HEADERS/ U
- NAC /INPSBS/ S
- SCU /INPSBS/ S
- FTU /INPSBS/ S
- CAUMIN /INPSBS/ S
- NSBTAB /INPSBS/ S
- NTAB /SBS/ U
- TSBCAM /INPSBS/ S
- KPBLK /PRSNDS/ S
- KDIR /PRSNDS/ S
- KRNODE /PRSNDS/ S
- MPBLK /INPATT/ S
- MDIR /INPATT/ S
- DRCATA /INPATT/ S
- HPU /HEADRS/ U
- TRFWU /LUN/ U
- LSTACC /PRSWU/ S
- NACC /PRSWU/ S
- NWU /NU/ U
- WUPRS /NU/ S
- MXPRN /ARRYS/ U
- MXCTA /ARRYS/ U
- MXACC /ARRYS/ U
- DEBUG /LUN/ U

3-47
LOCAL VARIABLES
B1,B2,B3 R*4 BASE RATES FOR THREE SCHEDULES OF TERRESTRIAL TARIFF
END I*4 INDEX OF LAST ITEM TO READ IN RECORD
HC I*4 HORIZONTAL COORDINATE
HEADER(2) R*8 FILE HEADER TO IDENTIFY FILE CONTENTS
I I*4 LOOP INDEX
INDX I*4 INDEX OF ARRAY ITEM TO PUT IN DIRECTORY
I1,I2,I3 R*4 INCREMENT CHANGES FOR THREE SCHEDULES OF TERRESTRIAL TARIFF
J I*4 LOOP INDEX
NPREC I*4 NUMBER OF ITEMS FOR RECORD
NREC I*4 NUMBER OF RECORDS
START I*4 INDEX OF FIRST ITEM TO READ IN RECORD
VC I*4 VERTICAL COORDINATES
CODL I*4 LEFT CODE SPECIFIED IN WU PRESTORED TARIFF
CODR I*4 RIGHT CODE SPECIFIED IN WU PRESTORED TARIFF
CTG I*4 RATE TYPE FOR PAIR SPECIFIED IN PRESTORED WU TARIFF
FOUND I*4 FLAG WHICH INDICATES WHICH OF ACCESS PAIR IS FOUND IN LIST OF ACCESS CITIES
0 NEITHER FOUND
1 ONLY CODL FOUND
2 ONLY CODR FOUND
3 BOTH FOUND
LPTR I*4 NUMBER OF ACCESS CITY IN LSTACC LIST
RPTR I*4 NUMBER OF RIGHT ACCESS CITY IN LSTACC LIST
PTR I*4 NUMBER OF PAIR IN WUPRS

SUBPROGRAMS CALLED
PTRPRS
TRFSKP

ERRORS
FATAL
2010 HEADER FROM TRAFFIC NODES NOT AS EXPECTED
2020 HEADER FROM PRESTORED TERRESTRIAL TARIFF NOT AS EXPECTED
2030 ATT TARIFF LEVELS EXCEEDS ARRAY SIZE IN MODEL
2040 HEADER FROM PRESTORED SBS TARIFF NOT AS EXPECTED
2050 SBS TARIFF LEVELS EXCEEDS ARRAY SIZE IN MODEL
2060 HEADER FROM PRESTORED WU TARIFF NOT AS EXPECTED
2070 NUMBER OF ITEMS REQUIRED FOR PRSNDS ARRAY EXCEEDS NUMBER ALLOWED
2080 NUMBER OF ITEMS REQUIRED FOR PRCATA ARRAY EXCEEDS NUMBER ALLOWED
2090 NUMBER OF ITEMS REQUIRED FOR LSTACC ARRAY EXCEEDS NUMBER ALLOWED

NONFATAL
NONE
Figure 3-18. BSTRCT Logic Flow
3.3.3 **EXCON Module**

SUBROUTINE EXCON(CONT)

**PURPOSE**
MAINTAIN EXECUTION CONTROL FILE

**INPUT ARGUMENTS**
NONE

**OUTPUT ARGUMENTS**
CONT LOGICAL INDICATES THAT AN EXECUTION CONTROL FILE WAS SUCCESSFULLY CREATED

**DATA STRUCTURES**
(U - USED, S - DEFINED, R - READ)
DEBUG /LUN/ U

**LOCAL VARIABLES**
IOPT 1*4 MENU OPTION

**SUBPROGRAMS CALLED**
XLIST
XCREAT
DIRGEN

**ERRORS**
FATAL
NONE
NONFATAL
NONE
Figure 3-19. EXCON Logic Flow
3.3.4 XCREAT Module

SUBROUTINE XCREAT(CONT)

PURPOSE
QUERY THE USER AND BUILD FROM THE RESPONSES THE EXECUTION FILE

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
CONT LOGICAL INDICATES THAT AN EXECUTION CONTROL FILE
       WAS SUCCESSFULLY CREATED

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
RUNID /INPEXC/ S
MAXRST /INPEXC/ U
NRSTRT /INPEXC/ S
OPTN /INPEXC/ S
NETFIL /INPEXC/ S
NTRF /INPEXC/ S
FILNAM /INPEXC/ S
MAXTRF /INPEXC/ U
NOUT /INPEXC/ U
OUTCNT /INPEXC/ S
FILES /TARFIL/ U
TARDEF /TARFIL/ U
NTFIL /TARFIL/ U
DEBUG /LUN/ U
LEVEL /INPEXC/ S

LOCAL VARIABLES
NULL R*8 RESPONSE OF <CR> TO REQUEST
TEMP R*4 NUMERIC RESPONSE TO QUERY REQUIRING
       DECIMAL
N I*4 POINTER TO NEXT SLOT AVAILABLE IN ARRAY
       CONTAINING TARIFF NAMES
I I*4 INDEX TO LOOP WHICH ACCUMULATES DATA FOR
       NRSTRT PROBLEMS
M I*4 NUMBER OF TARIFFS SPECIFIED FOR CURRENT
       PROBLEM
J I*4 LOOP INDEX FOR NUMBER OF TARIFFS FOR
       CURRENT PROBLEM
K,L I*4 LOOP INDEX
NTAB I*4 OUTPUT OPTION SUPRESSED
INDEX I*4 POINTER TO TARIFF NAME FOUND IN TARIFF
       DIRECTORY - IF 999 THEN NOT IN DIRECTORY
IFIL I*4 LOOP INDEX USED IN SEARCH OF TARIFF
       DIRECTORY
NETWRK I*4 CONSTANT - NETWORK DIRECTORY
DEFIND I*4 CONSTANT - FILE IS DEFINED
START I*4 BEGINNING OF LIST OF TARIFFS SPECIFIED
       FOR CURRENT PROBLEM WITHIN FILNAM

3-52
END I*4  END OF LIST OF TARIFFS SPECIFIED FOR CURRENT PROBLEM
IOPT I*4  OPTION FOR CURRENT PROBLEM
ITAR I*4  TARIFF TYPE
ERROR LOGICAL  TARIFF FILE SPECIFIED IS ALREADY IN LIST FOR THIS PROBLEM
PRSTAR R*8  CONSTANT - TARIFF NAMES OF PRESTORED TARIFFS
NO I*4  CONSTANT - RESPONSE OF 'N' FOR NO
IY Es I*4  CONSTANT - RESPONSE OF 'Y' FOR YES
IANS I*4  RESPONSE OF YES OR NO
IT I*4  CONSTANT /'T'/ RETURN TO INPUT MENU
IR I*4  CONSTANT /'R'/ REENTER
MAXINT R*4  MAXIMUM INTEGER

SUBPROGRAMS CALLED
FILDEF
NLSTDR
TLSTDR
EXCREA

ERRORS
FATAL
NONE
NONFATAL
NONE

3-53
Figure 3-20. XCREATE Logic Flow (Sheet 1 of 2)
ALL TARIFFS SPECIFIED?

GET NEXT TARIFF

DETERMINE OUTPUT REPORTS DESIRED

A

B

Figure 3-20. XCREATE Logic Flow (Sheet 2 of 2)
3.3.5 EXCREA Module

SUBROUTINE EXCREA

PURPOSE
CREATE THE EXECUTION CONTROL FILE

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HEXEC</td>
<td>/HEADRS/</td>
</tr>
<tr>
<td>NETFIL</td>
<td>/INPEXC/</td>
</tr>
<tr>
<td>FILNAM</td>
<td>/INPEXC/</td>
</tr>
<tr>
<td>RUNID</td>
<td>/INPEXC/</td>
</tr>
<tr>
<td>NRSTRT</td>
<td>/INPEXC/</td>
</tr>
<tr>
<td>NOUT</td>
<td>/INPEXC/</td>
</tr>
<tr>
<td>OPTN</td>
<td>/INPEXC/</td>
</tr>
<tr>
<td>NTRF</td>
<td>/INPEXC/</td>
</tr>
<tr>
<td>OUTCNT</td>
<td>/INPEXC/</td>
</tr>
<tr>
<td>EXCUTE</td>
<td>/LUN/</td>
</tr>
<tr>
<td>DEBUG</td>
<td>/LUN/</td>
</tr>
<tr>
<td>LEVEL</td>
<td>/INPEXC/</td>
</tr>
<tr>
<td>I,J</td>
<td>I*4</td>
</tr>
<tr>
<td>ASTRX</td>
<td>R*8</td>
</tr>
<tr>
<td>ITOT</td>
<td>I*4</td>
</tr>
</tbody>
</table>

LOCAL VARIABLES
BEGIN 
END 

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
NONE
NONFATAL
NONE
Figure 3-21. EXCREA Logic Flow
3.3.6 XLIST Module

SUBROUTINE XLIST

PURPOSE

LIST CURRENT EXECUTION CONTROL FILE

INPUT ARGUMENTS

NONE

OUTPUT ARGUMENTS

NONE

DATA STRUCTURES

(U - USED, S - DEFINED, R - READ)

EXECUTE /LUN/ U
HEXEC /HEADRS/ U
DEBUG /LUN/ U

LOCAL VARIABLES

HEADERS R*8 FILE HEADER
ID R*8 RUN ID
NPROB I*4 NUMBER OF RESTARTS
I,K I*4 LOOP INDICES
FLG(5) I*4 OUTPUT FLAGS
IDX I*4 WHICH OUTPUT OPTION
OPT(2) I*4 CONSTANT OUTPUT OPTIONS
OUTTYP(5) R*5 CONSTANT OUTPUT TYPE
OPTN I*4 RUN TYPE
DESCP(5) C*16 CONSTANT RUN TYPES
FILNAM R*8 FILENAME
NFIL I*4 NUMBER OF TARIFFS IN CURRENT PROBLEM
ASTRX R*8 WORD OF ***** SEPARATING PROBLEM SETS
IANS I*4 ANSWER OF <CR>

SUBPROGRAMS CALLED

NONE

ERRORS

FATAL
NONE
NONFATAL
NONE
Figure 3-22. XLIST Logic Flow
3.4 **NETCON Description**

The NETCON submodule allows the user to specify a network to be analyzed by the CNDC model. It is called by INPUT when the user selects the network construction option. This submodule also provides the user with the capability to manage a set of network files for use in multiple program runs. The user is provided the options of adding files to the set, deleting files from the set, and reviewing the contents of the stored files in the set. Upon entry into this submodule, a menu of options is displayed and the program accepts the user's choice from the following options:

1. Create a network file
2. Delete a network file
3. List a network file or directory
4. Exit from network session

The user may remain in NETCON as long as he likes. After returning from the control of the create, delete, or list options, the initial NETCON menu is redisplayed and any option may again be selected. The exit option must be selected in order to get out of NETCON and pass control back to the INPUT module. Figure 3-23 contains the hierarchy chart of the modules that make up NETCON.

When the user is creating a network he may specify the prestored nodes desired in his network in one of four ways.

1. ALL
2. ALL EXCEPT
3. ONLY
4. PERCENT
The method chosen should be that method that requires the least amount of input. The ALL option creates a network with all of the prestored nodes. The ALL EXCEPT option creates a network with all the prestored nodes except those specified by the user. When using the ONLY option, only those prestored nodes specified will be included. The PERCENT option will build a network from those nodes busiest to least busy that make up the specified percent of the traffic. Once the prestored nodes have been specified in one of the above manners, the user may specify up to 350 additional, non-prestored nodes.
Figure 3-23. NETCON Hierarchy Chart
3.4.1 NETCON Module

SUBROUTINE NETCON

PURPOSE
    MAINTAIN NETWORK FILES FOR USER

INPUT ARGUMENTS
    NONE

OUTPUT ARGUMENTS
    NONE

DATA STRUCTURES
    (U - USED, S - DEFINED, R - READ)
    NONE

LOCAL VARIABLES
    IOPT   I*4       MENU OPTION SELECTED

SUBPROGRAMS CALLED
    NINIT
    CREATE
    DELETE
    LIST
    NCLOSE

ERRORS
    FATAL
    NONE
    NONFATAL
    NONE
Figure 3-24. NETCON Logic Flow
3.4.2 NINIT Module

SUBROUTINE NINIT

PURPOSE
PERFORM INITIALIZATION FOR THE INPUT PROGRAMS NETWORK CONSTRUCTION OPTION

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)

NFILES /NETFIL/ S
NNFIL /NETFIL/ S
NETDEF /NETFIL/ U
DEBUG /LUN/ U
NETDIR /LUN/ U
HNET /HEADRS/ U

LOCAL VARIABLES
HEADER(S) R*8 FILE HEADER
MAXNET I*4 CONSTANT /20/ MAXIMUM NUMBER OF NETWORK FILES EXPECTED IN NETWORK
I I*4 LOOP INDEX

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
3010 FILE HEADER OF NETWORK DIRECTORY NOT AS EXPECTED
3020 NUMBER OF FILES IN DIRECTORY EXCEEDS THE CAPACITY OF THE ARRAYS IN /NETFIL/

NONFATAL
NONE
Figure 3-25. NINIT Logic Flow
3.4.3 CREATE Module

SUBROUTINE CREATE

PURPOSE
QUERY USER ABOUT ENTRIES TO MAKE IN A USER DEFINED NETWORK DEFINITION FILE

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
CROPT /INPNET/ S
NNDT /INPNOD/ S
CODE /INPNOD/ S
NPRNDS /PRSNDT/ U
TOTNDS /INPNET/ S
PERCNT /INPNET/ S
LVLTRF /INPNET/ S
VC /INPNET/ S
HC /INPNET/ S
MAXUSR /INPNET/ U
NTRFC /INPNET/ S
CODEL /INPNOD/ S
CODER /INPNOD/ S
VCIIRS /INPNET/ S
TRAFIC /TRFC/ U
DEBUG /LIN/ U
MAXNDS /INPNOD/ U

LOCAL VARIABLES
FILNAM R*8 NAME OF NETWORK FILE TO BE CREATED
IANS I*4 RESPONSE OF USER AFTER HE HAS ENTERED INVALID RESPONSE TO REQUESTED DATA
IT I*4 CONSTANT /T/ RETURN TO INPUT MENU
IR I*4 CONSTANT /R/ REENTER
NO I*4 CONSTANT /N/ NO
IYES I*4 CONSTANT /Y/ YES
VERIFY L IF USER WANTS RESPONSES ECHOED BACK WITH OPPORTUNITY TO REJECT
ITEMP I*4 TEMPORARY LOCATION OF INTEGER RESPONSES
NULL I*4 CONSTANT <CR>
FOUND L INDICATES IF CODE JUST ENTERED HAS ALREADY BEEN ENTERED
I I*4 LOOP INDEX
TEMP R*4 TEMPORARY LOCATION OF REAL RESPONSE
BIGNUM R*4 CONSTANT LARGE NUMBER
NADD I*4 NUMBER OF ADDITIONAL CITIES ADDED
BEGIN

I*4 START LOCATION WITHIN CODE FOR ADDITIONAL CITIES

IVC I*4 VERTICAL COORDINATE

IHC I*4 HORIZONTAL COORDINATE

NFROM I*4 WHEN REQUESTING INDIVIDUAL TRAFFIC VOLUMES THE FROM CODE

NTO I*4 WHEN REQUESTING INDIVIDUAL TRAFFIC VOLUMES THE TO CODE

NTFC I*4 NUMBER OF VOICE CIRCUITS FORM NFROM TO NTO

TRFOPT L INDICATES IF LISTING OF TRAFFIC TABLE DESIRED

MAXINT R*4 MAXIMUM INTEGER*4

FILSTT I*4 FILE STATUS

1 INVALID NAME

2 UNAVAILABLE

3 IN USE

IFRM I*4 NUMBER OF FROM NODE

ITO I*4 NUMBER OF TO NODE

CVC I*4 NUMBER OF VOICE CIRCUITS BETWEEN IFRM, ITO IN TRAFFIC TABLE

TVC I*4 TOTAL ADDITIONAL TRAFFIC SPECIFIED BETWEEN IFRM AND ITO

IEND I*4 END OF LOOP

SKIP L IF TRUE - DONT ENTER ADDITIONAL TRAFFIC JUST SPECIFIED

SUBPROGRAMS CALLED

FILDEF

SMSA

GETPER

DEFNOD

USRNET

RVTFC

NCREAT

GETTFC

MMPTR

ERRORS

FATAL

NONE

NONFATAL

NONE
GET AND VALIDATE NAME OF FILE TO CREATE

DETERMINE NODE SPECIFICATION OPTION

DEFINE AND VALIDATE PRE-STORED NODES

GET TRAFFIC LEVEL

GET AND VALIDATE ADDITIONAL CITIES

CREATE NETWORK

Figure 3-26. CREATE Logic Flow (Sheet 1 of 2)
Figure 3-26. CREATE Logic Flow (Sheet 2 of 2)
### 3.4.4 GETTFC Module

**SUBROUTINE GETTFC**

**PURPOSE**

BUILD THE TRAFFIC TABLE FOR THE USER DEFINED NETWORK
INCLUDING ONLY THAT TRAFFIC FROM THE PRESTORED TRAFFIC TABLE

**INPUT ARGUMENTS**

NONE

**OUTPUT ARGUMENTS**

NONE

**DATA STRUCTURES**

(U - USED, S - DEFINED, R - READ)

- TOTNDS /INPNET/ U
- LWLTRF /INPNET/ U
- TRFTBL /LUN/ U
- TRAFFIC /TRFC/ S
- HTFC /HEADRS/ U
- NPRNDS /PRSNDS/ U
- RTCNTR /RNODES/ S
- PRNDS /PRSNDS/ U
- DEBUG /LUN/ U

**LOCAL VARIABLES**

- HEADER(2) R*8 HEADER OF THE PRESTORED TRAFFIC TABLE
- NEL I*4 NUMBER OF ELEMENTS PER COLUMN IN TRAFFIC TABLE
- FACT I*4 NORMALIZATION FACTOR
- LSTCOL I*4 LAST COLUMN OF PRESTORED TRAFFIC TABLE READ
- CURCOL I*4 COLUMN WITHIN TRAFFIC TABLE BEING DEFINED
- I,J I*4 LOOP INDICES
- NSKIP I*4 NUMBER OF COLUMNS IN PRESTORED TRAFFIC TABLE TO SKIP TO GET TO NEXT RECORD TO READ
- TRF(350) I*4 NEXT RECORD FROM TRFTBL CONTAINING VALUES OF CURRENT COLUMN FROM TRAFFIC TABLE
- M I*4 NEXT USER DEFINED NODE IN COLUMN
- N I*4 NEXT ITEM IN RECORD
- VCCNV R*4 CONVERSION FACTOR TO COMPUTE NUMBER OF CIRCUITS FOR EACH NODE PAIR
- WHO I*2 TABLE WHICH IDENTIFIES EACH NODE WITHIN THE USER NETWORK WHICH IS IN THE PRESTORED NETWORK BY THE RELATIVE POSITION OF THAT NODE WITHIN THE PRESTORED NETWORK
- CURROW I*4 CURRENT ROW WITHIN TRAFFIC TABLE
SUBPROGRAMS CALLED
TRFSKP
WRTTRF

ERRORS
FATAL
  3030  FILE HEADER OR PRESTORED TRAFFIC TABLE NOT AS EXPECTED
NONFATAL
  NONE
IDENTIFY PRE-STORED NODES IN NETWORK

GET TRAFFIC OF PRESTORED NODES

RETURN

Figure 3-27. GETTFC Logic Flow
3.4.5 GETPER Module

SUBROUTINE GETPER

PURPOSE
BUILD CODE ARRAY OF NODES FROM TRFVOL UP TO PERCNT SPECIFIED

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
PERCNT /INPNET/ U
TRFVOL /LUN/ U
HVOL /HEADRS/ U
CODE /INPNOD/ S
Nnds /INPNOD/ S
TOTNDS /INPNET/ S
DEBUG /LUN/ U

LOCAL VARIABLES
FOUND LOGICAL INDICATES IF PERCENT FOUND IN INPUT TABLE
HEADER(2) R*8 HEADER RECORD READ FROM FILE
NXTNOD I*4 NODE READ FROM TRFVOL
ORDR I*4 POSITION OF NXTNOD IN PRESTORED NETWORK
PRCTG R*8 CUMULATIVE PERCENTAGE OF NETWORK UP TO NXTNOD
NENTRY I*4 NUMBER OF ENTRIES IN TRFVOL TABLE
PCNT R*8 PERCNT NORMALIZED
I I*4 LOOP INDEX

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
3040 FILE HEADER OF TRFVOL NOT AS EXPECTED
3050 PERCNT NOT FOUND IN THE TABLE IN TRFVOL

NONFATAL
NONE
Figure 3-28. GETPER Logic Flow
3.4.6 DEFNOD Module

SUBROUTINE DEFNOD

PURPOSE
BUILD USER DEFINED NETWORK IN RTCNTR ARRAY AND BUILD
NETWORK DIRECTORY

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
TOTNDS /INPNET/ U
NNDS /INPNOD/ U
CODE /INPNOD/ U
CROPT /INPNET/ U
NPRNDS /PRSNDS/ U
PRNDS /PRSNDS/ U
RTCNTR /RNODES/ S
V /RNODES/ S
H /RNODES/ S
VC /INPNET/ U
HC /INPNET/ U
NPBLK /RNODES/ S
NDR /RNODES/ S
DRNODE /RNODES/ S
NNODES /RUN/ S
LVLTRF /INPNET/ U
DEBUG /LUN/ U

LOCAL VARIABLES
IPRS  I*4  POINTER TO WITHIN PRNDS ARRAY
IUSR  I*4  POINTER TO WITHIN ORDR ARRAY
IRTC  I*4  POINTER TO WITHIN RTCNTR ARRAY
PTRUSR  I*4  POINTER TO WITHIN CODE, VC, HC ARRAYS
ORDER (600) I*4  POSITION OF EACH ELEMENT WITHIN CODE
              ARRAY SORTED
ONLY  I*4  CONSTANT /3/ OPTION OF ONLY
PRCNT  I*4  CONSTANT /4/ OPTION OF PERCENT
OMIT  I*4  CONSTANT /2/ OPTION OF ALL EXCEPT
I  I*4  LOOP INDEX
INDX  I*4  WHEN CREATING RTCNTR DIRECTORY NUMBER OF
              RTCNTR ITEM TO GO INTO NEXT POSITION IN
              DIRECTORY

SUBPROGRAMS CALLED
SORT1
ERRORS
FATAL
  3060
NONFATAL
  NONE

TOTNDS VALUE NOT EQUAL TO NUMBER OF ITEMS IN RTCNTR ARRAY
DEFNOD

SORT NODES SPECIFIED IN ALPHABETIC ORDER

BUILD ARRAY OF NODES IN NETWORK

BUILD DIRECTORY OF NODES

RETURN

Figure 3-29. DEFNOD Logic Flow
3.4.7 NCREAT Module

SUBROUTINE NCREAT (FILNAM, TRFOPT)

PURPOSE
CREATE A NEW NETWORK DEFINITION IN SPECIFIED FILE

INPUT ARGUMENTS
FILNAM R*8 NAME OF NETWORK FILE TO CREATE
TRFOPT L PRINT TRAFFIC TABLE OF .TRUE.

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
NNFIL /NETFIL/ U
NFILFIL /NETFIL/ U
NETDEF /NETFIL/ S
LUNNET /LUN/ U
TRFNDS /LUN/ U
UNET /HEADRS/ U
RTCNTR /RNODS/ U
V /RNODS/ U
H /RNODS/ U
LVLTRF /INPNET/ U
CODEL /INPNOD/ U
CODER /INPNOD/ U
TRAFIC /TRFC/ S
VCIRTS /LUN/ U
TOTNDS /INPNET/ U
NTRFDC /INPNET/ U
DEBUG /LUN/ U

LOCAL VARIABLES
I I*4 LOOP INDEX
FOUND L FILNAM FOUND IN DIRECTORY
UNIT I*4 LOGICAL UNIT OF NETWORK FILE TO BE CREATED
EMPTY I*4 CONSTANT /0/ FILE UNDEFINED
PCODE I*4 CODE OF RATE CENTER READ FROM TRFNDS
PV I*4 VERTICAL COORDINATE OF RATE CENTER READ FROM TRFNDS
PH I*4 HORIZONTAL COORDINATE OF CENTER READ FROM TRFNDS
ORDR(600) I*4 ORDER OF SORTED CODE ARRAY
FROM I*4 FROM NODE WHEN SPECIFYING TRAFFIC
TO I*4 TO NODE WHEN SPECIFYING TRAFFIC
SPOOL I*4 LOGICAL UNIT NUMBER TO SPOOL TRAFFIC TABLE FOR THIS USER DEFINED NETWORK
NEXT I*4 NEXT ENTRY OF TRAFFIC LEVELS IN ALPHABETIC ORDER
NETNUM I*4 NUMBER OF NETWORK FILE
C1,C2,C3 I*2 CHARACTERS THAT MAKE up NETWORK FILE NUMBER

SUBPROGRAMS CALLED
TRFSKP
SORT2
MNMPTR
WRTTRF
NCLOSE
INTCHR

ERRORS
FATAL
3070 FILE SPECIFIED FOR NETWORK FILE ALREADY DEFINED
3080 FILE SPECIFIED FOR NETWORK FILE NOT IN DIRECTORY
NONFATAL
NONE
Figure 3-30. NCREAT Logic Flow
3.4.8 DELETE Module

SUBROUTINE DELETE

PURPOSE
DELETE A NETWORK FILE

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
DEBUG     /LUN/    U

LOCAL VARIABLES
FILNAM     R*8      NAME OF FILE TO DELETE
IANS       I*4      USER RESPONSE
IYES       I*4      CONSTANT /'Y'/
NO          I*4      CONSTANT /'N'/
USED        I*4      CONSTANT /3/ FILE IN USE

SUBPROGRAMS CALLED
FILDEF
NDELETE

ERRORS
FATAL
NONE
NONFATAL
NONE
Figure 3-31. DELETE Logic Flow
3.4.9 NDELET Module

SUBROUTINE NDELET(FILNAM)

PURPOSE
DELETE THE NETWORK FILE SPECIFIED

INPUT ARGUMENTS
FILNAM   R*8    NAME OF NETWORK FILE TO BE DELETED

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
NNFIL    /NETFIL/   U
NFFILES  /NETFIL/   U
NETDEF   /NETFIL/   U
DEBUG    /LUN/      U

LOCAL VARIABLES
I       I*4      LOOP INDEX
FOUND   L         INDICATES IF FILNAM FOUND IN DIRECTORY

SUBPROGRAMS CALLED
NCLOSE

ERRORS
FATAL
3090    FILE NAME TO DELETE NOT IN DIRECTORY

NONFATAL
3510    FILE NAME TO DELETE HAS ALREADY BEEN DELETED
Figure 3-32. NDELETE Logic Flow
3.4.10 LIST Module

SUBROUTINE LIST

PURPOSE
LIST A NETWORK FILE OR THE NETWORK DIRECTORY ON SYSOUT

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
DEBUG /LUN/ U

LOCAL VARIABLES
FILNAM R*8 NAME OF NETWORK FILE TO LIST
NULL R*8 CONSTANT /' '/ INDICATES <CR>
RESPONSE
USED I*4 CONSTANT /3/ FILE DEFINED

SUBPROGRAMS CALLED
NLSTDR
NLSTFL
FILDEF

ERRORS
FATAL NONE
NONFATAL NONE
Figure 3-33. LIST Logic Flow
3.4.11 NLSTDR Module

SUBROUTINE NLSTDR

PURPOSE
LIST THE NETWORK DIRECTORY

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
NNFIL /NETFIL/ U
NETDEF /NETFIL/ U
NFILES /NETFIL/ U
DEBUG /LUN/ U

LOCAL VARIABLES
I I*8 LOOP INDEX
STDES R*8 ALPHA DESCRIPTION OF FILE STATUS
USED R*8 CONSTANT - CHARACTER STRING 'UNUSED'
IANS I*4 ANSWER OF <CR>

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL NONE
NONFATAL NONE
Figure 3-34. NLSTDR Logic Flow
3.4.12 NLSTFL Module

SUBROUTINE NLSTFL (FILNAM)

PURPOSE
LIST THE CONTENTS OF THE NETWORK FILE SPECIFIED

INPUT ARGUMENTS
FILNAM R*8 FILE TO BE LISTED

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)

<table>
<thead>
<tr>
<th>DATA STRUCTURE</th>
<th>SPECIFICATION</th>
<th>TYPE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>NNFIL</td>
<td>/NETFIL/</td>
<td>U</td>
<td></td>
</tr>
<tr>
<td>NFILES</td>
<td>/NETFIL/</td>
<td>U</td>
<td></td>
</tr>
<tr>
<td>NETDEF</td>
<td>/NETFIL/</td>
<td>U</td>
<td></td>
</tr>
<tr>
<td>LUNNET</td>
<td>/LUN/</td>
<td>U</td>
<td></td>
</tr>
<tr>
<td>UNET</td>
<td>/HEADRS/</td>
<td>U</td>
<td></td>
</tr>
<tr>
<td>DEBUG</td>
<td>/LUN/</td>
<td>U</td>
<td></td>
</tr>
</tbody>
</table>

LOCAL VARIABLES

<table>
<thead>
<tr>
<th>LOCAL VARIABLE</th>
<th>TYPE</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>HEADER(2)</td>
<td>R*8</td>
<td>HEADER RECORD OF FILE</td>
</tr>
<tr>
<td>FOUND</td>
<td>LOGICAL</td>
<td>FILNAM FOUND IN DIRECTORY</td>
</tr>
<tr>
<td>CNODES</td>
<td>I*4</td>
<td>NUMBER OF NODES SPECIFIED IN THIS NETWORK</td>
</tr>
<tr>
<td>CODE</td>
<td>I*4</td>
<td>UNIQUE FOUR CHARACTER CODE FOR EACH NODE IN THE NETWORK</td>
</tr>
<tr>
<td>CODEL</td>
<td>I*4</td>
<td>FOUR CHARACTER CODE OF NODE WHERE TRAFFIC ORIGINATES</td>
</tr>
<tr>
<td>CODER</td>
<td>I*4</td>
<td>FOUR CHARACTER CODE OF NODE WHERE TRAFFIC TERMINATES</td>
</tr>
<tr>
<td>HC</td>
<td>I*4</td>
<td>HORIZONTAL COORDINATE OF NODE</td>
</tr>
<tr>
<td>I,J</td>
<td>I*4</td>
<td>LOOP INDICES</td>
</tr>
<tr>
<td>NTRFC</td>
<td>I*4</td>
<td>NUMBER OF TRAFFIC TABLE ADDITIONS SPECIFIED IN FILE</td>
</tr>
<tr>
<td>TRFLVL</td>
<td>I*4</td>
<td>TRAFFIC LEVEL</td>
</tr>
<tr>
<td>UNIT</td>
<td>I*4</td>
<td>LOGICAL UNIT NUMBER OF NETWORK FILE</td>
</tr>
<tr>
<td>VC</td>
<td>I*4</td>
<td>VERTICAL COORDINATE OF NODE</td>
</tr>
<tr>
<td>VCIROS</td>
<td>I*4</td>
<td>AMOUNT OF TRAFFIC FROM CODEL TO CODER</td>
</tr>
<tr>
<td>EMPTY</td>
<td>I*4</td>
<td>CONSTANT /0/ FILE UNDEFINED</td>
</tr>
<tr>
<td>LPCRNR</td>
<td>I*4</td>
<td>CONSTANT /20/ LINES TO PRINT PER SCREEN</td>
</tr>
<tr>
<td>IANS</td>
<td>I*4</td>
<td>RESPONSE OF &lt;CR&gt; TO CONTINUE PRINTING FILE</td>
</tr>
</tbody>
</table>

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
- 3100 FILE SPECIFIED TO BE PRINTED NOT IN DIRECTORY
- 3110 HEADER RECORD OF FILE NOT AS EXPECTED
NONFATAL
- 3520 FILE SPECIFIED TO BE PRINTED NOT DEFINED
Figure 3-35. NLSTFL Logic Flow
3.4.13 NCLOSE Module

SUBROUTINE NCLOSE

PURPOSE
REPLACE NETWORK DIRECTORY WHEN CONTENTS OF DIRECTORY HAS CHANGED

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
NNFIL /NETFIL/ U
NFILES /NETFIL/ U
NETDEF /NETFIL/ U
HNET /HEADRS/ U
NETDIR /LUN/ U
DEBUG /LUN/ U

LOCAL VARIABLES
I I*4 LOOP INDEX

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
NONE
NONFATAL
NONE
Figure 3-36. NCLOSE Logic Flow
3.5 TARCON DESCRIPTION

The TARCON submodule allows the user to maintain a set of tariff files which can be specified as input to the program. This module is called directly from INPUT. TARCON organized existing tariff files into a directory - all user defined tariff files. This module does not provide the user access to the prestored AT&T, WU, or SBS tariff files. It does, however, provide the capability for the user to make alterations to those prestored tariff files and store the resulting structures as a user defined tariff. The user can manage his defined tariffs by creating new files, deleting old files, and listing the contents of any file and/or the directory itself. The user may specify files contained in the directory as input during the EXCON session.

Upon entry into TARCON, a menu of options will be displayed and the user will be asked to select one of:

1. Create a tariff file
2. Delete a tariff file
3. List a tariff file or the directory
4. Exit TARCON session

The hierarchy chart of the modules that make up TARCON is contained in Figure 3-37.
Figure 3-37. TARCON Hierarchy Chart
3.5.1 **TARCON Module**

SUBROUTINE TARCON

**PURPOSE**
MAINTAIN TARIFF FILES FOR USER

**INPUT ARGUMENTS**
NONE

**OUTPUT ARGUMENTS**
NONE

**DATA STRUCTURES**
(U - USED, S - DEFINED, R - READ)
DEBUG /LUN/ U`

**LOCAL VARIABLES**
IOPT I*4 MENU OPTION SELECTED

**SUBPROGRAMS CALLED**
TINIT
CREATT
DELETT
LISIT
TCLOSE

**ERRORS**
FATAL
NONE
NONFATAL
NONE
Figure 3-38. TARCON Logic Flow
3.5.2  TINIT Module

SUBROUTINE TINIT

PURPOSE
  PERFORM INITIALIZATION FOR THE INPUT PROGRAMS TARIFF CONSTRUCTION
  OPTION

INPUT ARGUMENTS
  NONE

OUTPUT ARGUMENTS
  NONE

DATA STRUCTURES
  (U - USED, S - DEFINED, R - READ)
  TFILES /TARFIL/  S
  NTfil /TARFIL/  S
  TARDEF /TARFIL/  S
  TRFDIR /LUN/  U
  HTAR /HEADRS/  U
  DEBUG /LUN/  U

LOCAL VARIABLES
  HEADER(2) R*8  FILE HEADER
  MAXTAR I*4  CONSTANT /13/  MAXIMUM NUMBER OF TARIFF FILES EXPECTED IN DIRECTORY
  I  I*4  LOOP INDEX

SUBPROGRAMS CALLED
  NONE

ERRORS
  FATAL
  4010  FILE HEADER OF TARIFF DIRECTORY NOT AS EXPECTED
  4020  NUMBER OF FILES IN DIRECTORY EXCEEDS THE CAPACITY OF THE ARRAYS IN /TARFIL/

NONFATAL
  NONE
Figure 3-39. TINIT Logic Flow
3.5.3 CREAT Module

SUBROUTINE CREAT

PURPOSE
CREATE A TARIFF FILE ACCORDING TO USER SPECIFICATIONS

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
DEBUG /LUN/ U

LOCAL VARIABLES
FILNAM R*8 NAME OF FILE TO CREATE
IANSE I*4 USER RESPONSE
IR I*4 CONSTANT '/R'/ USER OPTION TO REENTER
IT I*4 CONSTANT '/T'/ USER OPTION TO TERMINATE
TYPE I*4 TYPE OF TARIFF TO CREATE (1-ATT, 2-WU, 3-SBS)
FILSTT I*4 FILE STATUS
1 INVALID NAME
2 AVAILABLE
3 IN USE

SUBPROGRAMS CALLED
CRATT
CRSBS
FILDEF
TCREAT

ERRORS
FATAL
NONE
NONFATAL
NONE
Figure 3-40. CREATT Logic Flow
3.5.4 CRATT Module

SUBROUTINE CRATT(*)

PURPOSE
CREATE A TARIFF BASED ON THE AT&T PHILOSOPHY

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)

<table>
<thead>
<tr>
<th>Variable</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NNDS</td>
<td>/INPNOD/</td>
<td>S</td>
</tr>
<tr>
<td>CODE</td>
<td>/INPNOD/</td>
<td>S</td>
</tr>
<tr>
<td>ADJ</td>
<td>/INPATT/</td>
<td>S</td>
</tr>
<tr>
<td>INCSC</td>
<td>/INPATT/</td>
<td>S</td>
</tr>
<tr>
<td>TATMIL</td>
<td>/INPATT/</td>
<td>U</td>
</tr>
<tr>
<td>DEBUG</td>
<td>/LUN/</td>
<td>U</td>
</tr>
<tr>
<td>NTLEVEL</td>
<td>/INPATT/</td>
<td>U</td>
</tr>
</tbody>
</table>

LOCAL VARIABLES

<table>
<thead>
<tr>
<th>Variable</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ITEMP</td>
<td>I*4</td>
<td>BUFFERS FOR USER INPUTS</td>
</tr>
<tr>
<td>JTEMP</td>
<td>I*4</td>
<td>USER RESPONSE Y OR N</td>
</tr>
<tr>
<td>IANS</td>
<td>I*4</td>
<td>USER RESPONSE TO REENTER</td>
</tr>
<tr>
<td>IR</td>
<td>I*4</td>
<td>CONSTANT /'R'/ USER RESPONSE TO REENTER</td>
</tr>
<tr>
<td>IT</td>
<td>I*4</td>
<td>USER RESPONSE TO TERMINATE</td>
</tr>
<tr>
<td>J</td>
<td>I*4</td>
<td>LOOP INDEX</td>
</tr>
<tr>
<td>MINUS</td>
<td>I*4</td>
<td>CONSTANT /'-'/ INDICATES DELETION</td>
</tr>
<tr>
<td>N</td>
<td>I*4</td>
<td>FINAL LOOP INDEX</td>
</tr>
<tr>
<td>NPLUS</td>
<td>I*4</td>
<td>CONSTANT /'+'/ INDICATES INSERTION</td>
</tr>
<tr>
<td>NULL</td>
<td>I*4</td>
<td>CONSTANT /'/ INDICATES &lt;CR&gt; RESPONSE</td>
</tr>
<tr>
<td>INLIST</td>
<td>L</td>
<td>CODE ENTERED IS PRESTORED CATA</td>
</tr>
<tr>
<td>INCLUD</td>
<td>L</td>
<td>USER REQUEST TO ADD CODE AS CATA</td>
</tr>
<tr>
<td>EXCLUD</td>
<td>L</td>
<td>USER REQUEST TO EXCLUDE CODE FROM CATA</td>
</tr>
<tr>
<td>FOUND</td>
<td>L</td>
<td>CODE ALREADY IN ADJUSTMENT LIST</td>
</tr>
<tr>
<td>I</td>
<td>I*4</td>
<td>LOOP INDEX</td>
</tr>
</tbody>
</table>

SUBPROGRAMS CALLED

ATTLST

ERRORS

<table>
<thead>
<tr>
<th>Level</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FATAL</td>
<td>NONE</td>
</tr>
<tr>
<td>NONFATAL</td>
<td>NONE</td>
</tr>
</tbody>
</table>
GET ADDITIONS OR DELETIONS OF CAT-A STATUS

GET INCREMENT CHARGES FOR ALL SCHEDULES

RETURN
3.5.5 CRWU Module

SUBROUTINE CRWU(*)

PURPOSE
CREATE A TARIFF BASED ON THE WU PHILOSOPHY

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U' - USED, S - DEFINED, R - READ)
NADJ   /INPWU/   S
CODEL  /INPNOD/  S
CODER  /INPNOD/  S
CAT    /INPWU/   S
CHGLH  /INPWU/   S
CHGMH  /INPWU/   S
CHGSH  /INPWU/   S
DEBUG  /LUN/     U

LOCAL VARIABLES
ITEMP,JTEMP,KTEMP   I*4   BUFFER FOR USER INPUTS
IANS                I*4   USER RESPONSE Y OR N
IR                  I*4   CONSTANT '/R'/ USER CHOICE TO REENTER
IT                  I*4   CONSTANT '/T'/ USER CHOICE TO TERMINATE
NULL                I*4   CONSTANT '/ ' / INDICATES <CR> RESPONSE
FOUND               L     DUPLICATE ENTRY
I                  T*4   LOOP INDEX
PAIR               I*4   NODES SPECIFIED ARE A SATELLITE ACCESS CITY PAIR IN PRESTORED TARIFF
ADD                L     PAIR SPECIFIED SHOULD BE ADDED TO LIST OF ACCESS CITY PAIRS
DELETE             L     PAIR SPECIFIED SHOULD BE DELETED FROM LIST OF ACCESS CITY PAIRS

SUBPROGRAMS CALLED
WUPAIR

ERRORS
FATAL
NONE
NONFATAL
NONE

3-104
Figure 3-42. CRWU Logic Flow
3.5.6 CRSBS Module

SUBROUTINE CRSBS

PURPOSE
CREATE A TARIFF BASED ON THE SBS PHILOSOPHY

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
NAC  /INPSBS/ S
SCU  /INPSBS/ S
FTU  /INPSBS/ S
CAUMIN /INPSBS/ S
INCSB /INPSBS/ S
NSBTAB /INPSBS/ U
TSBCAM /INPSBS/ U
NACNMB /SBSMXD/ S
NACLOC /SBSMXD/ S
DEBUG /LUN/ U

LOCAL VARIABLES
I   I*4   LOOP INDEX
N   I*4   ONE LESS THAN THE NUMBER OF ITEMS IN THE
         CAU RATE TABLE
YES I*4   CONSTANT /'Y'/ RESPONSE OF YES
NO  I*4   CONSTANT /'N'/ RESPONSE OF NO
IANS I*4   RESPONSE TO YES/NO QUESTION
LOC I*4   NAC LOCATION AS SPECIFIED BY USER
NULL I*4   CONSTANT /' '/ RESPONSE OF <CR>
FOUND L    INDICATES IF NAC LOCATION ENTERED
            IS DUPLICATE ENTRY
FINISH L    INDICATES THAT USER COMPLETED ENTERING
            NAC LOCATIONS

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
NONE
NONFATAL
NONE
Figure 3-43. CRSBS Logic Flow
3.5.7  TCREAT Module

SUBROUTINE TCREAT(FILNAM, TYPE)

PURPOSE
CREATE NEW TARIFF DEFINITION IN SPECIFIED FILE

INPUT ARGUMENTS
FILNAM    R*8    NAME OF FILE TO CONTAIN NEW TARIFF
TYPE      I*4    TYPE OF FILE TO CREATE
           1   ATT
           2   WU
           3   SBS

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
NTFIL    /TARFIL/    U
TFILES   /TARFIL/    U
TARDEF   /TARFIL/    S
LUNTRF   /LUN/      U
DEBUG    /LUN/      U

LOCAL VARIABLES
FOUND    L    FILENAM FOUND IN LIST
LSTPTR   I*4    POSITION OF FILENAM IN LIST
UNIT     I*4    LOGICAL UNIT NUMBER OF FILE CREATED
I        I*4    LOOP INDEX
EMPTY    I*4    CONSTANT /0/  FILE UNDEFINED

SUBPROGRAMS CALLED
TCRATT
TCRWU
TCRSBS
TCLOSE

ERRORS
FATAL
  4030   FILE NAME NOT IN DIRECTORY
  4040   FILE SPECIFIED ALREADY CREATED
  4050   FILE TYPE INVALID
NONFATAL
  NONE
Figure 3-44. TCREAT Logic Flow
3.5.8 TCRATT Module

SUBROUTINE TCRATT(FILNAM, UNIT)

PURPOSE
CREATE ATT TARIFF DEFINITION AT SPECIFIED FILE

INPUT ARGUMENTS
FILNAM  R*8  NAME OF ATT TARIFF FILE TO BE CREATED
UNIT     I*4  LOGICAL UNIT NUMBER OF FILE TO BE CREATED

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
UTAR     /HEADRS/  U
NNDS     /INPNOD/  U
ADJ      /INPATT/  U
CODE     /INPNOD/  U
NTLEVEL  /INPATT/  U
TATMIL   /INPATT/  U
INCSC    /INPATT/  U
DEBUG    /LUN/    U

LOCAL VARIABLES
BASE1    R*4  BASE COST FOR SCHEDULE 1
BASE2    R*4  BASE COST FOR SCHEDULE 2
BASE3    R*4  BASE COST FOR SCHEDULE 3
I        I*4  LOOP INDEX
ORDR(600) I*4  INDEX TO RATE CENTER CODES IN CODE IN ALPHABETIC ORDER
PRVMLG   I*4  PREVIOUS MILEAGE BREAKPOINT USED IN COMPUTATION OF BASE COSTS
PTR      I*4  INDEX OF NEXT CODE TO BE PRINTED

SUBPROGRAMS CALLED
SORT1

ERRORS
FATAL
NONE
NONFATAL
NONE
Figure 3-45. TCRATT Logic Flow
3.5.9 **TCRWU Module**

**SUBROUTINE TCRWU(FILNAM,UNIT)**

**PURPOSE**
CREATE WU TARIFF DEFINITION IN SPECIFIED FILE

**INPUT ARGUMENTS**
- **FILNAM** R*8 NAME OF WU FILE TO BE CREATED
- **UNIT** I*4 LOGICAL UNIT NUMBER OF WU FILE BEING CREATED

**OUTPUT ARGUMENTS**
NONE

**DATA STRUCTURES**
(U - USED, S - DEFINED, R - READ)
- **UTAR** /HEADRS/ U
- **NADJ** /INPWU/ U
- **CODEL** /INPNOD/ U
- **CODER** /INPNOD/ U
- **CAT** /INPWU/ U
- **CHGLH** /INPWU/ U
- **CHGNH** /INPWU/ U
- **CHGSH** /INPWU/ U
- **DEBUG** /LUN/ U

**LOCAL VARIABLES**
- **ORDR(600)** I*4 ORDER OF CODEL AND CODER ARRAYS IN ALPHABETIC ORDER BY CODEL AND THEN BY CODER WITHIN CODEL
- **PTR** I*4 POINTER TO WITHIN CODEL, CODER, AND CAT OF NEXT ACCESS CITY PAIR IN ALPHABETIC ORDER
- **I** I*4 NEXT ACCESS CITY PAIR IN ALPHABETIC ORDER LOOP INDEX

**SUBPROGRAMS CALLED**
SORT2

**ERRORS**
- **FATAL**
- **NONE**
- **NONFATAL**
- **NONE**
Figure 3-46. TCRWU Logic Flow
3.5.10 TCRSBS Module

SUBROUTINE TCRSBS (FILNAM, UNIT)

PURPOSE
CREATE WU TARIFF DEFINITION IN SPECIFIED FILE

INPUT ARGUMENTS
FILNAM  R*8  NAME OF WU TARIFF TO CREATE
UNIT    I*4  LOGICAL UNIT NUMBER OF THE FILE TO BE CREATED

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
    UTAR   /HEADRS/   U
    NAC    /INPSBS/   U
    SCU    /INPSBS/   U
    FTU    /INPSBS/   U
    CAUMIN /INPSBS/   U
    NSBTAB /INPSBS/   U
    TSBCAM /INPSBS/   U
    INCNSB /INPSBS/   U
    NACNMB /SBSMXD/   U
    NACLOC /SBSMXD/   U
    DEBUG  /LUN/     U

LOCAL VARIABLES
    ORDR(600)   I*4  POSITION OF EACH NAC LOCATION IN ALPHABETIC ORDER
    PTR         I*4  POSITION WITHIN NACLOC OF NEXT NAC
    I,J         I*4  LOOP INDICES
    RECORD(16)  I*4  NEXT SET OF NAC LOCATIONS TO WRITE TO FILE
    NREC        I*4  NUMBER OF RECORDS OF NAC LOCATIONS TO WRITE TO FILE
    BEGIN       I*4  INDEX TO FIRST NAC IN THIS RECORD
    END         I*4  INDEX TO LAST NAC IN THIS RECORD
    NEXT        I*4  CURRENT ELEMENT IN RECORD BEING FILLED
    BASE        R*4  BASE CAU CHARGE
    PRVCNT      I*4  PREVIOUS CAU BREAKPOINT

SUBPROGRAMS CALLED
    SORT1

ERRORS
    FATAL      NONE
    NONFATAL   NONE
Figure 3-47. TCRSBS Logic Flow
3.5.11 DELETT Module

SUBROUTINE DELETT

PURPOSE
DELETE A TARIFF FILE

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
DEBUG /LUN/ U

LOCAL VARIABLES
FILNAM R*8 NAME OF FILE TO DELETE
IANS I*4 USER RESPONSE Y OR N
IYES I*4 CONSTANT /'Y'/
NO I*4 CONSTANT /'N'/

SUBPROGRAMS CALLED
TDELET
FILDEF

ERRORS
FATAL
NONE
NONFATAL
NONE
Figure 3-48. DELETT Logic Flow
SUBROUTINE TDELET(FILNAM)

PURPOSE
DELETE THE TARIFF FILE SPECIFIED

INPUT ARGUMENTS
FILNAM R*8 NAME OF NETWORK FILE TO BE DELETED

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
NTFIL /TARFIL/ U
TFILES /TARFIL/ U,
TARDEF /TARFIL/ U
DEBUG /LUN/ U

LOCAL VARIABLES
  I  I*4 LOOP INDEX
  FOUND L INDICATES IF FILNAM FOUND IN DIRECTORY

SUBPROGRAMS CALLED
  TCLOSE

ERRORS
  FATAL
    4060 FILE NAME TO DELETE NOT IN DIRECTORY
  NONFATAL
    4510 FILE NAME TO DELETE HAS ALREADY BEEN DELETED
Figure 3-49. TDELET Logic Flow
3.5.13 LISTT Module

SUBROUTINE LISTT

PURPOSE
LIST OF NETWORK FILE OR THE NETWORK DIRECTORY ON SYSOUT

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
DEBUG /LUN/ U

LOCAL VARIABLES
FILNAM R*8 NAME OF FILE TO LIST
NULL R*8 CONSTANT '/ ' INDICATES <CR> RESPONSE

SUBPROGRAMS CALLED
TLSTD
TLSTFL
FILDEF

ERRORS
FATAL
NONE
NONFATAL
NONE
Figure 3-50. LISTT Logic Flow
3.5.14 TLSTD Module

SUBROUTINE TLSTD

PURPOSE
LIST THE TARIFF DIRECTORY

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
- NTFIL /TARFIL/ U
- TARDEF /TARFIL/ U
- TFILES /TARFIL/ U
- DEBUG /LUN/ U

LOCAL VARIABLES
I I*8 LOOP INDEX
J I*4 FILE STATUS
STDES R*8 ALPHA DESCRIPTION OF FILE STATUS
AVAIL R*8 CONSTANT - CHARACTER STRING 'UNUSED'
TYPE(3) R*8 CONSTANT - TARIFF TYPE
IANS I*4 ANSWER OF <CR>

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
NONE
NONFATAL
NONE
Figure 3-51. TLSTDR Logic Flow
3.5.15 TLSTFL Module

SUBROUTINE TLSTFL(FILNAM)

PURPOSE
LIST TARIFF FILE SPECIFIED

INPUT ARGUMENTS
FILNAME R*8 NAME OF FILE TO BE LISTED

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
NTFIL /TARFIL/ U
TFILES /TARFIL/ U
TARDEF /TARFIL/ U
LUNTRF /LUN/ U
DEBUG /LUN/ U

LOCAL VARIABLES
FOUND L FILE NAME IN LIST
LSTPTR I*4 POSITION OF FILE NAME IN LIST
UNIT I*4 LOGICAL UNIT NUMBER OF FILE TO BE LISTED
I I*4 LOOP INDEX
EMPTY I*4 CONSTANT /0/ FILE UNDEFINED
TYPE I*4 TYPE OF FILE TO BE LISTED

SUBPROGRAMS CALLED
TLSTAT
TLSTWU
TLSTSB

ERRORS
FATAL
4070 FILE NAME NOT IN DIRECTORY
4080 FILE EMPTY
4090 FILE TYPE INVALID

NONFATAL
NONE
Figure 3-52. TLSTFL Logic Flow
3.5.16 TLSTAT Module

SUBROUTINE TLSTAT(FILNAM,UNIT)

PURPOSE
LIST FILE WHICH DESCRIBES ATT TARIFF

INPUT ARGUMENTS
FILNAM  R*8  ATT FILE TO BE LISTED
UNIT    I*4  LOGICAL UNIT NUMBER OF FILE TO BE LISTED

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
UTAR    /HEADRS/  U
DEBUG   /LUN/    U

LOCAL VARIABLES
LPSCRN  I*4  CONSTANT /20/ LINES PER SCREEN
TYPADJ  I*4  ADJUSTMENT TYPE
DESDADJ(2) R*8  CONSTANT /'LISTED ', 'UNLISTED'/
IANS    I*4  SIGNAL TO CONTINUE PRINT
INC(3)  R*4  INCREMENT FOR EACH SCHEDULE
BAS(3)  R*4  BASE CHARGE FOR EACH SCHEDULE
MLG     I*4  MILEAGE INCREMENT
CODE    I*4  CODE FOR CATA ADJUSTMENT
ADJ     I*4  NATURE OF ADJUSTMENT
NADJ    I*4  NUMBER OF ADJUSTMENTS SPECIFIED
NLVL    I*4  NUMBER OF MILEAGE BREAKPOINTS
HEADER(2) R*8  HEADER RECORD OF TARIFF FILE
I,J     I*4  LOOP INDICES
LABL(2) R*8  LABEL FOR RATE SCHEDULE
DUMY    I*4  STAND IN FOR MLG IN LAST RECORD OF ATT TARIFF

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
4100  FILE HEADER NOT AS EXPECTED
NONFATAL
NONE
Figure 3-53. TLSTAT Logic Flow
3.5.17 TLSTWU Module

SUBROUTINE TLSTWU(FILNAM,UNIT)

PURPOSE
LIST FILE WHICH DESCRIBES WU TARIFF

INPUT ARGUMENTS
FILNAM      R*8      NAME OF FILE TO BE LISTED
UNIT        I*4      LOGICAL UNIT NUMBER OF FILE TO BE LISTED

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
UTAR        /HEADRS/  U
DEBUG       /LUN/     U

LOCAL VARIABLES
LPSCRN      I*4      LINES TO PRINT PER SCREEN
TYPADJ      I*4      TYPE OF ADJUSTMENT
               1  EXCLUDE
               2  LONG
               3  MEDIUM
               4  SHORT
DESADJ      R*8      DESCRIPTION OF ADJUSTMENT TYPE
HEADER(2)   R*8      HEADER RECORD
NREC        I*4      NUMBER OF RECORDS WHICH CONTAIN ADJUSTMENTS TO SATELLITE ACCESS CITY STATUS
I,J         I*4      LOOP INDICES
CL          I*4      LEFT MOST CODE OF PAIR
CR          I*4      RIGHT MOST CODE OF PAIR
CTG         I*4      RATE TYPE OF PAIR
IANS        I*4      ANSWER OF <CR>
RATE        R*4      WU RATE

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
4110      HEADER READ FROM WU TARIFF NOT AS EXPECTED
NONFATAL
NONE
Figure 3-54. TLSTWU Logic Flow
3.5.18 TLSTSB Module

SUBROUTINE TLSTSB(FILNAM,UNIT)

PURPOSE
LIST FILE WHICH DESCRIBES SBS TARIFF.

INPUT ARGUMENTS
FILNAM R*8 NAME OF SBS TARIFF TO BE PRINTED
UNIT I*4 LOGICAL UNIT NUMBER FILE TO BE PRINTED

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
UTAR /HEADRS/ U
DEBUG /LUN/ U

LOCAL VARIABLES
HEADER(2) R*8 HEADER RECORD READ
LPSCRN I*4 CONSTANT /20/ LINES PER SCREEN
NREC I*4 NUMBER OF RECORDS CONTAINING NAC SPECIFICATIONS
NENTRY I*4 NUMBER OF NAC LOCATIONS SPECIFIED
RECORD(16) I*4 NEXT RECORD OF NAC LOCATIONS READ FROM TARIFF
END I*4 NUMBER OF ITEMS IN NEXT RECORD
IANS I*4 RESPONSE OF <CR>
BKP I*4 NEXT CAU BREAKPOINT
NTAB I*4 NUMBER OF ITEMS IN CAU RATE TABLE
BASE R*4 BASE RATE FOR CAU
INC R*4 INCREMENT CHARGE FOR CAU
LABL(2) R*8 LABEL FOR CAU TABLE
DUNY I*4 STAND FOR BKP ON FINAL RECORD OF CAU TABLE
IEND I*4 NUMBER OF TARIFF LEVELS TO PRINT WITH LABL(1)
I,J,K I*4 LOOP INDICES
NAC R*4 NAC CHARGE
SCU R*4 FTU CHARGE
CAUMIN R*4 MIN CAU CHARGE

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
4120 FILE HEADER READ FROM WU TARIFF NOT AS EXPECTED
NONFATAL
NONE
Figure 3-55. TLSTSB Logic Flow
3.5.19 TCLOSE Module

SUBROUTINE TCLOSE

PURPOSE
REPLACE TARIFF DIRECTORY WHEN CONTENTS OF DIRECTORY HAS CHANGED

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES

(U - USED, S - DEFINED, R - READ)
NTFIL /TARFIL/ U
TFILES /TARFIL/ U
TARDEF /TARFIL/ U
HTAR /HEADRS/ U
TRFDIR /LUN/ U
DEBUG /LUN/ U

LOCAL VARIABLES
I I*4 LOOP INDEX

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
NONE
NONFATAL
NONE
Figure 3-56. TCLOSE Logic Flow
3.6 INIT DESCRIPTION

The INIT module reads in turn the specifications for each problem from the Execution Control File. From these specifications it builds the data structures (network definition, traffic table, and tariffs) required for the specific CNDC problem, validating the input as it builds. If no error is encountered, INIT returns to the executive module, CNDC, and from there the appropriate optimization modules are initiated. If an error is encountered in the building of the data structures, INIT discontinues with the current problem description, locates within the Execution Control File the next problem description, and again attempts to build the required data structures, again validating the input in the process. INIT returns to CNDC whenever it has successfully constructed the data for a problem or when it has reached the end of the Execution Control File and the run is to terminate.

There are thirteen submodules which make up INIT. Figure 3-57 shows the hierarchical relationship of the thirteen submodules.
Figure 3-57. INIT Hierarchy Chart
3.6.1 INIT Module

```
SUBROUTINE INIT(CONT)

PURPOSE
READ THE RUN PARAMETERS FROM THE EXECUTION CONTROL FILE AND THE
TARIFF AND NETWORK FILES SPECIFIED TO BUILD THE DATA STRUCTURES
REQUIRED FOR A CNDC RUN

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
CONT LOGICAL
DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
EXECUTE /LUN/ U
FILNET /RUN/ R
HEXEC /HEADRS/ U
MAXTRF /INPEXC/ U
NCASE /RUN/ S
NOUT /INPEXC/ U
NRSTRT /INPEXC/ R
NTARIF /RUN/ R
OPTION /RUN/ R
OUTFLG(6) /RUN/ R
RUNID /INPEXC/ R
TF(13) /RUN/ R
LUNOUT /LUN/ U
DEBUG /LUN/ U

LOCAL VARIABLES
ERROR LOGICAL
HEADER(2) R*8
WORD I*4
ASTRX I*4

FILE HEADER READ FROM EXECUTE
FIRST WORD IN NEXT RECORD
CONSTANT '****'

SUBPROGRAMS CALLED
DIRGEN
DEFPRE
INBLNET
IBLTAR
```
ERRORS:

FATAL
1010 EXECUTION CONTROL FILE HEADER NOT AS EXPECTED

NONFATAL
1510 ERROR ENCOUNTERED IN BUILDING TARIFF INPUT FOR SPECIFIED PROBLEM
1520 NUMBER OF TARIFFS SPECIFIED IN EXECUTION CONTROL FILE OUT OF RANGE
1530 ERROR ENCOUNTERED IN BUILDING NETWORK FOR SPECIFIED PROBLEM
1540 ERROR ENCOUNTERED IN BUILDING DATA STRUCTURES FOR SPECIFIED CASE
1550 ERROR ENCOUNTERED IN INPUT FOR SPECIFIED CASE PROCEEDING TO NEXT CASE
Figure 3-58. INIT Logic Flow (Sheet 1 of 2)
**Figure 3-58. INIT Logic Flow (Sheet 2 of 2)**

1. **BUILD NETWORK DATA STRUCTURES**
2. **READ/VALIDATE NUMBER OF TARIFFS**
3. **GET TARIFF NAMES**
4. **BUILD TARIFF DATA STRUCTURES**
5. **ERROR?**
   - **YES** → **FIND START OF NEXT PROBLEM**
   - **NO** → **A**
3.6.2 **IBLNET Module**

**SUBROUTINE IBLNET(ERROR)**

**PURPOSE**
- READ THE NETWORK DESCRIPTION FILE AND BUILD THE NETWORK DATA STRUCTURES AND TRAFFIC TABLE DEFINED THEREIN

**INPUT ARGUMENTS**
NONE

**OUTPUT ARGUMENTS**
ERROR LOGICAL FLAG WHICH INDICATED IF AN ERROR WAS ENCOUNTERED WHILE VALIDATING INPUT

**DATA STRUCTURES**
(U - USED, S - DEFINED, R - READ)

- **FILNET /RUN/ U**
- **LUNNET /LUN/ U**
- **NETDEF /NETFIL/ U**
- **NFILES(20) /NETFIL/ U**
- **NNFIL /NETFIL/ U**
- **TRFLVL /RUN/ R**
- **UNET /HEADRS/ U**
- **OPTION /RUN/ U**
- **DEBUG /LUN/ U**

**LOCAL VARIABLES**

- **EMPTY I*4** CONSTANT /0/ FILE NOT DEFINED
- **FOUND 'LOGICAL** NETWORK FILE NAME FOUND IN NETWORK DIRECTORY
- **HEADER(2) R*8** FILE HEADER
- **I I*4** LOOP INDEX
- **SBONLY I*4** CONSTANT /3/ SBS TARIFF
- **UNIT I*4** LOGICAL UNIT NUMBER OF NETWORK FILE
- **WHO(600) I*2** IDENTIFIES FOR EACH NODE IN THE USER BY DEFINING THIS NUMBER OF THIS NODE IN THE PRESTORED NETWORK
- **WUONLY I*4** CONSTANT /2/ WU TARIFF
- **TEMP I*4** TRAFFIC LEVEL READ FROM NETWORK FILE

**SUBPROGRAMS CALLED**
- **IBLNOD**
- **IBLWHO**
- **IBLLST**
- **IBLTRF**

**ERRORS**

**FATAL**
- **1030** FILE HEADER OF USER DEFINED NETWORK NOT AS EXPECTED

**NONFATAL**
- **1560** NETWORK FILE REQUESTED IS NOT DEFINED
- **1570** NETWORK FILE REQUESTED NOT IN DIRECTORY
Figure 3-59. IBLNET Logic Flow (Sheet 1 of 2)
Figure 3-59. IBLNET Logic Flow (Sheet 2 of 2)
3.6.3 IBLNOD Module

SUBROUTINE IBLNOD(UNIT)

PURPOSE
READ THE SPECIFIED NETWORK FILE AND BUILD THE NETWORK AND THE DIRECTORY TO THE NETWORK

INPUT ARGUMENTS
UNIT I*4 LOGICAL UNIT NUMBER OF NETWORK FILE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
DRNODE /RNODES/ D
H(600) /RNODES/ D
NDIR /RNODES/ D
NNODES /RUN/ D
NPBLK /RNODES/ D
RTCNTR /RNODES/ D
V(600) /RNODES/ D
DEBUG /LUN/ U

LOCAL VARIABLES
I I*4 LOOP INDEX
INDX I*4 POINTER TO NEXT ELEMENT IN RATE CENTER ARRAY TO BE PUT INTO DIRECTORY

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
NONE
NONFATAL
NONE
IBLNOD

GET NUMBER OF NODES

ALL NODES READ?

YES

NO

GET NEXT NODE

BUILD DIRECTORY TO NODES

RETURN

Figure 3-60. IBLNOD Logic Flow
3.6.4 IBLWHO Module

SUBROUTINE IBLWHO(WHO)

PURPOSE
BUILD A TEMPORARY TABLE, WHO, REQUIRED TO BUILD THE EXECUTION TRAFFIC TABLE FROM THE PRESTORED TRAFFIC TABLE

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
WHO I*2 TABLE WHICH IDENTIFIES EACH NODE WITHIN THE USER NETWORK WHICH IS IN THE PRESTORED NETWORK BY THE RELATIVE POSITION OF THAT NODE WITHIN THE PRESTORED NETWORK. THIS ARRAY IS REQUIRED BY THE SUBPROGRAM WHICH BUILDS THE TRAFFIC TABLE TO PULL OUT THE CORRECT TRAFFIC LEVELS FROM THE PRESTORED TABLE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)

NNODES /RUN/ U
NPRNDS /PRSNDS/ U
RTCNTR /RNODES/ U
PRNDS /PRSNDS/ U
DEBUG /LUN/ U

LOCAL VARIABLES
I I*4 INDEX TO NEXT PRESTORED NODE
J I*4 INDEX TO NEXT INPUT NODE

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
NONE
NONFATAL
NONE
Figure 3-61. IBLWHO Logic Flow
3.6.5 IBLLST Module

SUBROUTINE IBLLST(_ERROR)

PURPOSE
BUILD THE DATA STRUCTURE LSTED WHICH IDENTIFIES ALL OF THE NODES WHICH ARE ATT CATEGORY A RATE CENTERS. THIS STRUCTURE IS REQUIRED ONLY IF THE RUN INVOLVES A TERRESTRIAL TARIFF

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
ERROR LOGICAL ERROR DETECTED WHILE VALIDATING INPUT

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
HPATT /HEADRS/ U
LSTED /ATT/ S
NCATA /INPATT/ S
NNODES /RUN/ U
PRCATA /INPATT/ S
RTCNTR /RNODES/ U
TRFATT /LUN/ U
NLEVL /ATT/ U
NPLEVL /ATT/ S
TATMLG /ATT/ S
DEBUG /LUN/ U

LOCAL VARIABLES
END I*4 POSITION IN ARRAY OF LAST CATA RATE CENTER READ FROM CURRENT RECORD OF PRESTORED ATT TARIFF
J I*4 LOOP INDEX
I I*4 INDEX TO NEXT RATE CENTER IN NETWORK
K I*4 INDEX TO NEXT PRESTORED CATA RATE CENTER
HEADER(2) R*8 HEADER RECORD READ FROM FILE TRFATT
NLEFT I*4 NUMBER CATA RATE CENTERS LEFT TO BE READ
NPREC I*4 NUMBER OF CATA RATE CENTERS PER RECORD
NREC I*4 NUMBER OF RECORDS CONTAINING CATA RATE CENTERS IN FILE TRFATT
DUM1-DUM6 R*4 DUMMY VARIABLES TO READ ATT TARIFF STRUCTURE
START I*4 POSITION IN ARRAY OF FIRST CATA RATE CENTER READ FROM CURRENT RECORD OF PRESTORED ATT TARIFF

SUBPROGRAMS CALLED
NONE
<table>
<thead>
<tr>
<th>ERRORS</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FATAL</strong></td>
<td>1020</td>
</tr>
<tr>
<td><strong>NONFATAL</strong></td>
<td>1640</td>
</tr>
</tbody>
</table>
GET ALL PRESTORED CATEGORY A RATE CENTERs

INITIALIZE LISTED STRUCTURE

ALL NODES PROCESSED?

GET PRESTORED MILEAGE BREAKPOINTS

RETURN

THIS NODE LISTED?

SET LISTED TO PRESTORED NUMBER

Figure 3-62. IBLLST Logic Flow
3.6.6 IBLTRF Module

SUBROUTINE IBLTRF(WHO,UNIT)

PURPOSE
BUILD THE TRAFFIC TABLE FROM THE PRESTORED TRAFFIC TABLE AND
THE ADDITIONAL TRAFFIC DEFINED IN THE NETWORK FILE

INPUT ARGUMENTS
WHO I*2 ARRAY IDENTIFYING POSITION WITHIN
PRESTORED NETWORK OF EVERY NODE IN
THE USER NETWORK WHICH IS FROM THE
PRESTORED SET

UNIT I*4 LOGICAL UNIT NUMBER OF USER NETWORK
DEFINITION FILE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)

NNODES /RUN/ U
TRFLVL /RUN/ U
TRFTBL /LUN/ U
TRAFIC /TRFC/ S
HTFC /HEADRS/ U
DEBUG /LUN/ U

LOCAL VARIABLES
HEADER(2) R*8 HEADER OF THE PRESTORED TRAFFIC TABLE
NEL I*4 NUMBER OF ELEMENTS PER ROW IN TRAFFIC
   TABLE
FACT I*4 NORMALIZATION FACTOR
LSTCOL I*4 LAST COLUMN OF PRESTORED TRAFFIC TABLE
   READ
CURROW I*4 ROW WITHIN TRAFFIC TABLE BEING DEFINED
I,J,K I*4 LOOP INDICES
NSKIP I*4 NUMBER OF ROWS IN PRESTORED TRAFFIC TABLE
   TO SKIP TO GET TO NEXT RECORD TO READ
TRF(350) I*4 NEXT RECORD FROM TRFTBL CONTAINING
   CURRENT COLUMN FROM TRAFFIC TABLE
M I*4 NEXT USERDEFINED MODE IN ROW
N I*4 NEXT ITEM IN RECORD
CURCOL I*4 CURRENT COLUMN BEING DEFINED
VCCNV R*4 CONVERSION FACTOR TO COMPUTE NUMBER OF
   VOICE CIRCUITS OR EACH NODE PAIR
NTRFC I*4 NUMBER OF SETS OF ADDITIONAL TRAFFIC
   SPECIFIED
ROWNAM I*4 ROW NAME
COLNAM I*4 COLUMN NAME OF ADDITIONAL TRAFFIC
VC I*4 NUMBER OF ADDITIONAL VOICE CIRCUITS
   BETWEEN ROWNAM AND COLNAM
ROWPTR I*4 ROW POSITION OF ROWNAM IN USER NETWORK

3-150
COLPTR  I*4  COLUMN POSITION OF COLNAM IN USER NETWORK

SUBPROGRAMS CALLED
   MNNPTR
   TRFSKP
   WRTTRF

ERRORS
   FATAL
      1040  FILE HEADER OF PRESTORED TRAFFIC TABLE NOT AS EXPECTED
   NONFATAL
      1650  ADDITIONAL TRAFFIC SPECIFIED FOR A NODE NOT IN
            THE USER DEFINED NETWORK
Figure 3-63. IBLTRF Logic Flow
3.6.7 IBLTAR Module

SUBROUTINE IBLTAR(ERROR)

PURPOSE
READ THE TARIFF FILES SPECIFIED IN THE EXECUTION CONTROL
FILE AND BUILD THE TARIFF DATA STRUCTURES

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
ERROR LOGICAL FLAG WHICH INDICATES IF AN ERROR
WAS ENCOUNTERED WHILE BUILDING THE
TARIFF DATA STRUCTURES

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
LUNTRF /LUN/ U
NTARIF /RUN/ U
NTFIL /TARFIL/ U
OPTION /RUN/ U
TARDEF(13) /TARFIL/ U
TF(13) /RUN/ U
TFILES(13) /TARFIL/ U
TRFATT /LUN/ U
TRFSBS /LUN/ U
TRFWU /LUN/ U
TTYP(13) /RUN/ D
DEBUG /LUN/ U

LOCAL VARIABLES
I I*4 LOOP INDEX
UNDFND I*4 CONSTANT /0/ FILE UNDEFINED
ATONLY I*4 CONSTANT /1/ ATT TARIFF
SBONLY I*4 CONSTANT /3/ SBS TARIFF
WUONLY I*4 CONSTANT /2/ WU TARIFF
WUMXED I*4 CONSTANT /4/ WU MIXED RUN
SBMxed I*4 CONSTANT /5/ SBS MIXED RUN
NAVAIL I*4 NEXT POSITION IN ATT STRUCTURES FOR NEXT
ATT TARIFF DATA
UNIT I*4 LOGICAL UNIT NUMBER OF TARIFF FILE
STATUS I*4 TYPE OF TARIFF DEFINED
1 ATT
2 WU
3 SBS
SAVTAR(13) I*4 SAVES NAMES OF TARIFFS THIS RUN
PRSTRD LOGICAL INDICATES IF TARIFF DEFINITION FILE IS
THE PRESTORED TARIFF
FOUND LOGICAL INDICATES IF FILE NAME FOUND IN DIRECTORY
TARNAM(3) R*8 CONSTANT- NAMES OF PRESTORED TARIFF FILES
K,J I*4 LOOP INDEX
SUBPROGRAMS CALLED
IBLWU1
IBLWU2
ISLWU3
IBLATT
IBLSB1
IBLSBS

ERRORS
FATAL
NONE
NONFATAL
1580 REQUESTED FILE NOT DEFINED
1590 STATUS OF FILE OUT OF RANGE
1600 MISMATCH BETWEEN RUN TYPE AND TARIFF FILE TYPE
1610 MORE THAN 2 TARIFF FILES SPECIFIED FOR MIXED RUN
1620 TARIFF OPTION OUT OF RANGE
1630 FOR MIXED MODE THERE SHOULD BE TWO TARIFFS DEFINED - ONE SHOULD DEFINE AN ATT TARIFF AND THE OTHER EITHER WU OR SBS
Figure 3-64. IBLTAR Logic Flow (Sheet 1 of 2)
Figure 3-64. IBLTAR Logic Flow (Sheet 2 of 2)
3.6.8 IBLATT Module

SUBROUTINE IBLATT(PRSTRD,UNIT,FILE,K,NAVAIL,ERROR)

PURPOSE
READ ATT TARIFF SPECIFIED, EITHER THE PRESTORED TARIFF OR
A NASA DEFINED ATT TYPE TARIFF, AND BUILD THE TERRESTRIAL
TARIFF DATA STRUCTURES

INPUT ARGUMENTS
PRSTRD L INDICATOR OF TYPE OF ATT TARIFF BEING
.DEFINED
.TRUE. PRESTORED ATT TARIFF
.FALSE. NASA DEFINED ATT TYPE TARIFF
UNIT I*4 LOGICAL UNIT NUMBER OF TARIFF FILE
FILE R*8 NAME OF FILE CONTAINING TARIFF DEFINITION
K I*4 NUMBER WITHIN CURRENT PROBLEM OF TARIFF
BEING DEFINED
NAVAIL I*4 NEXT AVAILABLE POSITION WITHIN THE ALSTD
DATA STRUCTURE WHICH DEFINE NASA ALTERA-
TIONS TO CATEGORY A RATE CENTER DEFINI-
TION

OUTPUT ARGUMENTS
NAVAIL I*4 NEXT AVAILABLE POSITION IN ALSTD DATA
STRUCTURE
ERROR L ERROR DETECTED VALIDATING INPUT

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
ALSTD /ATT/ S
NINLST /ATT/ S
TATBAS /ATT/ S
TATINC /ATT/ S
TATMLG /ATT/ U
TPTR /ATT/ S
NPLEVL /ATT/ U
HPATT /HEADRS/ U
UTAR /HEADRS/ U
DEBUG /LUN/ U

LOCAL VARIABLES
HEADER(2) R*8 FILE HEADER
NADJ I*4 NUMBER OF CATA ADJUSTMENTS IN USER
DEFINED TARIFF
NREC I*4 NUMBER OF RECORDS CONTAINING CATA
SPECIFICATIONS IN PRESTORED TARIFF
NCATA I*4 NUMBER OF CATA CENTERS IN PRESTORED
TARIFF
CODE I*4 CODE OF RATE CENTER SPECIFIED IN TARIFF
ADJUSTMENTS

3-157
<table>
<thead>
<tr>
<th>Variable</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADJ</td>
<td>I*4</td>
<td>NATURE OF ADJUSTMENT</td>
</tr>
<tr>
<td>PTR</td>
<td>I*4</td>
<td>NUMBER OF RATE CENTER IN CURRENT NETWORK</td>
</tr>
<tr>
<td>I</td>
<td>I*4</td>
<td>LOOP INDEX</td>
</tr>
<tr>
<td>NTLEVEL</td>
<td>I*4</td>
<td>NUMBER OF MILEAGE BREAKPOINTS SPECIFIED IN TARIFF FILE</td>
</tr>
<tr>
<td>MLG</td>
<td>I*4</td>
<td>MILEAGE BREAKPOINTS ON THIS FILE</td>
</tr>
<tr>
<td>ISTART</td>
<td>I*4</td>
<td>START OF ALSTD ARRAY TO PRINT FOR DEBUG</td>
</tr>
<tr>
<td>IEND</td>
<td>I*4</td>
<td>END OF ALSTD ARRAY TO PRINT FOR DEBUG</td>
</tr>
</tbody>
</table>

SUBPROGRAMS CALLED
MNMPTR
TRFSKP

ERRORS
FATAL
1050 HEADER RECORD OF TARIFF FILE NOT AS EXPECTED
1060 HEADER RECORD OF PRESTORD ATT FILE NOT AS EXPECTED
NONFATAL
1660 NUMBER OF MILEAGE BREAKPOINTS SPECIFIED IN TARIFF FILE NOT AS EXPECTED
1670 MILEAGE BREAKPOINTS SPECIFIED IN TARIFF FILE NOT AS EXPECTED

3-158
Figure 3-65. IBLATT Logic Flow
3.6.9 IBLWUI Module

SUBROUTINE IBLWUI

PURPOSE
IDENTIFY ALL OF THE NODES IN THE USER DEFINED NETWORK FOR
THIS RUN WHICH ARE SATELLITE ACCESS CITIES AS DEFINED IN
THE PRESTORED WU TARIFF

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
SA /WU/ S
NNODES /RUN/ U
HPWU /HEADRS/ U
TRFWU /LUN/ U
WUPRS /WU/ S
NWUTRI /WU/ U
NTARIF /RUN/ U
DEBUG /LUN/ U

LOCAL VARIABLES
HEADER(2) R*8 HEADER RECORD READ FROM TRFWU
I,J I*4 LOOP INDEX
NPAIRS I*4 NUMBER OF SATELLITE ACCESS CITY PAIRS IN
PRESTORED TARIFF
CODEL I*4 LEFT MOST CODE SPECIFIED IN TARIFF
CODER I*4 RIGHT MOST CODE SPECIFIED IN TARIFF
CAT I*4 TARIFF RATE
PCODEL I*4 PREVIOUS LEFT MOST CODE READ
DUMY I*4 CONSTANT TO INITIALIZE PCODEL
LPtr I*4 POSITION OF CODEL IN USER DEFINED NETWORK
    IF LPtr=0, CODEL NOT IN NETWORK
RPTR I*4 POSITION OF CODER IN USER DEFINED NETWORK
    IF RPTR=0, CODER NOT IN NETWORK

SUBPROGRAMS CALLED
MNMPTR

ERRORS
FATAL
1070 HEADER RECORD READ FROM WU PRESTORED TARIFF NOT AS
    EXPECTED

NONFATAL
NONE
Figure 3-66. IBLWU1 Logic Flow
SUBROUTINE IBLWU2(PRSTRD,UNIT,FILE,K,SAYTAR)

PURPOSE
IDENTIFY ALL OF THE NODES IN THE USER DEFINED NETWORK FOR
THIS PROBLEM WHICH ARE SATELLITE ACCESS CITY AS IDENTIFIED
IN THE USER DEFINED WU TARIFFS

INPUT ARGUMENTS
PRSTRD  L  INDICATES IF THE CURRENT WU TARIFF IS
          .TRUE.  PRESTORED
          .FALSE. USER DEFINED
UNIT     I*4  LOGICAL UNIT NUMBER OF CURRENT WU TARIFF
FILE     I*4  NAME OF FILE DEFINING CURRENT WU TARIFF
K        I*4  NUMBER OF CURRENT TARIFF

OUTPUT ARGUMENTS
SAYTAR(13) I*4  LUN FOR EACH WU TARIFF DEFINED FOR THIS
             PROBLEM

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
UTAR      /HEADR/  U
SA        /WU/    S
DEBUG     /LUN/   U

LOCAL VARIABLES
HEADER(2)  R*8  FILE HEADER READ FROM CURRENT TARIFF
NADJ      I*4  NUMBER OF ADJUSTMENTS TO TARIFF
PCODEL    I*4  PREVIOUS LEFT CODE IN TARIFF
DUMY      I*4  CONSTANT FOR INITIALIZING PCODEL
CODEL     I*4  LEFT CODE READ FROM CURRENT TARIFF
I         I*4  LOOP INDEX
CODER     I*4  RIGHT CODE SPECIFIED IN CURRENT TARIFF
LPTR      I*4  NUMBER OF CODEL IN USER DEFINED NETWORK
RPTR      I*4  NUMBER OF CODER IN USER DEFINED NETWORK
CAT       I*4  RATE TYPE OF SATELLITE CITY PAIR

SUBPROGRAMS CALLED
MMMPTR

ERRORS
FATAL      1080  FILE HEADER READ FROM USER DEFINED WU TARIFF NOT AS
               EXPECTED
NONFATAL   NONE
Figure 3-67. IBLWU2 Logic Flow
SUBROUTINE IBLWU3(SAVTAR,ERROR)

PURPOSE
DETERMINE THE NUMBER OF SATELLITE ACCESS CITIES IN THIS
PROBLEM AND BUILD REMAINING WU TARIFF DATA STRUCTURES

INPUT ARGUMENTS
SAVTAR(13) I*4 ARRAY OF LOGICAL UNIT NUMBERS FOR WU TARIFF

OUTPUT ARGUMENTS
ERROR LOGICAL INDICATES IF ERROR ENCOUNTERED WHILE
VALIDATING WU TARIFF INPUT

DATA STRUCTURES
{U - USED, S - DEFINED, R - READ}
NNODES /RUN/ U
NTARIF /RUN/ U
OPTION /RUN/ U
TRFWU /LUN/ U
NSATAX /WU/ S
SA /WU/ S
SATACC /WU/ S
TWU /WU/ S
WUPRS /WU/ S
DEBUG /LUN/ U
HPWU /HEADRS/ U
WURTCT /WU/ S
MAXSAC /WU/ U
UTAR /LUN/ U
TF /RUN/ U
NWUCAT /WU/ U

LOCAL VARIABLES
BEGIN I*4 FIRST TARIFF TO CONSIDER
END I*4 LAST TARIFF TO CONSIDER
HEADER(2) R*8 HEADER RECORD FROM WU TARIFF
PCODEL I*4 PREVIOUS LEFT CODE READ FROM TARIFF
CDELT I*4 LEFT CODE IN TARIFF CITY PAIR
CODER I*4 RIGHT CODE IN TARIFF CITY PAIR
DUMY I*4 CONSTANT TO INITIALIZE PCODEL
NPAIRS I*4 NUMBER OF ACCESS CITY PAIRS SPECIFIED IN TARIFF
CAT I*4 TARIFF CATEGORY
I,J,K I*4 LOOP INDICES
LPTR I*4 POSITION OF CDELT IN USER DEFINED NETWORK
RPTR I*4 POSITION OF CODER IN USER DEFINED NETWORK
PTR I*4 POSITION WUPRS OF CITY PAIR LPTR, RPTR
FOUND L INDICATES ONE OF WU TARIFFS IS THE PRE-
STORED TARIFF
UNIT I*4 LOGICAL UNIT NUMBER OF THE TARIFF UNDER CONSIDERATION
JEND I*4 NUMBER OF ROWS TO CONSIDER
KBEG I*4 BEGINNING TO NEXT ROW IN UPPER RIGHT TRIANGLE OF SQUARE MATRIX
LPTR2 I*4 POSITION OF CODEL IN SATELLITE ACCESS CITIES
RPTR2 I*4 POSITION OF CORDER IN SATELLITE ACCESS CITIES
FIRST I*4 INDEX TO PRINT ARRAY IN DEBUG PRINT
SEC I*4 INDEX TO PRINT ARRAY IN DEBUG PRINT
LAST I*4 INDEX TO PRINT ARRAY IN DEBUG PRINT
FINL I*4 INDEX TO PRINT ARRAY IN DEBUG PRINT
NEXT I*4 INDEX TO PRINT ARRAY IN DEBUG PRINT

SUBPROGRAMS CALLED
MNMPTR
PTRPRS

ERRORS
FATAL
1090 FILE HEADER OF PRESTORED WU TARIFF NOT AS EXPECTED
1100 FILE HEADER OF USER DEFINED W5 TARIFF NOT AS EXPECTED

NONFATAL
1680 NUMBER OF SATELLITE ACCESS CITIES FOR THIS PROBLEM EXCEEDS ARRAY CAPACITY OF THE MODEL
Figure 3-68  IBLWU3 Logic Flow
3.6.12 IBLSBI Module

SUBROUTINE IBLSBI(ERROR)

PURPOSE
DETERMINE PRESTORED NUMBER OF CAU PER NAC BREAKPOINTS

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
ERROR L INDICATES IF ERROR DETECTED DURING INPUT

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
TRFSBS /LUN/ U
HPSBS /HEADRS/ U
TSBBKP /SBS/ S
NPTAB /SBS/ S
NTAB /SBS/ U
DEBUG /LUN/ U

LOCAL VARIABLES
HEADER(2) R*8 HEADER READ FROM PRESTORED SBS TARIFF
DUM1-DUM2 R*4 DUMMY VARIABLES READ FROM SBS TARIFF
I I*4 LOOP INDEX

SUBPROGRAMS CALLED
TRFSKP

ERRORS
FATAL
1110 FILE HEADER READ FROM PRESTORED SBS TARIFF NOT AS EXPECTED

NONFATAL
1690 NUMBER OF ENTRIES IN CAU COST TABLE GREATER THAN CAPACITY OF ARRAYS IN MODEL
Figure 3-69. IBLSB1 Logic Flow
3.6.13 IBLSBS Module

SUBROUTINE IBLSBS(PRSTRD,UNIT,FILE,K,ERROR)

PURPOSE
READ THE PRESTORED OR NASA DEFINED SBS TARIFF SPECIFIED AND
BUILD SBS TARIFF DATA STRUCTURES

INPUT ARGUMENTS
PRSTRD L LOGICAL FLAG WHICH INDICATES IF THE
PRESTORED SBS OR NASA DEFINED SBS
FILE IS TO BE READ
.TRUE. PRESTORED SBS TARIFF
.FALSE. NASA DEFINED SBS TARIFF
UNIT I*4 LOGICAL UNIT NUMBER OF TARIFF TO BE READ
FILE R*8 FILE NAME OF TARIFF
K I*4 NUMBER OF CURRENT TARIFF IN THIS PROBLEM

OUTPUT ARGUMENTS
ERROR L INDICATES IF ERROR DETECTED WHILE VALIDATING TARIFF INPUT

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
TSBBKP /SBS/ U
TSBCAB /SBS/ S
TSBcai /SBS/ S
TSBCMn /SBS/ S
TSBFTU /SBS/ S
TSBNAC /SBS/ S
TSBSGU /SBS/ S
NPTAB /SBS/ U
HPSBS /HEADRS/ U
UTAR /HEADRS/ U
NACNMB /SBSMXD/ S
NACLOC /SBSMXD/ S
SBMX /SBSMXD/ S
OPTION /RUN/ U
DEBUG /LUN/ U

LOCAL VARIABLES
NTABL I*4 NUMBER OF BREAKPOINTS IN INPUT SBS TABLE
BKP I*4 BREAKPOINT READ FROM INPUT SBS TABLE
HEADER(2) R*8 HEADER READ FROM SBS TARIFF FILE
NREC I*4 NUMBER OF RECORDS CONTAINING NAC LOCATIONS
NENTRY I*4 NUMBER OF NAC SITES SPECIFIED
NPREC I*4 NUMBER OF NAC SPECIFICATIONS PER RECORD
NLEFT I*4 NUMBER OF NAC SPECIFICATIONS LEFT TO READ
I,J I*4 LOOP INDICES
END I*4 NUMBER OF NACS SPECIFIED ON CURRENT RECORD

3-169
NXTNAC(16) I*4 ONE RECORDS WORTH OF NACS

SUBPROGRAMS CALLED
USRNET

ERRORS
FATAL
1120 FILE HEADER OF SBS TARIFF NOT AS EXPECTED
1130 FILE HEADER OF PRESTORED SBS TARIFF NOT AS EXPECTED
NONFATAL
1700 NUMBER OF BREAKPOINTS IN SBS TABLE NOT AS EXPECTED
1710 BREAKPOINT VALUES IN SBS TABLE NOT AS IN PRESTORED TARIFF
Figure 3-70. IBLSBS Logic Flow
3.7 OUTPUT DESCRIPTION

The CNDC model provides six categories of output: (1) user inputs to a problem, (2) city pair output - exclusive of other network traffic, (3) city pair output for least cost network, (4) network totals, (5) tariff totals, and (6) traffic table. The user selects which output is desired for each problem set at the time the execution control file is built. These six categories of output are contained in six tables which will be described below.

Table 1 - Input

This table presents the runid, current problem number, run type (terrestrial, Western Union, SBS, mixed Western Union, and mixed SBS), the name of the network file, number of tariffs and name of each tariff file. Each node in the network is identified by its four character code and name. The traffic level of the prestored traffic table is specified in voice circuits. Finally, each tariff included in the run is fully described.

Table 2 - City Pair Output - Exclusive of Other Network Traffic

The least-cost route and the transmission medium that is necessary to satisfy the traffic requirement for each unique pair of traffic nodes, exclusive of other network traffic, is listed in tabular form. For each unique city pair, a description of the least-cost route is followed by a table summarizing information about each link in the route.

Each unique pair of traffic nodes and the nodes in the corresponding least-cost route are identified by using the alphabetic codenames assigned to the cities. An ordered list of the codenames of the nodes in the least-cost route is followed by the total circuit mileage on the route. The link summary is a table containing one line of output for each link in the least-cost route.
There are seven items included in each link summary. The alphabetic codenames of the service nodes defining the link are indicated for each link. The airline mileage is calculated from the vertical and horizontal coordinates of the associated service nodes using the standard mileage formula. The traffic volume (in voice circuits) on each line is calculated by summing all traffic internal to the least-cost route that traverses the link. A facility size is determined to meet the requirements of this traffic volume. The cost per circuit that was determined to be a minimum for the link considering all tariffs in the run is included. The total cost of circuits for the link is the product of the traffic volume in voice circuits and the cost of a single circuit. The tariff used to determine the cost per circuit for the link is identified by its character description.

Entries in the line summary are added until all the lines in the least-cost route have been summarized. The same information is then output for the next unique city pair. Least-cost routes are described for all pairs of cities in the network.

Table 3 - City Pair Output for Least-Cost Network

For every city pair, considering traffic in the entire network, the least-cost route and medium that is necessary to satisfy the traffic requirements of the network is included in the CNDC output. If the satellite-only option was chosen any non-satellite access cities among the network nodes are indicated by a list of their alphabetic codenames. Non-satellite access cities do not appear further in the least-cost network output.

Each city pair and the nodes in the corresponding least-cost route are identified using alphabetic codenames. For each pair of cities, an enumeration of the nodes in the least-cost route is followed by a tabular line summary.

There are five items included in each link summary. The alphabetic codenames of the service nodes defining the link are identified for each link. The network traffic volume on each link represents the total number of 3-173
voice circuits on the link. This number is the sum of the traffic volumes between all pairs of cities where the least-cost route between the cities traverses the link in question. The facility size required to accommodate the total link volume is included in the link summary. The cost per circuit that was determined to be a minimum for the link considering all tariffs in the run is listed. The tariff used to provide service for the link is specified by its character description (TRFWU, TRFO03, etc).

Entries in the link summary are added until all the links in the least-cost route have been summarized. The same information is then output for the next unique city pair. Least-cost routes and the medium necessary to satisfy the traffic requirements of the network are described for all pairs of cities in the network.

Table 4 - Output Network Totals by Service

Overall totals for the entire least-cost network by service (satellite and terrestrial) are included in the CNDC output. The total terrestrial circuit mileage, the total satellite circuit mileage, and the combined total circuit mileage are printed. The total number of terrestrial voice circuits, the total number of satellite voice circuits, and the combined total number of voice circuits are printed. The total cost of all the terrestrial circuits, the total cost of all the satellite circuits, and the combined total cost of circuits are printed.

Table 5 - Output Network Totals by Tariff

For every tariff evaluated in the CNDC Model, a summary of its utilization in the least-cost routing network is included in the model output. Each tariff is identified by its character description. The total circuit mileage associated with each tariff in the least-cost routing network is printed. The total traffic volume associated with each tariff is expressed as a number of voice circuits and as a percentage of the total number of voice circuits in the least-cost routing network. The total cost associated with each
tariff is expressed as a cost and as a percentage of the total cost of the least-cost routing network.

Table 6 - Traffic Table

Directional traffic from each node in the network to every other node (expressed in voice circuits) is presented in the traffic table.

A hierarchy chart of the output module can be seen in figure 3-71.
Figure 3-71. OUTPUT Hierarchy Chart
3.7.1 Output Module

SUBROUTINE OUTPUT

PURPOSE
PREPARE AND WRITE ALL MODEL OUTPUTS

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)

OUTFLG /RUN/ U
LUNOUT /LUN/ U
RUNID /INPEXC/ U
NCASE /RUN/ U
OPTION /RUN/ U
TRAFIC /TRFC/ U
RTCNTR /RNODES/ U
NNODES /RUN/ U
NTARIF /RUN/ U

LOCAL VARIABLES

HEADER L HEADER TO LIST ON NON ACCESS CITIES
TOTAL I*4 TOTAL CIRCUITS AMONG NON ACCESS CITIES
COL I*4 COLUMN WITHIN TRAFFIC TABLE
ENDCOL I*4 LAST ITEM IN COLUMN TO SEARCH
ROW I*4 ROW WITHIN TRAFFIC TABLE
VFS I*4 TRAFFIC BETWEEN NON ACCESS CITY PAIR
NODA I*4 ALPHABETIC DESCRIPTION OF ACCESS CITY
NODB I*4 ALPHABETIC DESCRIPTION OF ACCESS CITY
FND L INDICATES IF NODE PAIR IS ACCESS CITY PAIR FOR ANY WU TARIFF
K I*4 NEXT TARIFF

SUBPROGRAMS CALLED
OUT1
OUT2
OUT3
OUT4
WRTTRF
CATEG

ERRORS
NONE
Figure 3-72. OUTPUT Logic Flow
3.7.2 OUT1 Module

SUBROUTINE OUT1

PURPOSE
PRINT INPUT AS OUTPUT

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
NCASE /RUN/  U
LUNOUT /LUN/  U
RUNID /INPEXC/  U
NNODES /RUN/  U
RTCNTR /RNODES/  U
TRFLVL /RUN/  U
NTARIF /RUN/  U
TYP /RUN/  U
TF /RUN/  U
DEBUG /LUN/  U
TRFNDS /LUN/  U
FILNET /RUN/  U

LOCAL VARIABLES
DESCR(5)  R*8  CONSTANT - LABEL OF PROBLEM TYPE
TYPE  I*4  TARIFF TYPE
I,J  I*4  LOOP INDICES
DUMY  I*4  DUMMY VALUE READ FROM TRAFFIC NODES
USERDEF(9)  I*4  CONSTANT /'USER DEFINED'/
RTC  I*4  RATE CENTER READ FROM TRFNDS
VT  I*4  VERTICAL COORDINATE
HT  I*4  HORIZONTAL COORDINATE
CITY(9)  I*4  CITY DESCRIPTION OF RATE CENTER

SUBPROGRAMS CALLED
OUTATT
OUTWU
OUTSBS
SMSA

ERRORS
FATAL
NONE
NONFATAL
6000  TARIFF IN PROBLEM HAS INVALID TYPE
Figure 3-73. OUT1 Logic Flow
3.7.3 OUTATT Module

SUBROUTINE OUTATT(TARNAM, TARNUM)

PURPOSE
PRINT DESCRIPTION OF ATT TARIFF IN OUTPUT FILE

INPUT ARGUMENTS
TARNAM R*8 NAME OF TARIFF FILE BEING DESCRIBED
TARNUM I*4 NUMBER OF THIS TARIFF IN PROBLEM

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
LUNOUT /LUN/ U
NPLEVEL /ATT/ U
TATMLG /ATT/ U
TATBAS /ATT/ U
TATINC /ATT/ U
NNODES /RUN/ U
RTCNTR /RNODES/ U
DEBUG /LUN/ U

LOCAL VARIABLES
PRSTRO R*8 CONSTANT /'TRFATT'/
CTGA(10) I*4 RECORD OF CATA RATE CENTERS TO PRINT TO FILE
NCTG I*4 NUMBER OF RECORDS OF CATA RATE CENTERS TO PRINT
UNLSTD I*4 CONSTANT /1/ UNLISTED RATE CENTER
I,J I*4 LOOP INDICES
IEND I*4 LAST TARIFF LEVEL TO DESCRIBE IN LOOP
NEXT I*4 START MILEAGE FOR TARIFF LEVEL
LAST I*4 END MILEAGE FOR TARIFF LEVEL
NREC I*4 NUMBER OF RECORDS OF CATA RATE CENTERS TO PRINT

SUBPROGRAMS CALLED
LISTED

ERRORS
FATAL NONE
NONFATAL NONE

3-181
Figure 3-74. OUTATT Logic Flow
3.7.4 OUTWU Module

SUBROUTINE OUTWU(TARNAM, TARNUM)

PURPOSE
PRINT DESCRIPTION OF WU TARIFF IN OUTPUT FILE

INPUT ARGUMENTS
TARNUM R*8 NUMBER OF TARIFF FILE BEING DESCRIBED
TARNUM I*4 NUMBER OF THIS TARIFF IN PROBLEM

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
LUNOUT /LUN/ U
NNODES /RUN/ U
RTCNTR /RNODES/ U
TWU /WU/ U
DEBUG /LUN/ U

LOCAL VARIABLES
PRSTRO R*8 CONSTANT NAME OF PRESTORED TARIFF
NCTG I*4 NUMBER OF SATELLITE ACCESS CITY PAIRS
I,J I*4 LOOP INDICES
CTG I*4 RATE CATEGORY
END I*4 LAST RATE CENTER TO CONSIDER FOR LEFT
BEGIN I*4 FIRST RATE CENTER TO CONSIDER FOR RIGHT
NODE OF SATELLITE ACCESS PAIR POSSIBILITY
NODE OF SATELLITE ACCESS RATE POSSIBILITY
DESCP C*16 DESCRIPTION OF RATE CATEGORY

SUBPROGRAMS CALLED
CATEG

ERRORS
FATAL NONE
NONFATAL NONE

3-183
Figure 3-75. OUTWU Logic Flow
3.7.5 OUTSBS Module

SUBROUTINE OUTSBS(TARNAM, TARNUM)

PURPOSE
PRINT DESCRIPTION OF SBS TARIFF IN OUTPUT FILE

INPUT ARGUMENTS
TARNAM         R*8        NAME OF TARIFF FILE BEING DESCRIBED
TARNUM         I*4        NUMBER OF THIS TARIFF IN PROBLEM

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
LUNOUT        /LUN/        U
SBMX          /SBSMXD/     U
NACNMB        /SBSMXD/     U
NACLOC        /SBSMXD/     U
TSBNAC        /SBS/        U
TSBCSU        /SBS/        U
TSBFTU        /SBS/        U
TSBCMN        /SBS/        U
NPTAB         /SBS/        U
TSBBKP        /SBS/        U
TSBCAB        /SBS/        U
TSBCAI        /SBS/        U
DEBUG         /LUN/        U
OPTION        /RUN/        U
RTCWTR        /RNODES/     U

LOCAL VARIABLES
PRSTRD         R*8        CONSTANT /'TRFSBS'/ NAME OF PRESTORED SBS TARIFF
NREC          I*4        NUMBER OF RECORDS TO CONTAIN NAC LOCATIONS
BEGIN         I*4        INDEX TO FIRST ENTRY OF NACLOC ARRAY TO BE PRINTED ON NEXT OUTPUT RECORD
END           I*4        INDEX TO LAST ENTRY OF NACLOC ARRAY TO BE PRINTED ON NEXT OUTPUT RECORD
I,J           I*4        LOOP INDICES
NL            I*4        NUMBER OF NEXT NAC
REC(10)       I*4        ONE LINES WORTH OF NAC NAMES
NITM          I*4        NUMBER OF ITEMS TO PUT IN PRINT LINE

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
NONE
NONFATAL
NONE
Figure 3-76. OUTSBS Logic Flow
3.7.6 OUT2 Module

SUBROUTINE OUT2

PURPOSE
PREPARE AND WRITE TABLE 2. LEAST-COST ROUTES (EXCLUSIVE OF OTHER NETWORK TRAFFIC)

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)

RTCNTR /RNODES/ U
MINCPC /CSTOUT/ U
NNODES /RUN/ U
OROUTE /CSTOUT/ U
SVC /CSTOUT/ U
TF /RUN/ U
TRAFIC /TRFC/ U
OPTION /RUN/ U
LUNOUT /LUN/ U
SATNVC /SBSMX2/ U

LOCAL VARIABLES

DASH I*4 CONSTANT /'-'/
I I*4 LOOP INDEX - FROM CITIES
IEND I*4 END LOOP
I2END I*4 END LOOP
TINT I*4 LOOP INDEX - LINKS
ISTART I*4 START LOOP
ITEMP I*4 NODE NUMBER - USED IN TRACING ROUTE
IRT I*4 FROM NODE - USED IN SUMMING ROUTE TRAFFIC ON LINKS
IPRT I*4 USED FOR PRINTING CITY MNEMONICS
J I*4 LOOP INDEX - TO CITIES
JINT I*4 LOOP INDEX - FOR SUMMING ROUTE TRAFFIC ON LINKS
JRT I*4 TO NODE - USED IN SUMMING ROUTE TRAFFIC ON LINKS
JPRT I*4 USED FOR PRINTING CITY MNEMONICS
K I*4 LOOP INDEX - LINKS
KFROM I*4 ORIGINATING NODE OF A LINK
KFPRRT I*4 USED TO PRINT FROM CITY MNEMONICS
KPRT I*4 USED FOR PRINTING CITY MNEMONICS
KTPRT I*4 USED TO PRINT TO CITY MNEMONICS
KTO I*4 TERMINATING NODE OF A LINK
LNKMLS I*4 TOTAL CIRCUIT MILEAGE ON A ROUTE
LNKTOT I*4 ARRAY CONTAINING TOTAL ROUTE TRAFFIC ON LINKS

3-187
MAXLNK I*4  CONSTANT /20/ MAXIMUM LINKS ON A ROUTE
MPRT  I*4  USED TO PRINT MILEAGE ON A LINK
NB  I*4  NUMBER OF BASE GROUPS IN FACILITY
TRFC  I*4  NUMBER OF VOICE CIRCUITS BETWEEN NODAL PAIR
RMIN  R*4  DOLLAR PART OF MULTIPLIER
RTMP  R*4  DOLLAR PART OF RESULT
IMIN  I*4  CENTS PART OF MULTIPLIER
ITMP  I*4  CENTS PART OF RESULT
CA1-CA3 I*2  CHARACTER REPRESENTATION OF IMIN
CB1-CB3 I*2  CHARACTER REPRESENTATION OF ITMP
NJ  I*4  NUMBER OF JUMBO GROUPS IN FACILITY
MLINKS  I*4  NUMBER OF LINKS IN A ROUTE
NM  I*4  NUMBER OF MASTER GROUPS IN FACILITY
NS  I*4  NUMBER OF SUPER GROUPS IN FACILITY
ROUTE  I*2  ARRAY CONTAINING NODE NUMBERS IN LEAST-COST ROUTE
SAT  C*16  CONSTANT / 'SATELLITE' /
SERVICE  C*16  VARIABLE SET EQUAL TO SAT OR TERR
TEMP  R*4  COST OF CIRCUITS ON ROUTE
TERR  C*16  CONSTANT / 'TERRESTRIAL' /
TPRT  R*4  USED TO PRINT TARIFF USED ON
RNAME  I*4  NAME OF EACH LINK IN PATH
STLTE  I*4  CONSTANT /2/ SATELLITE SERVICE EACH LINK
SRV  I*4  SERVICE TYPE
1 TERRESTRIAL
2 SATELLITE
VC  I*2  NUMBER OF VOICE CIRCUITS FROM NAC
SCHED  I*4  TERRESTRIAL TARIFF SCHEDULE USED
NVC  I*4  NUMBER OF VOICE CIRCUITS
RTSBS  L  .TRUE. IF DESCRIPTION OF SBS CHARGES TO BE PRINTED
ITR  I*4  TARIFF NUMBER

SUBPROGRAMS CALLED
FACSIZ
MILES
OUT2SB
PRTSBS
LISTED
MTFY
MTPLY
INTCHR

ERRORS
FATAL
NONE
NONFATAL
NONE
Figure 3-77. OUT2 Logic Flow
3.7.7 **OUT2SB Module**

**SUBROUTINE OUT2SB(I,IPRT,J,JPRT,VC)**

**PURPOSE**
PREPARE AND WRITE TABLE 2 LEAST COST ROUTES (EXCLUSIVE OR OTHER
NETWORK TRAFFIC) FOR PATH WHICH INCLUDES SBS SATELLITE

**INPUT ARGUMENTS**

| I  | I*4  | FROM CITY |
| J  | I*4  | TO CITY   |
| IPRT | I*4 | CITY MNEMONIC OF FROM CITY |
| JPRT | I*4 | CITY MNEMONIC OF TO CITY |

**OUTPUT ARGUMENTS**

| VC | I*4 | NUMBER OF VOICE CIRCUITS FROM THIS PARTITION |

**DATA STRUCTURES**

(U - USED, S - DEFINED, R - READ)

| NODPTR   | /SBSMXD/ | U |
| NACLOC   | /SBSMXD/ | U |
| RTCNTR   | /RNODES/ | U |
| TRAFIC   | /TRFC/   | U |
| MINCPC   | /CSTOUT/ | U |
| TF       | /RUN/    | U |
| SATCST   | /SBSMX2/ | U |
| OROUTE   | /CSTOUT/ | U |
| LUNOUT   | /LUN/    | U |
| DEBUG    | /LUN/    | U |
| SATNVC   | /SBSMX2/ | U |

**LOCAL VARIABLES**

| DASH     | I*4  | CONSTANT '/-'/ |
| LNKMLS   | I*4  | TOTAL CIRCUIT MILEAGE ON A ROUTE |
| K, II, JJ | I*4  | LOOP INDICES |
| MAXLNK   | I*4  | CONSTANT '/20/ MAX LINKS ON A ROUTE |
| ROUTE(20) | I*2  | ARRAY CONTAINING NODE NUMBERS IN LEAST COST ROUTE |
| LNKTOT(20) | I*4  | ARRAY CONTAINING TOTAL ROUTE TRAFFIC ON LINKS |
| RNAME(20) | I*4  | NAME OF EACH NODE IN PATH |
| BLANK    | I*4  | CONSTANT '/-'/ |
| NLINKS   | I*4  | NUMBER OF LINKS IN PATH BETWEEN TWO NODES |
| CURFRM   | I*4  | FROM NODE OF CURRENT LINK |
| SATFRM   | I*4  | FROM SATELLITE IN THIS PATH |
| PART     | I*4  | PARTITION OF SENDING SATELLITE |
| SATTO    | I*4  | THE RECEIVING SATELLITE |
| CURTO    | I*4  | TO NODE OF CURRENT LINK |
| IRT      | I*4  | FROM NODE |
| JRT      | I*4  | TO NODE |
| NLA      | I*4  | NUMBER OF LINKS UNDER CONSIDERATION IN I, J PATH |
NLB  I*4  NUMBER OF LINKS IN RTB PATH
RTB(20) I*4  LINKS THAT MAKE UP PATH BETWEEN ANY TWO
SAME  L  TWO PATHS - SUBSET OF ROUTE AND RTB - ARE
       THE SAME
TEMP  R*4  COST OF CIRCUITS ON ROUTE
CPC  R*4  COST PER CIRCUIT
KFROM I*4  ORIGINATING NODE OF A LINK
TRFC I*4  NUMBER VOICE CIRCUITS BETWEEN NODAL PAIR
ITEMP I*4  SUM OF CENTS PORTION OF COST
RMIN R*4  DOLLAR PORTION OF MULTIPLIER
IMIN I*4  CENTS PORTION OF MULTIPLIER
RTMP R*4  DOLLAR PORTION OF RESULT
ITMP I*4  CENTS PORTION OF RESULT
CA1-CA3 I*2  CHARACTER REPRESENTATION OF IMIN
CB1-CB3 I*2  CHARACTER REPRESENTATION OF ITMP
KTO I*4  TERMINATING NODE OF A LINK
NJ I*4  NUMBER OF JUMBO GROUPS
NM I*4  NUMBER OF MASTER GROUPS
NS I*4  NUMBER OF SUPER GROUPS
NB I*4  NUMBER OF BASE GROUPS
SERVICE C*16  TYPE OF SERVICE - SATELLITE OR
       TERRESTRIAL
TERR C*16  CONSTANT '/TERRESTRIAL'/
SAT  C*16  CONSTANT '/SATELLITE'/
TPRT R*8  NAME OF TARIFF
KFPRT I*4  4 CHARACTER CODE OF FROM NODE
KTPRT I*4  4 CHARACTER CODE OF TO NODE
MPRT I*4  LINK MILEAGE
IBEG I*4  FIRST ITEM OF ARRAY TO BE REFERENCED
IEND I*4  LAST ITEM OF ARRAY TO BE REFERENCED
ILAST I*4  LAST ITEM OF ARRAY TO BE REFERENCED
SRV I*4  SERVICE TYPE
       1  TERRESTRIAL
       2  SATELLITE
SCHED I*4  TERRESTRIAL SCHEDULE

SUBPROGRAMS CALLED
MILES
GETRTE
FACSIZ
LISTED
MTPY
MTPLY
INTCHR

ERRORS
FATAL
NONE
NONFATAL
NONE
NLB  I*4  NUMBER OF LINKS IN RTB PATH
RTB(20)  I*4  LINKS THAT MAKE UP PATH BETWEEN ANY TWO
          NODES IN PATH I, J
SAMF   L   TWO PATHS - SUBSET OF ROUTE AND RTB - ARE
          THE SAME
ILMP   R*4  COST PER CIRCUIT ON ROUT.
CPC    R*4  COST PER CIRCUIT
IFROM  I*4  ORIGINATING NODE OF A LINK
TRFC   I*4  NUMBER VOICE CIRCUITS BETWEEN NODAL PAIR
ITEMP  I*4  SUM OF CENTS PORTION OF COST
RMIN   R*4  DOLLAR PORTION OF MULTIPLIER
IMIN   I*4  CENTS PORTION OF MULTIPLIER
RTPM   R*4  DOLLAR PORTION OF RESULT
ITPM   I*4  CENTS PORTION OF RESULT
CA1-CA3 I*2  CHARACTER REPRESENTATION OF IMIN
CB1-CB3 I*2  CHARACTER REPRESENTATION OF ITMP
KTO    I*4  TERMINATING NODE OF A LINK
NJ     I*4  NUMBER OF JUMBO GROUPS
NM     I*4  NUMBER OF MASTER GROUPS
NS     I*4  NUMBER OF SUPER GROUPS
NB     I*4  NUMBER OF BASE GROUPS
SERVICE C*16  TYPE OF SERVICE - SATELLITE OR
              TERRESTRIAL
TERR   C*16  CONSTANT /'TERRESTRIAL'/
SAT    C*16  CONSTANT /'SATELLITE'/
TPRT   R*8  NAME OF TARIFF
KFPT   I*4  4 CHARACTER CODE OF FROM NODE
KTPRT  I*4  4 CHARACTER CODE OF TO NODE
MPRT   I*4  LINK MILEAGE
IBEG   I*4  FIRST ITEM OF ARRAY TO BE REFERENCED
IEND   I*4  LAST ITEM OF ARRAY TO BE REFERENCED
ILAST  I*4  LAST ITEM OF ARRAY TO BE REFERENCED
SRV    I*4  SERVICE TYPE
              1  TERRESTRIAL
              2  SATELLITE
SCHED  I*4  TERRESTRIAL SCHEDULE

SUBPROGRAMS CALLED
MILES
GETRTE
FACSZ
LISTED
MTPY
MTPLY
INTCHR

ERRORS
FATAL
NONE
NONFATAL
NONE

3-192
LOCATE SATELLITE LOCATIONS IN PATH

TAIL TO SATELLITE?

NO

YES

DESCRIBE TAIL PATH

DEFINE SATELLITE LINK

TAIL FROM SATELLITE?

NO

YES

DESCRIBE TAIL PATH

Figure 3-78. OUT2SB Logic Flow (Sheet 1 of 2)
Figure 3-78. OUT2SB Logic Flow (Sheet 2 of 2)
3.7.8 PRTSBS Module

SUBROUTINE PRTSBS(VC,TARIFF,LUN)

PURPOSE
COMPUTE EQUIPMENT COST FOR SBS TARIFF SPECIFIED

INPUT ARGUMENTS
VC I*4 NUMBER OF VOICE CIRCUITS
TARIFF I*4 NUMBER OF TARIFF IN QUESTION
LUN I*4 LOGICAL UNIT NUMBER WHERE EQUIPMENT COSTS TO BE PRINTED

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
CPNA /SBS/ U
VCPTU /SBS/ U
NSCUPN /SBS/ U
NPTAB /SBS/ U
TSBBKP /SBS/ U
TSBCAB /SBS/ U
TSBCAI /SBS/ U
TSBCMN /SBS/ U
TSBNAC /SBS/ U
TSBFTU /SBS/ U
TSBSCU /SBS/ U
DEBUG /LUN/ U
NTARIF /RUN/ U

LOCAL VARIABLES
COST I*4 TOTAL COST
IEND I*4 MAXIMUM NUMBER OF ITERATIONS TO ASSIGN NVC TO CONFIGURATION OF 1 NAC AND UP TO NSCUPN SCUS
CPN I*4 MAXIMUM CAUS PER NAC (SCU)
I,J I*4 LOOP INDICES
NCAU I*4 NUMBER OF CAUS AT NODE
NFTU I*4 NUMBER OF NACS AT NODE
NNAC I*4 NUMBER OF NACS REQUIRED
NNAC(1) NUMBER OF NACS
NNAC(2)-NNAC(4) NUMBER OF SCUS
PREV I*4 LOWER BOUND OF NUMBER OF CAUS/NAC FOR BREAKPOINT IN RATE TABLES
TMPCMN R*8 MINIMUM CAU COST
MXPNAC I*4 MAXIMUM NUMBER OF CAU PER NAC/SCU CONFIGURATION
NLEFT I*4 NCAU LEFT WHEN ASSIGN CAU TO TOTAL NAC/SCU CONFIGURATION
N I*4 NUMBER OF NAC AND SCU REQUIRED FOR NLEFT
IC1  I*4    TARIFF LEVEL FOR NGRP GROUPS OF CAUS
IC2  I*4    TARIFF LEVEL FOR NLEFT CAUS
DIFF1 I*4    EXCESS CAUS FOR NGRP GROUPS
DIFF2 I*4    EXCESS CAUS FOR NLEFT CAUS
PRV1  I*4   PREVIOUS COUNT FOR NGRP GROUPS
PRV2  I*4   PREVIOUS COUNT FOR NLEFT CAUS
NGRP  I*4    NUMBER OF NACS AT NODE
SUBCST R*4    COST OF TARIFF ITEM
NSCU  I*4    NUMBER OF SCUS

SUBPROGRAMS CALLED
MTPY

ERRORS
FATAL
   6010    INPUT ARGUMENTS OUT OF RANGE
NONFATAL
   NONE

ORIGINAL PAGE IS
OF POOR QUALITY
Figure 3-79. PRTSBS Logic Flow
3.7.9 GETRTE Module

SUBROUTINE GETRTE(FROM, TO, RTE, NL)

PURPOSE
   DESCRIBE THE ROUTE BETWEEN TWO NODES IN TERMS OF ALL NODES INCLUDED IN THE ROUTE

INPUT ARGUMENTS
   FROM  I*4  FROM NODE
   TO    I*4  TO NODE

OUTPUT ARGUMENTS
   RTE(20) I*4  LIST OF NODES BETWEEN FROM AND TO INCLUDING FROM AND TO
   NL     I*4  NUMBER OF LINKS IN PATH BETWEEN FROM AND TO

DATA STRUCTURES
   (U - USED, S - DEFINED, R - READ)
   SVC    /CSTOUT/ U
   OROUTE /CSTOUT/ U
   NODPTR /SBSMXD/ U
   NACLOC /SBSMXD/ U
   DEBUG  /LUN/  U

LOCAL VARIABLES
   SATLTE  I*4  CONSTANT /2/ SATELLITE SERVICE
   CURFRM  I*4  CURRENT FROM NODE IN ROUTE
   SATFRM  I*4  TRANSMITTING NAC
   SATTO   I*4  RECEIVING NAC
   PART    I*4  PARTITION THAT NODE BELONGS TO
   CURTO   I*4  CURRENT TO NODE IN ROUTE
   IEND    I*4  LAST ITEM OF ARRAY TO BE PRINTED

SUBPROGRAMS CALLED
   NONE

ERRORS
   FATAL
      NONE
   NONFATAL
      NONE
Figure 3-80. GETRTE Logic Flow
3.7.10 OUT3 Module

SUBROUTINE OUT3

PURPOSE

PREPARE AND WRITE TABLE 3. LEAST-COST NETWORK

INPUT ARGUMENTS

NONE

OUTPUT ARGUMENTS

NONE

DATA STRUCTURES

(U - USED, S - DEFINED, R - READ)

RTCNTR /RNODES/ U
NACNMB /SBSMXD/ U
NACLOC /SBSMXD/ U
MINCPC /CSTOUT/ U
NETOIL /CSTOUT/ U
NNODES /RUN/ U
OPTION /CSTOUT/ U
SA /WU/ U
SVC /CSTOUT/ U
TF /RUN/ U
TRAFIC /TRFC/ U
QNAC /SBSEQP/ U
QSCU /SBSEQP/ U
QFTU /SBSEQP/ U
QCAU /SBSEQP/ U
LUNOUT /LUN/ U
OUTFLG /RUN/ U
SATNVC /SBSMX2/ U

LOCAL VARIABLES

DASH I*4 CONSTANT /'-'/
I I*4 LOOP INDEX - FROM CITIES
IPRT I*4 USED TO PRINT FROM CITY MNEMONICS
ISTEMP I*4 TAKES VALUES FROM SVC ARRAY - USED TO
ITEMP I*4 NODE NUMBER - USED IN TRACING ROUTES
J I*4 LOOP INDEX - TO CITIES
JPRT I*4 USED TO PRINT TO CITY MNEMONICS
JTEMP I*4 NODE NUMBER - USED IN SUMMING
LIINK TRAFFIC TOTALS
K I*4 LINK INDEX - USED IN PRINTING NON-
KPRT I*4 USED TO PRINT MNEMONICS ON NON-SATELLITE
ACCESS CITIES
ACCESS CITIES
NB I*4 NUMBER OF BASE GROUPS IN FACILITY
NJ I*4 NUMBER OF JUMBO GROUPS IN FACILITY
NM I*4 NUMBER OF IMASTER GROUPS IN FACILITY
NS I*4 NUMBER OF SUPER GROUPS IN FACILITY

3-200
SAT  C*16  CONSTANT / 'SATELLITE' /
SERVICE  C*16  VARIABLE SET EQUAL TO SAT OR TERR
STEMP  R*8  USED TO PRINT NAME OF TARIFF USED
TEMP  R*4  TOTAL COST OF CIRCUITS FOR PRINTING
TERR  C*16  CONSTANT / 'TERRESTRIAL' /
RNAME(20)  I*4  NAME OF NODES IN LEAST COST PATH
STLTE  I*4  CONSTANT /2/ SATELLITE
VC  I*4  NUMBER OF SBS VOICE CIRCUITS IN SBS MIXED
NVC  I*4  NUMBER OF SBS VOICE CIRCUITS
KK  I*4  LOOP INDEX
PTSB  L  .TRUE. IF DESCRIPTION OF SBS CHARGES TO BE PRINTED, OTHERWISE .FALSE.
ITR  I*4  TARIFF NUMBER
SRV  I*4  SERVICE TYPE
1  TERRESTRIAL
2  SATELLITE
SKIP  L  LINK SUMMARY NOT PRINTED BECAUSE EITHER OR BOTH NODES NOT ACCESS CITY IN WU ONLY RUN
RMIN  R*4  DOLLAR PART OF MULTIPLIER
RTMP  R*4  DOLLAR PART OF MULTIPLICAND
IMIN  I*4  CENTS PART OF MULTIPLIER
ITMP  I*4  CENTS PART OF MULTIPLICAND
CA1,CA2,CA3  I*2  CHARACTERS ASSOCIATED WITH INTEGER VALUES
CB1,CB2,CB3  I*2  CHARACTERS ASSOCIATED WITH INTEGER VALUES

SUBPROGRAMS CALLED
FACSIZ
MTPLY
PRTSBS
LISTED
OUT3SB

ERRORS
NONE
OUT3

PRINT HEADER

SBS MIX?

YES

NO

CALCULATE NETWORK VOLUMES FOR ALL LINKS

ALL PAIRS PROCESSED?

YES

RETURN

NO

SBS MIXED WITH SATELLITE?

YES

DESCRIBE PATH WITH SBS SATELLITE

NO

DESCRIBE EACH LINK IN PATH

SBS EQUIPMENT?

YES

DESCRIBE SBS EQUIPMENT

NO

Figure 3-81. OUT3 Logic Flow
3.7.11 OUT3SB Module

SUBROUTINE OUT3SB(FROM,FRMprt,TO,TOPRt,VC)

PURPOSE
PREPARE AND WRITE TABLE 3 LEAST COST NETWORK (INCLUSIVE OF OTHER NETWORK TRAFFIC) FOR LINKS WHICH INCLUDE SBS SATELLITE

INPUT ARGUMENTS
FROM I*4 FROM NODE 
FRMprt I*4 CITY MNEMONIC OF FROM NODE 
TO I*4 TO NODE 
TOPRt I*4 CITY MNEMONIC OF TO NODE 

OUTPUT ARGUMENTS
VC I*4 NUMBER OF VOICE CIRCUITS FROM NAC LOCATION 

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
RTCNTR /RNODS/ U 
LUNOUT /LUN/ U 
NETDNL /CSTOUT/ U 
MINCPC /CSTOUT/ U 
TF /RUN/ U 
NODPRT /SBSMXD/ U 
SATCST /SBSMX2/ U 
SATOTL /SATMXD/ U 
DEBUG /LUN/ U 
SATNVC /SBSMX2/ U 

LOCAL VARIABLES
DASH I*4 CONSTANT /'-'/ 
RTE(20) I*4 NODE NUMBERS OF EACH NODE IN PATH BETWEEN FROM AND TO 
NL I*4 NUMBER OF LINKS IN PATH BETWEEN FROM AND TO 
RNAME(20) I*4 NAME OF EACH LINK IN PATH 
I I*4 LOOP INDEX 
IPRt I*4 AS PROCEED ALONG PATH FROM NODE OF CURRENT LINK 
CURTO I*4 AS PROCEED ALONG PATH, TO NODE OF CURRENT LINK 
NVC I*4 NUMBER OF VOICE CIRCUITS ON LINK 
NJ I*4 NUMBER OF JUMBO GROUPS 
NM I*4 NUMBER OF MASTER GROUPS 
NS I*4 NUMBER OF SUPER GROUPS 
NB I*4 NUMBER OF BASE GROUPS 
SERVICE C*16 TYPE OF SERVICE ON LINK - SATELLITE OR TERRESTRIAL 
CVC R*4 COST PER VOICE CIRCUIT 
CURFRM I*4 AS PROCEED ALONG PATH, FROM NODE OF CURRENT LINK 

3-203
PTARIF  R*8   NAME OF TARIFF
SAT     C*16  CONSTANT /'SATELLITE'/
TERR    C*16  CONSTANT /'TERRESTRIAL'/
PART    I*4   PARTITION WHERE NAC IS LOCATED
PRTFRM  I*4   NAME OF FROM NODE OF CURRENT LINK
PRTTO   I*4   NAME OF TO NODE OF CURRENT LINK
IEND    I*4   LAST ITEM OF ARRAY TO BE PRINTED
SCHED   I*4   TERRESTRIAL TARIFF SCHEDULED
SRV     I*4   SERVICE TYPE
         1   TERRESTRIAL
         2   SATELLITE
RMIN    R*4   DOLLAR PORTION OF MULTIPLIER
IMIN    I*4   CENTS PORTION OF MULTIPLIER
RTMP    R*4   DOLLAR PORTION OF RESULT
ITMP    R*4   CENTS PORTION OF MULTIPLIER
CA1-CA3 I*4   CHARACTER REPRESENTATION OF IMIN
CB1-CB3 I*2   CHARACTER REPRESENTATION OF ITMP

SUBPROGRAMS CALLED
FACSIZ
GETRTE
LISTED
MTPY
MTPLY
INTCHR

ERRORS
FATAL
NONE
NONFATAL
NONE
Figure 3-82. OUT3SB Logic Flow
3.7.12 OUT4 Module

SUBROUTINE OUT4

PURPOSE

PREPARE AND WRITE TABLE 4. OUTPUT NETWORK TOTALS AND TABLE 5. TARIFF SUMMARY

INPUT ARGUMENTS

NONE

OUTPUT ARGUMENTS

NONE

DATA STRUCTURES

(U - USED, S- DEFINED, R - READ)

OUTFLG /RUN/ U
MINCPC /CSTOUT/ U
NETOTL /CSTOUT/ U
NNODES /RUN/ U
NTARIF /RUN/ U
OPTION /RUN/ U
SVC /CSTOUT/ U
TF /RUN/ U
TRRMLG /CSTMXD/ U
TRRCS /CSTMXD/ U
TRRCKT /CSTMXD/ U
STLMLG /CSTMXD/ U
STLCST /CSTMXD/ U
STLCKT /CSTMXD/ U
LUNOUT /LUN/ U
QNC /SBESEQ/ U
QSCU /SBESEQ/ U
QFTU /SBESEQ/ U
QCAU /SBESEQ/ U
SBMX /SBSMXD/ U
TOTSAT /SBSMX2/ U
NACNMB /SBSMXD/ U
STABLE /SBSMX2/ U
NACLOC /SBSMXD/ U
RTCNTR /RNOS/ U
SATNVC /SBSMX2/ U

LOCAL VARIABLES

I I*4 LOOP INDEX
USRDEF I*4 CONSTANT /1/ USER DEFINE NAC LOCATION
NITER I*4 NUMBER OF ITERATIONS OF SBSMXD ALGORITHM TO FIND OPTIMAL NUMBER OF NAC LOCATIONS
NLOC I*4 NUMBER OF NAC LOCATIONS
ISAT I*4 USED TO PRINT SATELLITE NETWORK TOTALS
ITERR I*4 USED TO PRINT TERRESTRIAL NETWORK TOTALS
J I*4 LOOP INDEX
PCKT R*4 PERCENT OF CIRCUITS FOR EACH TARIFF
PCOST \( R^{**4} \) PERCENT OF COST FOR EACH TARIFF
SAT \( C^{*16} \) CONSTANT / 'SATELLITE' /
SERVICE \( C^{*16} \) VARIABLE SET EQUAL TO SAT OR TERR
T \( I^{*2} \) TAKES VALUES OF EACH TARIFF SUCCESSIVELY TO CALCULATE TARIFF TOTALS
TCKTS \( I^{*4} \) ARRAY CONTAINING TOTAL CIRCUITS FOR EACH TARIFF
TCOSTS \( R^{*4} \) ARRAY CONTAINING TOTAL COSTS FOR EACH TARIFF
TERR \( C^{*16} \) CONSTANT / 'TERRESTRIAL' /
TMILES \( I^{*4} \) ARRAY CONTAINING TOTAL MILEAGE FOR EACH TARIFF
TOTCKT \( I^{*4} \) TOTAL NETWORK CIRCUITS
TOTCST \( R^{*4} \) TOTAL NETWORK COST
TOTMIL \( I^{*4} \) TOTAL NETWORK MILEAGE
TERCST \( R^{*4} \) TERRESTRIAL COST
CSTSAT \( R^{*4} \) SATELLITE COST
NREC \( I^{*4} \) NUMBER OF RECORDS OF NAC LOCATIONS TO PRINT
END \( I^{*4} \) LAST NAC IN CURRENT PRINT RECORD
BEGIN \( I^{*4} \) FIRST NAC IN CURRENT PRINT RECORD
NL \( I^{*4} \) NAC NUMBER
NITM \( I^{*4} \) NUMBER OF ITEMS IN PRINT LINE
REC(10) \( I^{*4} \) PRINT LINE
BIGNUM \( R^{*4} \) 16.0**62
N \( I^{*4} \) NUMBER OF SPECIFIED NACS WHICH ACTUALLY HAVE SATELLITE TRAFFIC

SUBPROGRAMS CALLED
MILES
SUMEQP
MTPY

ERRORS
FATAL
NONE
NONFATAL
NONE
Figure 3-83. OUT4 Logic Flow
3.7.13 SUMEQP Module

SUBROUTINE SUMEQP

PURPOSE
SUMMARY OF EARTH STATION EQUIPMENT FOR SBS MIXED MODE

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
NACNMB  /SBSMXD/  U
QNAC    /SBSEQP/  S
QSCU    /SBSEQP/  S
QFTU    /SBSEQP/  S
QCAU    /SBSEQP/  S
SATNVC  /SBSMX2/  U
VCPFTU  /SBS/    U
NSCUPN  /SBS/    U
CPNA    /SBS/    U
OPTION  /RUN/    U
NNODES  /RUN/    U

LOCAL VARIABLES
I     I*4  LOOP INDEX
NVC   I*4  NUMBER OF ORIGINATING VOICE CIRCUITS
NCAU  I*4  NUMBER CAUs REQUIRED
NFTU  I*4  NUMBER FTUs REQUIRED
MXPNAC I*4  MAXIMUM CAUs AT NAC/SCU CONFIGURATION
NGRP  I*4  NUMBER OF MAXIMUM NAC/SCU CONFIGURATIONS
NLEFT I*4  NUMBER OF CAUs NOT IN MAX CONFIGURATION
NADTNL I*4  NUMBER OF ADDITIONAL NAC, SCUs REQUIRED
NNAC  I*4  NUMBER OF NACs REQUIRED
MSCU  I*4  NUMBER OF SCUs REQUIRED
NMNBAC I*4  NUMBER OF NAC LOCATIONS

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
NONE
NONFATAL
NONE
Figure 3-34. SUMEQP Logic Flow
3.8 UTILITY FUNCTIONS

CNDC contains 28 general purpose modules which are utilized by many other modules in the model. Because they cannot be easily grouped with a single portion of the model, they are grouped together as utility functions. For the most part the functions they perform are single, simple functions on a set of input arguments.

They all follow the same format. When the utility function is invoked, its input arguments are validated. If an error is detected, the value of the input argument is out of range, an error message is printed and the model is terminated.

Since, for the most part, the logic within these modules is relatively simple, logic flow figures have not been included.
3.8.1 ACC Module

LOGICAL FUNCTION ACC(NODE, ITARIF)

PURPOSE
LOGICAL FUNCTION WHICH DETERMINES IF RATE CENTER AT NODE IS A SATELLITE ACCESS CITY FOR TARIFF SPECIFIED

INPUT ARGUMENTS
NODE I*4 NODE NUMBER OF A RATE CENTER
ITARIF I*4 TARIFF IN QUESTION

OUTPUT ARGUMENTS
ACC L LOGICAL FUNCTION VALUE

- TRUE. RATE CENTER AT NODE IS A SATELLITE ACCESS CITY FOR SPECIFIED TARIFF
- FALSE. RATE CENTER AT NODE IS NOT A SATELLITE ACCESS CITY FOR TARIFF SPECIFIED

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
NNODES /RUN/ U
NTARIF /RUN/ U
SA /WU/ U
SATACC /WU/ U
DEBUG /LUN/ U

LOCAL VARIABLES
IACC I*4 POINTER TO POSITION WITHIN WU STRUCTURES OF THIS INDICATES IF NODE SATELLITE ACCESS CITY FOR THIS TARIFF
TACC L FUNCTION VALUE

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL 5000 INPUT ARGUMENTS OUT OF RANGE
NONFATAL NONE
3.8.2 ATCOST Module

REAL FUNCTION ATCOST(MILAGE, ITARIF, ISCHED)

PURPOSE
DETERMINE COST PER CIRCUIT OF TERRESTRIAL SERVICE FOR TARIFF
AND SCHEDULE SPECIFIED AT GIVEN MILEAGE

INPUT ARGUMENTS
MILAGE I*4  DISTANCE OVER WHICH COST TO BE DETERMINED
ITARIF I*4  TARIFF FOR WHICH COST TO BE DETERMINED
ISCHED I*4  SCHEDULE FOR WHICH COST TO BE DETERMINED

OUTPUT ARGUMENTS
ATCOST R  COST PER CIRCUIT FOR TARIFF USING
SCHEDULE AT MILEAGE SPECIFIED

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
NTARIF /RUN/  U
NSCHED /ATT/  U
NPLEVL /ATT/  U
NPLEVL /ATT/  U
TATBAS /ATT/  U
TATINC /ATT/  U
TATMLG /ATT/  U
DEBUG /LUN/  U

LOCAL VARIABLES
I I*4  LOOP INDEX
MLPREV I*4  MILEAGE AT PREVIOUS BREAKPOINT
DLTMLG R  DIFFERENCE BETWEEN PREVIOUS BREAKPOINT
MILEAGE AND MILEAGE RATE DESIRED FOR
COST R  TARIFF COST

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
5010  INPUT ARGUMENTS OUT OF RANGE
NONFATAL
NONE
3.8.3 ATTLST Module

LOGICAL FUNCTION ATTLST(MNEM)

PURPOSE
DETERMINE IF A RATE CENTER SPECIFIED BY ITS FOUR CHARACTER CODE
IS ONE OF THE PRESTORED ATT CATEGORY A RATE CENTERS

INPUT ARGUMENTS
MNEM I*4 CODE OF RATE CENTER IN QUESTION

OUTPUT ARGUMENTS
ATTLST L .TRUE. MNEM IS A PRESTORED CATEGORY A RATE CENTER

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
DRCATA /INPATT/ U
MDIR /INPATT/ U
MPBLK /INPATT/ U
NCATA /INPATT/ U
PRCATA /INPATT/ U
DEBUG /LUN/ U

LOCAL VARIABLES
FOUND L . MNEM FOUND IN DIRECTORY OF CATA LIST
PTR I*4 POINTER TO ITEMS IN DIRECTORY OR CATA LIST BEING REFERENCED
END I*4 END OF PORTION OF CATA LIST TO BE SEARCHED
START I*4 BEGINNING OF PORTION OF CATA LIST TO BE SEARCHED

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL NONE
NONFATAL NONE
3.8.4 CATEG Module

INTEGER FUNCTION CATEG(NODE1,NODE2,ITARIF)

PURPOSE
DETERMINES THE SATELLITE ACCESS CATEGORY BETWEEN NODE1 AND
NODE2 FOR THE TARIFF SPECIFIED

INPUT ARGUMENTS
NODE1 I*4 FIRST NODE OF THE PAIR
NODE2 I*4 SECOND NODE OF THE PAIR
ITARIF I*4 TARIFF IN QUESTION

OUTPUT ARGUMENTS
CAT I*4 SATELLITE ACCESS CATEGORY BETWEEN NODE1
AND NODE2 FOR TARIFF SPECIFIED
0 NON ACCESS
1 LONG HAUL
2 MEDIUM HAUL
3 SHORT HAUL

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
NNODES /RUN/ U
NSATA2 /WU/ U
NTARIF /RUN/ U
SA /WU/ U
WUPRS /WU/ U
DEBUG /LUN/ U

LOCAL VARIABLES
WUA I*4 POINTER TO NODE IN WU STRUCTURES IF NODE
1 IS SATELLITE ACCESS CITY FOR ANY TARIFF
WUB I*4 POINTER TO NODE IN WU STRUCTURE IF NODE
2 IS SATELLITE ACCESS CITY FOR ANY TARIFF
PTR I*4 POSITION OF PAIR WITHIN WUPRS ARRAY
CAT I*4 RATE CATEGORY OF PAIR

SUBPROGRAMS CALLED
PTRPRS

ERRORS
FATAL
5020 INPUT ARGUMENTS OUT OF RANGE
NONFATAL
NONE
3.8.5 DEFPRE Module

SUBROUTINE DEFPRE

PURPOSE
DEFINE PRESTORED NETWORK

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
HNOD       /HEADRS/  U
NPRNDS     /PRSNDS/  S
PRNDS       /PRSNDS/  S
TRFNDS      /LUN/   U
DEBUG       /LUN/   U

LOCAL VARIABLES
HEADER(2) R*8   FILE HEADER
HT          I*4   HORIZONTAL COORDINATE
I           I*4   LOOP INDEX
VT          I*4   VERTICAL COORDINATE

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
5090 FILE HEADER OF PRESTORED NETWORK FILE NOT AS EXPECTED
NONFATAL
NONE
3.8.6 DIRGEN Module

SUBROUTINE DIRGEN.

PURPOSE
GET LIST OF FILENAMES IN NETWORK AND TARIFF DIRECTORIES

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
HNET(2) /HEADRS/ U
HTAR(2) /HEADRS/ U
NETDEF(20) /NETFIL/ R
NETDIR /LUN/ U
NFILES(20) /NETFIL/ R
NNFIL /NETFIL/ R
NTFIL /TARFIL/ R
TARDEF(13) /LUN/ U
TFILES(13) /TARFIL/ R
DEBUG /LUN/ U

LOCAL VARIABLES
HEADER(2) R*8 FILE HEADER READ
I I*4 LOOP INDEX

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
5100 FILE HEADER READ FROM TARIFF DIRECTORY NOT AS EXPECTED
5110 FILE HEADER READ FROM THE NETWORK DIRECTORY NOT AS EXPECTED

NONFATAL
NONE
3.8.7 FACSIZ Module

SUBROUTINE FACSIZ(VFS,NJUMBO,NMSTER,NSUPER,NBASE)

PURPOSE
DETERMINES FACILITY SIZE GIVEN THE NUMBER OF VOICE CIRCUITS

INPUT ARGUMENTS
VFS I*4 NUMBER OF VOICE CIRCUITS

OUTPUT ARGUMENTS
NJUMBO I*4 NUMBER OF JUMBO GROUPS
NMSTER I*4 NUMBER OF MASTER GROUPS
NSUPER I*4 NUMBER OF SUPER GROUPS
NBASE I*4 NUMBER OF BASE GROUPS

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
DEBUG /LUN/ U

LOCAL VARIABLES
NLEFT I*4 NUMBER OF VOICE CIRCUITS REMAINING TO BE ATTRIBUTED TO A GROUP

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
5120 NUMBER OF INPUT VOICE CIRCUITS OUT OF RANGE
NONFATAL
NONE
3.8.8 FILDEF Module

INTEGER FUNCTION FILDEF(DIR,INFIL)

PURPOSE
 INTEGER FUNCTION WHICH DETERMINES IF FILENAME SPECIFIED IS IN
 CURRENT DIRECTORY (NETWORK OR TARIFF)

INPUT ARGUMENTS
 DIR I*4 DIRECTORY TO BE SEARCHED
       1 NETWORK DIRECTORY
       2 TARIFF DIRECTORY
 INFIL I*4 FILE NAME TO BE SEARCHED FOR IN DIRECTORY

OUTPUT ARGUMENTS
 FILDEF I*4 INTEGER INDICATOR OF STATUS OF FILNAM
        1 FILE NAME INVALID
        2 VALID NAME BUT FILE UNDEFINED
        3 VALID NAME AND FILE IS DEFINED

DATA STRUCTURES
 (U - USED, S - DEFINED, R - READ)
 NNFIL /NETFIL/ U
 NFILES /NETFIL/ U
 NETDEF /NETFIL/ U
 NTFIL /TARFIL/ U
 TFILES /TARFIL/ U
 TARDEF /TARFIL/ U
 DEBUG /LUN/

LOCAL VARIABLES
 FOUND LOGICAL INDICATES IF NAME FOUND IN DIRECTORY
 I I*4 LOOP INDEX
 IVAL I*4 STATUS OF FILE IF IN DIRECTORY

SUBPROGRAMS CALLED
 NONE

ERRORS
 FATAL DIRECTORY REQUESTED IS NOT VALID
  5150
 NONFATAL NONE
3.8.9 INTCHR Module

SUBROUTINE INTCHR(INVAL,C1,C2,C3)

PURPOSE
CONVERT RIGHT MOST THREE DIGITS OF THE INPUT INTEGER VALUE TO CHARACTER REPRESENTATION.

INPUT ARGUMENTS
INVAL I*4 INPUT INTEGER

OUTPUT ARGUMENTS
C1 I*2 CHARACTER REPRESENTATION OF DIGIT IN HUNDREDS POSITION
C2 I*2 CHARACTER REPRESENTATION OF DIGIT IN TENS POSITION
C3 I*2 CHARACTER REPRESENTATION OF DIGIT IN ONES POSITION

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
DEBUG /LUN/ U

LOCAL VARIABLES
ITRNK I*4 INTEGER VALUE OF RIGHT MOST THREE DIGITS OF INPUT VALUE
I1 I*4 INTEGER VALUE OF DIGIT IN HUNDREDS POSITION
I2 I*4 INTEGER VALUE OF DIGIT IN TENS POSITION
I3 I*4 INTEGER VALUE OF DIGIT IN ONES POSITION

SUBPROGRAMS CALLED
NONE

ERRORS
NONE
3.8.10 LISTED Module

INTEGER FUNCTION LISTED(NODE, ITARIF)

PURPOSE
DETERMINES IF THE NODE SPECIFIED IN A CATEGORY A FOR TARIFF
SPECIFIED

INPUT ARGUMENTS
NODE I*4 INTEGER NUMBER OF NODE OF THE RATE CENTER
IN QUESTION
ITARIF I*4 INTEGER NUMBER OF THE TARIFF IN QUESTION

OUTPUT ARGUMENTS
LISTED I INDICATOR OF CATEGORY OF INPUT NODE
1 CATEGORY B UNLISTED
0 CATEGORY A LISTED

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
ALSTD /ATT/ U
LSTED /ATT/ U
NINLST /ATT/ U
NNODES /RUN/ U
NTARIF /RUN/ U
TPTR /ATT/ U
DEBUG /LUN/ U

LOCAL VARIABLES
IATT I*4 CATA STATUS OF THIS NODE IN PRESTORED
TARIFF
PTR I*4 WHERE DESCRIPTION OF THIS TARIFF STARTS
NUMB I*4 NUMBER OF ITEMS IN LIST THAT DESCRIBES
THIS TARIFF
END I*4 NUMBER OF LAST ITEM IN LIST THAT
DESCRIBES THIS TARIFF

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
5030 INPUT ARGUMENTS OUT OF RANGE
NONFATAL
NONE
INTEGER FUNCTION MILES(NODE1,NODE2)

PURPOSE
CALCULATES THE AIRLINE MILEAGE BETWEEN NODE1 AND NODE2 USING
METHOD DESCRIBED IN SECTION 3.2 OF FCC TARIFF NUMBER 264

INPUT ARGUMENTS
NODE1   I*4  INTEGER NUMBER OF FIRST NODE
NODE2   I*4  INTEGER NUMBER OF SECOND NODE

OUTPUT ARGUMENTS
MILES   I*4  AIRLINE MILEAGE BETWEEN NODE1 AND NODE2

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
H   /RNODES/  U
V   /RNODES/  U
NNODES  /RUN/  U
DEBUG  /LUN/  U

LOCAL VARIABLES
IRL   I*4  INTEGER DISTANCE BETWEEN TWO NODES
RL   R*4  REAL DISTANCE BETWEEN TWO NODES

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
5040  INPUT ARGUMENTS OUT OF RANGE
NONFATAL
NONE
3.8.12 MNMPTR Module

INTEGER FUNCTION MNMPTR(MNEM)

PURPOSE
    INTEGER FUNCTION WHICH FINDS NODE NUMBER WITHIN USER
    DEFINED NETWORK OF THE NODE WITH CODE MNEM

INPUT ARGUMENTS
    MNEM         I*4    FOUR CHARACTER CODE OF RATE CENTER
                  IN QUESTION

OUTPUT ARGUMENTS
    MNMPTR       I*4    INTEGER NODE NUMBER OF RATE CENTER WITH
                      CODE MNEM. MNMPTR MUST BE IN THE RANGE
                      0 TO NNODES. IF MNMPTR=0, THEN MNEM IS
                      NOT A VALID CODE IN THE CURRENT RUN

DATA STRUCTURES
    (U - USED, S - DEFINED, R - READ)
    NNodes        /RUN/  U
    RtcTr         /RNODES/  U
    NdIR          /RNODES/  U
    Npblk         /RNODES/  U
    Dnnode        /RNODES/  U
    Debug         /Lun/   U

LOCAL VARIABLES
    Ptr           I*4    RATE CENTER IN DIRECTORY OR RATE CENTER
                    LIST UNDER CONSIDERATION
    Start         I*4    START POSITION OF SEARCH IN LIST
    End           I*4    END POSITION OF SEARCH IN LIST
    Found         I*4    .TRUE. IF MNEM FOUND IN LIST OF DIRECTORY

SUBPROGRAMS CALLED
    NONE

ERRORS
    FATAL
        NONE
    NONFATAL
        NONE
3.8.13 MTPLY Module

SUBROUTINE MTPLY(A1,A2,M1,M2,R1,R2)

PURPOSE
MULTIPLY REAL COST ROUNDED TO TWO DECIMAL PLACES BY INTEGER QUANTITY

INPUT ARGUMENTS
A1 R*4 COST
A2 I*4 QUANTITY

OUTPUT ARGUMENTS
M1 R*4 DOLLAR PORTION OF A1
M2 I*4 CENTS PORTION OF A1
R1 R*4 DOLLAR PORTION OF RESULTS OF M1*A2
R2 I*4 CENTS PORTION OF RESULT OF M2*A2

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
DEBUG /LUN/ U

LOCAL VARIABLES
P R*4 CENTS PORTION OF M1 TO THREE DECIMAL PLACES PLUS 5 ADDED IN FOR ROUNDING

SUBPROGRAMS CALLED
NONE

ERRORS
NONE
REAL FUNCTION MTPY(ARG1,ARG2)

PURPOSE
MULTIPLY REAL COST ROUNDED TO TWO DECIMAL PLACES BY INTEGER QUANTITY

INPUT ARGUMENTS
ARG1 R*4 COST
ARG2 I*4 QUANTITY

OUTPUT ARGUMENTS
MTPY R*4 ROUNDED RESULT OF MULTIPLICATION

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
DEBUG /LUN/ U

LOCAL VARIABLES
RARG2 R*4 INTEGER ARGUMENT EXPRESSED AS REAL
PART1 R*4 WHOLE NUMBER PART OF REAL ARGUMENT
PART2 R*4 DECIMAL PORTION OF REAL ARGUMENT
FULL R*4 PRODUCT OF ARG1 ROUNDED TO 2 DECIMAL PLACES AND ARG2
AARG R*4 ABSOLUTE VALUE OF REAL ARGUMENT

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
NONE
NONFATAL
NONE
3.8.15 PTRPRS Module

INTEGER FUNCTION PTRPRS(NODE1,NODE2,NUM)

PURPOSE
GETS CORRECT POINTEK FOR EACH NODE PAIR SPECIFIED BY NODE1 AND NODE2 INTO A ONE DIMENSIONAL ARRAY WHICH REPRESENTS THE TRIANGLE ABOVE THE DIAGONAL FROM THE UPPER LEFT TO THE LOWER RIGHT CORNERS OF A NUM*NUM MATRIX

INPUT ARGUMENTS
NODE1  I*4  INTEGER NUMBER OF THE FIRST NODE
       MUST BE IN THE RANGE OF 1 AND NUM
NODE2  I*4  INTEGER NUMBER OF THE SECOND NODE
       MUST BE IN THE RANGE OF 1 AND NUM
NUM    I*4  NUMBER OF NODES IN THE MATRIX

OUTPUT ARGUMENTS
PTRPRS I*4  INTEGER POINTER DESIRED

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
DEBUG /LUN/ U

LOCAL VARIABLES
FIRST  I*4  INPUT NODE WHICH IS THE SMALLEST
SECOND I*4  DISTANCE INTO FINAL ROW OF POINTER
PKUA   I*4  DESIRED POSITION IN ONE DIMENSIONAL
        ARRAY
IEND   I*4  NUMBER OF INTERATIONS REQUIRED TO DETER-
          MINE DESIRED POSITION
I      I*4  LOOP INDEX

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
5050  ARGUMENTS OUT OF RANGE

NONFATAL
NONE
3.8.16 ROWTOT Module

INTEGER FUNCTION ROWTOT(NODE)

PURPOSE
SUM SPECIFIED ROW OF TRAFFIC TABLE

INPUT ARGUMENTS
NODE I*4 ROW TO BE SUMMED

OUTPUT ARGUMENTS
ROWTOT I*4 SUM OF ROW NODE OF TRAFFIC TABLE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
NNODES /RUN/ U
TRAFIC /TRFC/ U
DEBUG /LUN/ U

LOCAL VARIABLES
TOTAL I*4 SUM OF ROW
I I*4 LOOP INDEX

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
5060 INPUT ARGUMENT OUT OF RANGE
NONFATAL
NONE
INTEGER FUNCTION RVTFC(FROM, TO)

PURPOSE
REVIEW TRAFFIC BETWEEN ANY PAIR OF NODES IN USER DEFINED NETWORK

INPUT ARGUMENTS
FROM I*4 CODE OF FROM NODE
TO I*4 CODE OF TO NODE

OUTPUT ARGUMENTS
RVTFC I*4 NUMBER OF VOICE CIRCUITS BETWEEN FROM AND TO

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
TRAFFIC /TRFC/ U
DEBUG /LUN/ U
NNODES /RUN/ U

LOCAL VARIABLES
IFROM I*4 POINTER TO FROM NODE IN NETWORK
ITO I*4 POINTER TO TO NODE IN NETWORK
NVC I*4 NUMBER OF VOICE CIRCUITS

SUBPROGRAMS CALLED
MNMPTR

ERRORS
FATAL
5130 NODES SPECIFIED DON'T EXIST IN USER DEFINED NETWORK
NONFATAL
NONE
REAL FUNCTION SBCOST(VC,TARIFF)

PURPOSE
DETERMINE THE COST PER CIRCUIT FOR SBS TARIFF SPECIFIED

INPUT ARGUMENTS
VC I*4 NUMBER OF VOICE CIRCUITS
TARIFF I*4 NUMBER OF TARIFF IN QUESTION

OUTPUT ARGUMENTS
SBCOST R COST PER CIRCUIT FOR SBS TARIFF SPECIFIED

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)

CPNA /SBS/ U
VCPFTU /SBS/ U
NSCUPN /SBS/ U
MAXGRP /SBS/ U
NPTAB /SBS/ U
TSBBKP /SBS/ U
TSBCAB /SBS/ U
TSBCAI /SBS/ U
TSCCMN /SBS/ U
TSSNAC /SBS/ U
TSBFTU /SBS/ U
TSBSCU /SBS/ U
DEBUG /RUN/ U
NTARIF /RUN/ U

LOCAL VARIABLES
COST R*4 TOTAL COST
IEND I*4 MAXIMUM NUMBER OF ITERATIONS TO ASSIGN
VC TO CONFIGURATION OF 1 NAC AND UP TO
NSCUPN SCUS
CPN I*4 MAXIMUM OF CAUS PER NAC (SCU)
CSTCAU R*4 COST FOR CAUS
I,J I*4 LOOP INDICES
NCAU I*4 NUMBER OF CAUS AT NODE
NFTU I*4 NUMBER OF FTUS REQUIRED
NGRP I*4 NUMBER OF NACS AT NODE
NNAC I*4 NUMBER OF NACS REQUIRED
NNAC(1) NUMBER OF NACS
NNAC(2)-NNAC(4) NUMBER OF SCUS
PREV I*4 LOWER BOUND OF NUMBER OF CAUS/NAC FOR
BREAKPOINT IN RATE TABLES
TCAU R*4 CAU COST AT ONE NAC/SCU CONFIGURATION
TMPCMN R*4 MINIMUM CAU COST
MXPNAC I*4 MAXIMUM NUMBER OF CAU PER NAC/SCU
CONFIGURATION

3-229
NLEFT  I*4    NCAU LEFT WHEN ASSIGN CAU TO TOTAL NAC/SCU CONFIGURATION
N      I*4    NUMBER OF NAC AND SCU REQUIRED FOR NLEFT

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
5070    INPUT ARGUMENTS OUT OF RANGE
NONFATAL
NONE
3.8.19 SMSA Module

LOGICAL FUNCTION SMSA(MNEM)

PURPOSE
LOGICAL FUNCTION WHICH DETERMINES IF NODE MNEM IS WITHIN THE
PRESTORED NETWORK

INPUT ARGUMENTS
MNEM I*4 FOUR CHARACTER CODE OF RATE CENTER
IN QUESTION

OUTPUT ARGUMENTS
SMSA LOGICAL .TRUE. MNEM IS A PRESTORED NODE
.FALSE. MNEM IS NOT A PRESTORED NODE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
NPRNDS /PRSNDS/ U
PRNDS /PRSNDS/ U
KDIR /PRSNDS/ U
KPBLK /PRSNDS/ U
KRNODE /PRSNDS/ U
DEBUG /LUN/ U

LOCAL VARIABLES
PTR I*4 RATE CENTER IN DIRECTORY OR RATE CENTER
LIST UNDER CONSIDERATION
START I*4 START POSITION OF SEARCH IN LIST
END I*4 END POSITION OF SEARCH IN LIST
FOUND L .TRUE. IF MNEM FOUND IN LIST OR DIRECTORY

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
NONE
NONFATAL
NONE
3.8.20 SORT1 Module

SUBROUTINE SORT1(CODE, ORDER, NEL)

PURPOSE
SORTS THE NEL ELEMENTS IN CODE ARRAY IN ASCENDING ORDER

INPUT ARGUMENTS
CODE  I*4 INTEGER ARRAY TO BE READ
NEL   I*4 NUMBER OF ELEMENTS TO BE SORTED

OUTPUT ARGUMENTS
ORDER I*4 POSITION OF ITEMS IN CODE IN ASCENDING ORDER; IT IS ASSUMED THAT ORDER HAS BEEN INITIALIZED BY CALLING ROUTINE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
NONE

LOCAL VARIABLES
INDX1 I*4 POINTER TO FIRST OF TWO ITEMS IN CODE TO BE COMPARED
INDX2 I*4 POINTER TO SECOND OF TWO ITEMS IN CODE TO BE COMPARED
IEND  I*4 NUMBER OF COMPARISONS
SORTED L INDICATES IF LIST FULLY SORTED
I    I*4 LOOP INDEX

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
NONE
NONFATAL
NONE
SUBROUTINE SORTID(CODE,ORDER,NEL)

PURPOSE
SORT THE NEL ELEMENTS IN CODE IN DESCENDING ORDER

INPUT ARGUMENTS
NEL I*4 NUMBER OF ELEMENTS TO BE SORTED
CODE I*4 ARRAY OF ELEMENTS TO BE SORTED

OUTPUT ARGUMENTS
ORDER I*4 POINTER TO ITEMS IN CODE ARRAY IN SORTED ORDER

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
NONE

LOCAL VARIABLES
INDX1 I*4 POINTER TO FIRST OF TWO ITEMS IN CODE TO BE COMPARED
INDX2 I*4 POINTER TO SECOND OF TWO ITEMS IN CODE TO BE COMPARED
IEND I*4 NUMBER OF COMPARISONS
SORTED L INDICATES IF LIST FULLY SORTED
I I*4 LOOP INDEX

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL NONE
NONFATAL NONE
3.8.22 SORT2 Module

SUBROUTINE SORT2(CODEL,CODER,ORDER,NEL)

PURPOSE
SORTS THE NEL ITEMS IN CODEL AND CODER IN ASCENDING ORDER BY
CODEL AND THEN BY CODER WITHIN CODEL

INPUT ARGUMENTS
NEL I*4 NUMBER OF ITEMS IN ARRAYS TO BE SORTED
CODEL I*4 ARRAY TO BE SORTED
CODER I*4 ARRAY TO BE SORTED WITHIN CODEL

OUTPUT ARGUMENTS
ORDER I*4 ARRAY OF POINTER TO CODEL, CODER PAIR IN
ASCENDING ORDER

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
NONE

LOCAL VARIABLES
INDEX1 I*4 POINTER TO FIRST OF TWO ITEMS IN CODEL TO
BE COMPARED
INDEX2 I*4 POINTER TO SECOND OF TWO ITEMS IN CODEL TO
BE COMPARED
IEND I*4 NUMBER OF COMPARISONS
SORTED L INDICATES IF LIST FULLY SORTED
I I*4 LOOP INDEX

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL NONE
NONFATAL NONE
3.8.23 TARTYP Module

INTEGER FUNCTION TARTYP(TARIFF)

PURPOSE
DETERMINES THE TYPE OF THE TARIFF (ATT, WU, SBS) GIVEN THE TARIFF NUMBER

INPUT ARGUMENTS
TARIFF I*4 NUMBER OF TARIFF

OUTPUT ARGUMENTS
TARTYP I*4 TARIFF TYPE

1 ATT
2 WU
3 SBS

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
NTARIF /RUN/ U
OPTION /RUN/ U
DEBUG /LUN/ U

LOCAL VARIABLES
TYPE I*4 TARIFF TYPE

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
5080 INPUT ARGUMENT OUT OF RANGE
NONFATAL
NONE
3.8.24 **TRFSKP Module**

**SUBROUTINE TRFSKP(NREC,UNIT)**

**PURPOSE**
READ THE SPECIFIED NUMBER OF RECORDS FROM THE SPECIFIED LOGICAL UNIT NUMBER

**INPUT ARGUMENTS**
- **NREC** I*4 NUMBER OF RECORDS TO READ
- **UNIT** I*4 LOGICAL UNIT NUMBER OF FILE TO BE READ

**OUTPUT ARGUMENTS**
NONE

**DATA STRUCTURES**
- **DEBUG /LUN/ U**

**LOCAL VARIABLES**
- **I** I*4 LOOP INDEX
- **WORD** I*4 FIRST WORD ON RECORD

**ERROR**
NONE
3.8.25 USRNET Module

LOGICAL FUNCTION USRNET(MNEM)

PURPOSE
DETERMINE IF A RATE CENTER SPECIFIED BY ITS 4 CHARACTER CODE IS ONE
OF THE NODES IN THE CURRENTLY DEFINED NETWORK

INPUT ARGUMENTS
MNEM I*4 4 CHARACTER CODE OF NODE

OUTPUT ARGUMENTS
USRNET LOGICAL .TRUE. MNEM IS IN THE USER DEFINED
NETWORK
FALSE. MNEM IS NOT IN THE NETWORK

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
DEBUG /LUN/ U

LOCAL VARIABLES
NET LOGICAL .TRUE. MNEM IS IN THE USER DEFINED
NETWORK
FALSE. MNEM IS NOT IN THE USER DEFINED
NETWORK
PTR I*4 POINTER TO MNEM IN THE NETWORK

SUBPROGRAMS CALLED
MNMPTR

ERRORS
FATAL NONE
NONFATAL NONE
3.8.26 WRITRF Module

SUBROUTINE WRITRF(UNIT)

PURPOSE
PRINT INTO FILE SPECIFIED BY UNIT, CURRENT TRAFFIC TABLE

INPUT ARGUMENTS
UNIT       I*4    LOGICAL UNIT NUMBER OF FILE TO CONTAIN LISTING OF TRAFFIC TABLE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
NNODES  /RUN/   U
RTCNTR  /RNODES/ U
TRAFIC  /TRFC/  U
DEBUG   /LUN/   U

LOCAL VARIABLES
I, J, K       I*4    LOOP INDICES
NBK           I*4    NUMBER OF BLOCKS OF TRAFFIC TABLE VALUES TO BE PRINTED
HEADER(5)    I*4    HEADER LINE OF TO RATE CENTERS IN THIS BLOCK
TFC(5)        I*4    TRAFFIC VALUES FOR CURRENT PRINT LINE
BEG           I*4    INDEX OF FIRST TO RATE CENTER IN THIS BLOCK
LAST          I*4    INDEX OF LAST TO RATE CENTER IN THIS BLOCK
LAST          I*4    LAST ELEMENT IN HEADER OR TFC TO BE FILLED FOR THIS BLOCK
NPLIN         I*4    CONSTANT NUMBER OF VALUES PER PRINT LINE

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
NONE
NONFATAL
NONE
REAL FUNCTION WUCOST(CAT,TARIFF)

PURPOSE
DETERMINES THE WU COST PER CIRCUIT GIVEN THE RATE CATEGORY
OF THE CITY PAIR AND THE DESIRED TARIFF

INPUT ARGUMENTS
CAT I*4 RATE CATEGORY OF THE CITY PAIR
1 LONG HAUL
2 MEDIUM HAUL
3 SHORT HAUL

TARIFF I*4 NUMBER OF THE TARIFF

OUTPUT ARGUMENTS
WUCOST R COST PER CIRCUIT GIVEN THE RATE CATEGORY
OF THE CITY PAIR AND THE DESIRED TARIFF

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
TWU /WU/ U
NTARIFF /RUN/ U
NWUCAT /WU/ U
DEBUG /LUN/ U

LOCAL VARIABLES
COST R WU COST PER CIRCUIT

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
5140 INPUT ARGUMENTS OUT OF RANGE
NONFATAL
NONE
3.8.28 WUPAIR Module

INTEGER FUNCTION WUPAIR(MNEM1, MNEM2)

PURPOSE
DETERMINE IF RATE CENTERS SPECIFIED ARE AN ACCESS CITY PAIR

INPUT ARGUMENTS
MNEM1 I*4 FOUR CHARACTER CODE OF RATE CENTER
MNEM2 I*4 FOUR CHARACTER CODE OF SECOND RATE CENTER

OUTPUT ARGUMENTS
WUPAIR LOGICAL FUNCTION VALUE OF .TRUE. IF PAIR
SPECIFIED IS SATELLITE ACCESS CITY PAIR IN PRESTORED WU TARIFF

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
NACC /PRSWU/ U
LSTACC /PRSWU/ U
WUPRS /WU/ U
DEBUG /LUN/ U

LOCAL VARIABLES
RPTR I*4 POINTER OF MNEM1 IN LSTACC IF IT IS THERE
LPTR I*4 POINTER OF MNEM2 IN LSTACC IF IT IS THERE
PAIR I*4 RATE CATEGORY OF PAIR IF THEY ARE A SATELLITE ACCESS CITY PAIR, ELSE 0
I I*4 LOOP INDEX
PTR I*4 POINTER OF MNEM1, MNEM2 RATE CATEGORY IN WUPRS IF THEY ARE IN ACCESS CITYPAIR

SUBPROGRAMS CALLED
PTRPRS

ERRORS
FATAL NONE
NONFATAL NONE
3.9 BATCH DESCRIPTION

As described earlier, the CNDC model has two phases: one in which the user may define networks and tariffs to model and the second in which the optimization of a selected network over specified tariffs takes place. The input phase is conducted through an interaction between the user and the model, which guides the user through network and/or tariff building by offering choices, posing questions, and evaluating responses. The model may also be run in a batch rather than an interactive environment. To provide this capability, several of the model modules which communicate with the user, asking questions and evaluating responses, were modified to have them simply read the input from a file or card deck.

The user may execute the model in batch mode or may create network and/or tariff files. Several of the services provided by the input function of the interactive version, however, are not available in the batch version. The user may not delete or list files, but may only create files. The batch version will read and evaluate the input and create the specified tariff and/or network files if there are no errors. If there are errors, however, descriptive error messages are printed and the desired files are not created. The user must simply correct the input using the error messages and resubmit the job.

Figure 3-85 contains a hierarchy chart of the modules which perform the batch function. The modules specified in broken boxes are those from the main model, described previously. Below is a list of the batch modules described in this section. Next to each is the name of its interactive counterpart.

<table>
<thead>
<tr>
<th>BATCH MODULE</th>
<th>INTERACTIVE COUNTERPART</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNDCB</td>
<td>CNDC</td>
</tr>
<tr>
<td>INPUTB</td>
<td>INPUT, NETCON, TARCON, EXCON</td>
</tr>
<tr>
<td>NETB</td>
<td>CREATE</td>
</tr>
<tr>
<td>TARB</td>
<td>CREATTT</td>
</tr>
<tr>
<td>CRATTB</td>
<td>TCRATT</td>
</tr>
<tr>
<td>BATCH MODULE</td>
<td>INTERACTIVE COUNTERPART</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------------------</td>
</tr>
<tr>
<td>CRWUB</td>
<td>TCRWU</td>
</tr>
<tr>
<td>CRSBSB</td>
<td>TCRSBS</td>
</tr>
<tr>
<td>EXCONB</td>
<td>XCREAT</td>
</tr>
</tbody>
</table>
Figure 3-85. BATCH Hierarchy Chart
3.9.1 CNDCB Module

PURPOSE
DIRECT OVERALL PROCESSING OF BATCH VERSION OF CNDC MODEL

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
NCASE /RUN/ S

LOCAL VARIABLES
FLAG LOGICAL INDICATES WHETHER TO CONTINUE PROCESSING
OR TERMINATE
.TRUE. CONTINUE
.FALSE. TERMINATE

SUBPROGRAMS CALLED
INPUTB
INIT
COSTNG
NETWRK
OUTPUT
SBSMIX

ERRORS
FATAL
NONE
NONFATAL
NONE
Figure 3-86. CNDCB Logic Flow

3-245
3.9.2 INPUTB Module

SUBROUTINE INPUTB(FLAG)

PURPOSE
READ BATCH INPUTS AND CREATE NETWORK, TARIFF, AND EXECUTION CONTROL FILES

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
FLAG LOGICAL INDICATES WHETHER TO CONTINUE PROCESSING OR TERMINATE
.TRUE. CONTINUE
.FALSE. TERMINATE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
DEBUG /LUN/ U
LUNOUT /LUN/ U

LOCAL VARIABLES
IOPT I*4 OPTION CODE DEFINING TYPE OF INPUT SECTION
1 - CREATE NETWORK FILE
2 - CREATE TARIFF FILE
3 - CREATE EX CON FILE
4 - EXECUTE MODEL
5 - TERMINATE MODEL

SUBPROGRAMS CALLED
BSTRCT
NINIT
NETB
NCLOSE
TINIT
TARB
TCLOSE
EXCONB

ERRORS
FATAL
B1000 INVALID OPTION CODE
NONFATAL
NONE
Figure 3-87. INPUTB Logic Flow
3.9.3 NETB Module

SUBROUTINE NETB

PURPOSE
READ NECESSARY INPUTS AND CREATE A NETWORK FILE

INPUT ARGUMENTS
NONE

OUTPUT ARGUMENTS
NONE

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
CROPT /INPNET/ S
NNDS /INPNOD/ S
CODE /INPNOD/ S
NPRNDS /PRSND/ U
TOTNDS /INPNET/ S
PERCNT /INPNET/ S
LVLTRF /INPNET/ S
VC /INPNET/ S
HC /INPNET/ S
MAXUSR /INPNET/ U
NTRFC /INPNET/ S
CODEL /INPNOD/ S
CODER /INPNOD/ S
VCIRTS /INPNET/ S
DEBUG /LUN/ U
MAXNDS /INPNOD/ U
LUNOUT /LUN/ U

LOCAL VARIABLES
ERROR L INDICATES WHETHER A FATAL ERROR HAS BEEN FOUND: .TRUE. THERE IS A FATAL ERROR .FALSE. NO FATAL ERROR
FILNAM R*8 NAME OF NETWORK FILE TO BE CREATED
IANS I*4 0 OR 1 TO HAVE TRAFFIC TABLE SPOOLED
ITEMP I*4 TEMPORARY LOCATION OF INTEGER RESPONSES
NULL I*4 CONSTANT <CR>
I I*4 LOOP INDEX
TEMP R*4 TEMPORARY LOCATION OF REAL RESPONSE
NADD I*4 NUMBER OF ADDITIONAL CITIES ADDED
BEGIN I*4 START LOCATION WITHIN CODE FOR ADDITIONAL CITIES
IVC I*4 VERTICAL COORDINATE
IHC I*4 HORIZONTAL COORDINATE
NFROM I*4 WHEN REQUESTING INDIVIDUAL TRAFFIC VOLUMES THE FROM CODE
NTO I*4 WHEN REQUESTING INDIVIDUAL TRAFFIC VOLUMES THE TO CODE
TRFOPT L INDICATES IF LISTING OF TRAFFIC TABLE IS DESIRED
MAXINT R*4 CONSTANT/2147483647./LARGEST I*4 VALUE
FILSTT I*4 FILE STATUS
  1 INVALID NAME
  2 UNAVAILABLE
  3 IN USE
SUBPROGRAMS CALLED
  DEFNOD
  FILDEF
  GETPER
  GETTFC
  NCREAT
  SMSA
  USRNET
ERRORS
FATAL
  B2010 INVALID NETWORK FILENAME
  B2020 NETWORK FILE ALREADY DEFINED
  B2030 INVALID NETWORK OPTION
  B2040 INVALID PRESTORED CITY CODE
  B2050 EXCEEDED # OF PRESTORED NODES
  B2060 INVALID PERCENT OF SELECTING PRESTORE NODES
  B2070 INVALID TRAFFIC LEVEL
  B2080 MAXIMUM ADDITIONAL CITIES EXCEEDED
  B2090 INVALID ADDITIONAL TRAFFIC VOLUME
  B2100 ADDITIONAL TFC ENTRY CITY NOT IN NETWORK
  B2110 TRAFFIC TABLE SPOOL OPTION NOT 0 OR 1
  B2120 INVALID ADDITIONAL CITY COORDINATE.
NONFATAL
  B2500 PRESTORED NODE DUPLICATION?
  B2510 ADDITIONAL CITY A PRESTORED NODE
  B2520 ADDITIONAL CITY DUPLICATION
  B2530 ADDITIONAL TFC PAIR DUPLICATION
FOR LOGIC FLOW, SEE FIGURE 3-26 - CREATE LOGIC FLOW.
3.9.4 **TARB Module**

**SUBROUTINE TARB**

**PURPOSE**
READ NECESSARY INPUTS AND CREATE A TARIFF FILE

**INPUT ARGUMENTS**
NONE

**OUTPUT ARGUMENTS**
NONE

**DATA STRUCTURES**
(U - USED, S - DEFINED, R - READ)
DEBUG /LUN/ U
LUNOUT /LUN/ U

**LOCAL VARIABLES**
ERROR LOGICAL INDICATES WHETHER A FATAL ERROR HAS BEEN FOUND: .TRUE. FATAL ERROR .FALSE. NO FATAL ERROR
FILNAM R*8 NAME OF TARIFF FILE TO CREATE
FILSTT I*4 FILE STATUS
  1 - INVALID NAME
  2 - AVAILABLE
  3 - IN USE
TYPE I*4 TYPE OF TARIFF TO CREATE
  1 - ATT; 2 - WU; 3 - SBS

**SUBPROGRAMS CALLED**
CRATTB
CRWUB
CRSBSB
FILDEF
TCREAT

**ERRORS**
NORMAL
B3000 INVALID TARIFF FILENAME
B3010 TARIFF FILE ALREADY DEFINED
B3020 INVALID TARIFF FILE TYPE
NONFATAL
NONE

FOR LOGIC FLOW, SEE FIGURE 3-40.
3.9.5 CRATTB Module

SUBROUTINE CRATTB(ERROR,FILNAM)

PURPOSE
READ INPUTS TO CREATE A TARIFF BASED ON THE AT&T PHILOSOPHY

INPUT ARGUMENTS
ERROR LOGICAL INDICATES WHETHER A FATAL ERROR HAS BEEN FOUND: .TRUE. - FATAL
FILNAM REAL*8 NAME OF TARIFF FILE TO BE CREATED

OUTPUT ARGUMENTS
ERROR LOGICAL INDICATES WHETHER A FATAL ERROR HAS BEEN FOUND: .TRUE. - FATAL ERROR

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
ADJ /INPATT/ S
CODE /INPNOD/ S
INCSC /INPATT/ S
NDNS /INPNOD/ S
NTLEV/ /INPATT/ U
DEBUG /LUN/ U
LUNOUT /LUN/ U

LOCAL VARIABLES
ITEMP,JTEMP I*4 BUFFERS FOR USER INPUTS
MINUS I*4 CONSTANT '/-' INDICATES DELETION
NPLUS I*4 CONSTANT '/+' INDICATES INSERTION
INLIST LOGICAL CITY CODE ENTERED IN LISTED CITY
INCLUD LOGICAL USER WANTS TO ADD LISTED CITY
EXCLUD LOGICAL USER WANTS TO DELETE LISTED CITY
I,J I*4 LOOP INDEXES
NULL I*4 CONSTANT '/ ' INDICATES BLANK CARD

SUBPROGRAMS CALLED
ATTLST

ERRORS
FATAL
B3100 INVALID LISTED CITY ADDITION/DELETION
B3110 LISTED CITY ADDITION/DELETION DUPLICATION
NONFATAL
B3500 CITY TO BE ADDED ALREADY A LISTED CITY
B3510 CITY TO BE DELETED NOT A LISTED CITY

FOR LOGIC FLOW SEE FIGURE 3-45.

3-251
3.9.6 CRWUB Module

SUBROUTINE CRWUB(ERROR,FILNAM)

PURPOSE
READ INPUTS TO CREATE A TARIFF BASED ON THE WU PHILOSOPHY

INPUT ARGUMENTS
ERROR LOGICAL. INDICATES WHETHER A FATAL ERROR IS FOUND: .TRUE. - FATAL ERROR
FILNAM REAL*8 NAME OF TARIFF FILE TO BE CREATED

OUTPUT ARGUMENTS
ERROR LOGICAL INDICATES WHETHER A FATAL ERROR IS FOUND: .TRUE. - FATAL ERROR

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
CAT /INPWU/ S
CHGLH /INPWU/ S
CHGMH /INPWU/ S
CHGSH /INPWU/ S
CODEL /INPNOD/ S
CODER /INPNOD/ S
NADJ /INPWU/ S
DEBUG /LUN/ U
LUNOUT /LUN/ U

LOCAL VARIABLES
ITEMP I*4 BUFFERS
JTEMP I*4 FOR USER
KTEMP I*4 INPUTS
NULL I*4 CONSTANT '/INDICATES BLANK CARD
ADD LOGICAL PAIR SPECIFIED TO BE ACCESS CITIES
DELETE LOGICAL PAIR SPECIFIED TO BE DELETED AS ACCESS CITIES
PAIR I*4 INDICATES WHETHER NODES SPECIFIED ARE AN ACCESS PAIR IN PRESTORED TARIFF
.I TRUE. - NODES ARE ACCESS PAIR
.FALSE. - NODES ARE NOT IN ACCESS PAIR
I I*4 LOOP INDEX

SUBPROGRAMS CALLED
WUPAIR

ERRORS
FATAL
NONE
NONFATAL
B3600 ACCESS PAIR CATEGORY CHANGE ERRONEOUS
B3610 ACCESS PAIR TO BE DELETED NOT IN LIST

3-252
B3620 CATEGORY OUT OF RANGE
B3630 ACCESS PAIR DUPLICATION

FOR LOGIC FLOW SEE FIGURE 3-46.
3.9.7 CRSBSB Module

SUBROUTINE CRSBSB(ERROR,FILNAM)

PURPOSE
READ INPUTS TO CREATE A TARIFF BASED ON THE SBS PHILOSOPHY

INPUT ARGUMENTS
ERROR LOGICAL INDICATES IF THERE IS A FATAL ERROR
.FALSE. - NO FATAL ERROR
FILNAM REAL*8 NAME OF TARIFF FILE TO BE CREATED

OUTPUT ARGUMENTS
ERROR LOGICAL INDICATES IF THERE IS A FATAL ERROR
.TRUE. - FATAL ERROR
.FALSE. - NO FATAL ERROR

DATA STRUCTURES
(U - USED, S - DEFINED, R - READ)
NAC /INPSBS/ S
SCU /INPSBS/ S
FTU /INPSBS/ S
CAUMIN /INPSBS/ S
INCSB /INPSBS/ S
NSBTAB /INPSBS/ U
NACNMB /SBSMXD/ S
NACLOC /SBSMXD/ S
DEBUG /LUN/ U
LUNOUT /LUN/ U

LOCAL VARIABLES
I I*4 LOOP INDEX
LOC I*4 NAC LOCATION AS SPECIFIED BY USER
NULL I*4 CONSTANT /' / RESPONSE OF <CR>

SUBPROGRAMS CALLED
NONE

ERRORS
FATAL
NONE
NONFATAL
NONE

FOR LOGIC FLOW, SEE FIGURE 3-47.
3.9.8 **EXCONB Module**

**SUBROUTINE EXCONB**

**PURPOSE**
READ INPUTS AND BUILD THE EXECUTION CONTROL FILE

**INPUT ARGUMENTS**
NONE

**OUTPUT ARGUMENTS**
NONE

**DATA STRUCTURES**
(U - USED, S - DEFINED, R - READ)

- **RUNID** /INPEXC/ S
- **MAXRST** /INPEXC/ U
- **NRSTRT** /INPEXC/ S
- **OPTN** /INPEXC/ S
- **NETFIL** /INPEXC/ S
- **NTRF** /INPEXC/ S
- **FILNAM** /INPEXC/ S
- **MAXTRF** /INPEXC/ U
- **NOUT** /INPEXC/ U
- **OUTCNT** /INPEXC/ S
- **TFILES** /TARFIL/ U
- **TARDEF** /TARFIL/ U
- **NTFIL** /TARFIL/ U
- **LEVEL** /INPEXC/ S
- **DEBUG** /LUN/ U
- **LUNOUT** /LUN/ U

**LOCAL VARIABLES**

- **NULL** R*8
- **TEMP** R*4
- **N** I*4
- **I** I*4
- **M** I*4
- **K** I*4
- **L** I*4
- **NTAB** I*4
- **INDEX** I*4
- **IFIL** I*4
- **DEFIND** I*4
- **START** I*4

**CONSTANTS**

- **/I** INDICATES BLANK CARD
- **REAL BUFFER**
- **POINTER TO NEXT SLOT AVAILABLE IN ARRAY CONTAINING TARIFF NAMES**
- **INDEX TO LOOP THAT ACCUMULATES DATA FOR NRSTRT PROBLEMS**
- **NUMBER OF TARIFFS SPECIFIED FOR CURRENT PROBLEM**
- **NUMBER OF OUTPUT TABLES TO SUPPRESS**
- **NUMBER OF OUTPUT TABLE TO BE SUPPRESSED**
- **POINTER TO TARIFF NAME FOUND IN DIRECTORY - 999 IF NOT FOUND**
- **CONSTANT - NETWORK DIRECTORY**
- **CONSTANT - FILE IS DEFINED**
- **BEG OF LIST OF TARIFFS SPECIFIED FOR CURRENT PROBLEM WITHIN FILNAM**

3-255
END I*4  END OF LIST OF TARIFFS SPECIFIED FOR 
CURRENT PROBLEM
IOPT I*4  OPTION FOR CURRENT PROBLEM
ITAR I*4  TARIFF TYPE
PRSTAR R*8  CONSTANT TARIFF NAMES OF PRESTORED 
TARIFFS
ERROR LOGICAL  INDICATES A FATAL ERROR HAS BEEN FOUND 
FDUP LOGICAL  TARIFF FILE DUPLICATED WITHIN A RUN 
J I*4  LOOP INDEX

SUBPROGRAMS CALLED
FILDEF
DIRGEN
EXCREA

ERRORS
FATAL
B4000  NEGATIVE NUMBER OF RESTARTS SPECIFIED
B4010  INVALID OPTION
B4020  UNDEFINED NETWORK FILE
B4030  UNDEFINED TARIFF FILE (MIXED RUN)
B4040  TARIFF NOT ATT-TYPE
B4050  INVALID FILE TYPE
B4060  NON-POSITIVE # OF TARIFF FILES SPECIFIED
B4070  NO VALID TARIFF FILES SPECIFIED

NONFATAL
B4500  MAXIMUM RESTARTS EXCEEDED - WILL RUN MAX 
B4510  MAXIMUM TARIFFS EXCEEDED - WILL RUN MAX 
B4520  TARIFF FILE DUPLICATION
B4530  UNDEFINED TARIFF FILE (NON-MIXED RUN)
B4540  INVALID FILE TYPE (NON-MIXED RUN)
B4550  INVALID OUTPUT TABLE NUMBER

FOR LOGIC FLOW SEE FIGURE 3-20.
4.0 DATA STRUCTURES

The data structures required by the CNDC model are read from the data base files or built during the execution of the model. These data structures are contained in arrays and variables and maintained in common blocks. Following is a dictionary of these structures. Following each structure name and size, is the name of the common block where it is maintained, its data type, and definition.

* ADJ (600) /INPATT/ I*4
indicates if associated code is an ATT category A addition or deletion
  -1 category B
  +1 category A

* ALTSD (200) /ATT/ I*4
array of alterations to ATT rate center listed/unlisted classification for each tariff in the current run. This array is used only for runs involving terrestrial tariffs. The values in this array represent the number of the rate center of concern. If the value is negative, then for this tariff, the rate center is unlisted. If the value is positive, then for this tariff the rate center is listed. The alterations for each tariff are grouped together in this array. Within each tariff grouping, the rate center numbers are arranged in ascending order by absolute value of the rate center number. One extracts the desired data from this array by using two other arrays - TPTR (NTARIF) and NINLST (NTARIF).

* CAT (600) /INPWU/ I*4
tariff category of satellite access city pair
  0 not satellite access city pair
  1 long haul
  2 medium haul
  3 short haul

* CAUMIN /INPSBS/ R*4
minimum CAU charge per NAC

* CHGLH /INPWU/ R*4
long haul channel charge

* CHGMH /INPWU/ R*4
medium haul channel charge
* CHGSH /INPWU/ R*4
  short haul channel charge
* CODE (600) /INPNOD/ I*4
  unique four character code of each node
* CODEL (600) /INPNOD/ I*4
  unique four character code of originator of traffic
* CODER (600) /INPNOD/ I*4
  unique four character code of receiver of traffic
* CODER (600) /INPNOD/ I*4
  unique four character code of receiver of traffic
* COST (600,600) /CSTOUT/ R*4
  for each link in network, the cost per circuit that is the minimum over all tariffs that service that link
* CPNA /SBS/ I*4
  maximum number of CAUs per SCU
* CROPT /INPNET/ I*4
  network creation option
  1  all prestored nodes
  2  all prestored nodes except those specified
  3  only those prestored nodes specified
  4  all prestored nodes which make up specified percent of traffic
* DEBUG /LUN/ I*4
  logical unit number of file DEBUG.OUT which contains debug print
* DRCATA (20) /INPATT/ I*4
  directory of the prestored category A rate center array PRCATA
* DRNODE (25) /RNODES/ I*4
  directory to the RTCNTR array of network nodes. Use of this directory will shorten the search time required to find an element in RTCNTR.
* EXCUTE /LUN/
  logical unit number of the execution control file

* FILNAM (130) /INPEXC/
  tariff files to be used for all problem sets in an execution

* FILNET /RUN/
  network file used for current optimization problem

* FLNM (20) /CURFIL/
  names of user defined network files

* FTU /INPSBS/
  charge per full time transmission unit

* H (600) /RNODES/
  horizontal coordinates of rate centers in current network

* HC (600) /INPNET/
  horizontal coordinate of each node specified

* HEXEC (2) /HEADRS/
  header of execution control file

* HNATT /HEADRS/
  header of user defined ATT tariff file

* HNET /HEADRS/
  header of network directory file

* HNOD (2) /HEADRS/
  header of file of prestored nodes

* HNSBS /HEADRS/
  header of user defined SBS tariff

* HNWU /HEADRS/
  header of user defined Western Union tariff
* HPATT (2) /HEADRS/ R*8
  header of prestored ATT tariff
* HPSBS (2) /HEADRS/ R*8
  header of prestored SBS tariff
* HPWU (2) /HEADRS/ R*8
  header of prestored Western Union tariff
* HTAR (2) /HEADRS/ R*8
  header of tariff directory
* HTFC (2) /HEADRS/ R*8
  header of traffic table
* HVOL (2) /HEADRS/ R*8
  header of prestored file of nodes arranged by traffic volume
* INCSB (3) /INPSBS/ R*4
  input SBS CAU increment charges
* INCSC (9,3) /INPATT/ R*4
  increment charge at each mileage breakpoint for each terrestrial schedule
* KDIR /PRSNDS/ I*4
  number of entries in directory KRNODE to PRNDS array
* KPBLK /PRSNDS/ I*4
  number of PRNDS items per block of items referenced by directory KRNODE
* KRNODE (20) /PRSNDS/ I*4
  directory to PRNDS array of prestored nodes
* LSTACC (65) /PRSWU/ I*4
  list of satellite access cities in prestored Western Union tariff
* LSTED (600) /ATT/ I*4
  if the rate center referenced is an ATT category A rate center, LSTED contains 1, otherwise it contains a 0
* MAXUSR /INPNET/ I*4
maximum number of user specified nodes

* LUNNET /LUN/ I*4
start logical unit number for prestored network files

* LUNOUT /LUN/ I*4
logical unit number of file to contain optimization output

* LUNSPL /LUN/ I*4
start logical unit number of files to contain traffic table using of network file being defined

* LUNTRF /LUN/ I*4
start logical unit number of prestored tariff files

* LVLTRF /INPNET/ I*4
level of traffic for prestored nodes as number of voice circuits

* MAXACC /PRSWU/ I*4
maximum slots in array LSTACC which contains prestored WU access cities

* MAXGRP /SBS/ I*4
no longer used in SBS

* MAXNDS /INPNOD/ I*4
maximum number of nodes in network

* MAXRST /INPEXC/ I*4
maximum number of problem sets in execution control file

* MAXSAC /WU/ I*4
maximum slots for satellite access cities for any optimization run using Western Union tariffs

* MAXTRF /INPEXC/ I*4
maximum number of user defined tariffs

* MDIR /INPATT/ I*4
number of entries in the directory to the prestored category A rate centers
for all ordered pairs of nodes, the cost per circuit to connect the nodes using the least case route.

- **MINCPC (600,500)** /CS10U/  
  number of prestored category A rate centers per block of items referenced by directory

- **MPBLK** /INPATT/  
  maximum number of prestored satellite access cities in Western Union tariff

- **MXACCC** /ARRYS/  
  maximum number of category A rate centers in prestored ATT tariff

- **MXCTA** /ARRYS/  
  maximum number of prestored nodes

- **MXPRN** /ARRYS/  
  maximum number of prestored nodes

- **NAC** /INPSBS/  
  SBS charge per network access center

- **NACC** /PRSWU/  
  number of prestored network access centers in Western Union tariff

- **NACLOC (600)** /SBSMXD/  
  NACLOC (1) to NACLOC (N) gives the number of the N nodes in the current network which will be NAC location in an SBS mixed problem

- **NACNMB** /SBSMXD/  
  number of NAC locations in the current network in an SBS mixed problem

- **NADJ** /INPWU/  
  number of satellite access only pair additions and deletions specified in user defined Western Union tariff

- **NALT** /ATT/  
  maximum number of alterations to ATT category A status
* NCASE /RUN/  I*4
  current optimization problem being run from the set of problems specified
  in the execution control file
* NCATA /INPATT/  I*4
  number of category A rate centers
* NDEF (20) /NETFIL/  I*4
  status of the files in the user defined network directory
* NDIR /RNODES/  I*4
  number of entries in the directory, DRNODE, to the RTCNTR array
* NETDIR /LUN/  I*4
  logical unit number of directory to user defined networks
* NETFIL (10) /INPEXC/  R*8
  network file to be used for each problem set of execution control file
* NETOL (600,600) /CSTOUT/  I*4
  for each link in the least cost network, the total network traffic
  traversing the link
* NFIL /CURFIL/  I*4
  number of files listed in directory of interest
* NFILES (20) /NETFIL/  R*8
  names of the files in the user defined network directory
* NINLST (13) /ATT/  I*4
  array of number of alterations to ATT listed/unlisted status made by each
tariff. Below is an example of how the three arrays ALSTD, TPTR, and
NINLST work together:

<table>
<thead>
<tr>
<th>TARIFF NUMBER</th>
<th>NINLST</th>
<th>TPTR</th>
<th>ALSTD</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>+1</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>1</td>
<td>-4</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>5</td>
<td>+33</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>-77</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>+17</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>-62</td>
</tr>
<tr>
<td>NTARIF</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
In this example, terrestrial tariff 2 makes changes to the All listed/unlisted status of 4 rate centers. These changes are listed in the ALSTD starting at position 1. For tariff 2, rate center 1 will now be listed, rate center 4 unlisted, etc. Tariff 1 makes no changes.

* NLEVEL   /ATT/   I*4
  maximum number of mileage breakpoints in model arrays involving ATT tariff rates

* NNDS     /INPNOD/  I*4
  number of nodes specified in network being created. Also, in creation of ATT tariff, the number of adjustments to category A status of a rate center

* NNFIL    /NETFIL/  I*4
  number of files in network directory

* NNODES   /RUN/   I*4
  number of nodes in current network

* NODPTR (600) /SBSMXD/  I*4
  for each node in the network of an SBS mixed problem, the node which contains the SBS facilities that this node must use. If NODPTR (I) is negative than node I is an SBS facility.

* NOUT     /INPEXC/  I*4
  number of output tables generated within model

* NPBLK    /RNODES/  I*4
  number of RTCNTR items per block of items referenced by the directory DRNODE

* NPLEVL   /ATT/   I*4
  number of mileage breakpoint in ATT tariff being used

* NPRNDS   /PRSNDS/  I*4
  number of prestored nodes

* NPTAB    /SBS/   I*4
  number of CAU rate table entries in SBS tariff
* NRSTRT /INPEXC/ I*4
number of problems defined in current execution control file

* NSBTAB /INPSBS/ I*4
number of CAU per NAC breakpoints entries in tables describing the monthly charge per CAU

* NSCHED /ATT/ I*4
number of terrestrial schedules

* NSCUPN /SBS/ I*4
maximum number of SCUs per NAC

* NTARIF /RUN/ I*4
number of tariffs being considered in current problem

* NTFIL /TARFIL/ I*4
number of files in prestored tariff directory

* NSATAx /WU/ I*4
number of Western Union satellite access cities this problem

* NTLEVL /INPATT/ I*4
number of mileage breakpoints in terrestrial rate table

* NTRF (10) /INPEXC/ I*4
number of tariffs specified for each problem set in execution control file

* NTRFC /INPNET/ I*4
number of specifications of traffic levels between a pair of nodes

* NWUCAT /WU/ I*4
number of Western Union rate categories
number of values in table of satellite access city pair rates
NSATAX*(NSATAX -1)/2

Since for Western Union, the rate from A to B is the same as the rate from B to A, it is not necessary to put this information in a NSATAX*NSATAX array. All that is required is the triangle of values above the diagonal from the upper left to the lower right corner. The array WUPRS contains this upper triangle of data for each Western Union tariff in a problem. The integer function PTRPRS gets the pointer to the location in WUPRS for a specified pair.

* OPTION /RUN/ I*4

type of run
1 terrestrial only
2 WU only
3 SBS only
4 Mixed WU
5 Mixed SBS

* OPTN (10) /INPEXC/ I*4

type of problem for each problem set within the execution control file
1 terrestrial only
2 WU only
3 SBS only
4 mixed WU
5 mixed SBS

* ORROUTE (600,600) /CSTOUT/ I*2

for each ordered pair of nodes in network the optimum successor node along the least cost path between the nodes

* OUTCNT (6,10) /INPEXC/ I*4

output control flags indicating which output tables are desired for each problem set in the execution control file

* OUTFLG (6) /RUN/ I*4

output desired for current problem

* PERCNT /INPNET/ I*4

when creating a network using the percent option, the percent desired
sorted list of the codes of the category A rate centers as defined in the ATT terrestrial tariff

array of nodes in prestored network

number of CAUs in SBS solution

number of FTUs in SBS solution

number NACs in SBS solution

number of SCUs in SBS solution

sorted list of the four character codes of the rate centers in the current network

color character run identification of execution control file

if the rate center I is a satellite access city for any tariff in a problem, SA(I) is the pointer to the data about this satellite access city. If I is not an access city, SA(I) = 0.

SATACC (I,J) = 1 if satellite access city I is an access city for tariff J

SATCST (I) contains the cost per voice circuit for SBS satellite service from NAC location I
SATNVC (600) /SBSMX2/ I*4

SATNVC (I) is the number of voice circuits using satellite from NAC location I.

SATOTL (600, 600) /SATMXD/ I*4

for each SBS satellite link in the least cost network, the total satellite traffic traversing the link

SBMX /SBSMXD/ I*4

NAC placement option for this SBS mixed problem. If SBXM = 1, the user wishes to place NACs at the nodes specified in the user defined SBS tariff. Otherwise the model will determine the optimal placement of NACs.

SCU /INPSBS/ R*4

charge per supplemental capacity unit in SBS tariff

STABLE (600) /SBSMX2/ I*4

STABLE (I) specifies how many nodes actually contain SBS facilities for the proposed configuration of I NAC locations

STATUS (20) /CURFIL/ I*4

status of each listed in the network directory

STLCKT /CSTMXD/ I*4

number of satellite circuits in optimal SBS mixed network

STLCST /CSTMXD/ R*4

total satellite cost in optimal SBS mixed network

STLMLG /CSTMXD/ I*4

number of miles of satellite circuits in optimal SBS mixed network
* SVC (600,600) /CSTOUT/ I*2

for all links in the network, the tariff that yields the minimum cost per circuit

* TARDEF (13) /TARFIL/ I*4

type of tariff defined in each of the user defined tariffs
0 available for definition
1 ATT tariff
2 WU tariff
3 SBS tariff

* TATBAS (9,3,13) /ATT/ R*4

base charge for each mileage breakpoint in terrestrial tariff for each of three schedules and each of the tariffs to be used in a problem

* TATINC (9,3,13) /ATT/ R*4

incremental charge for each mileage breakpoint in terrestrial tariff for each of three schedules and each of terrestrial tariffs to be used in current problem

* TATMIL (9) /INPATT/ I*4

mileage breakpoints in the terrestrial rate table. TATMIL (9) = MAXINT to take care of all mileages above the maximum mileage listed in the table TATMIL (8). The increment charge corresponding to TATMIL (9) is therefore the charge per mile for any distance greater than TATMIL (8). In the same way, as an example, the increment associated with TATMIL (4) is the rate per mile for any distance in excess of TATMIL (3) and less than or equal to TATMIL (4) used when creating an ATT tariff.

* TATMLG (9) /ATT/ I*4

mileage breakpoints in terrestrial tariffs being used in current runs. (See TATMIL for further description.)

* TF (13) /RUN/ R*8

names of tariffs being compared in current problem

* TFILES (13) /TARFIL/ R*8

names of user defined tariff in tariff directory

* TOTNDS /INPNET/ I*4

total nodes in network being defined
TOTSAT (1) contains the total system cost for a proposed configuration of NAC locations.

TPTR (13) array of pointers to position within ALSTD array where this tariff's alterations to listed/unlisted designations begin. A value of 0 indicates that there are no alterations to ATT listed/unlisted status for this tariff.

TRAFIC (600,600) traffic level in voice circuits from each node to every other node in the network of the current problem.

TRFATT logical unit number of prestored ATT terrestrial tariff.

TRFDIR logical unit number of directory to NASA defined tariffs.

TRFLVL total traffic level of prestored nodes as number of voice circuits.

TRFNDS logical unit number of file of prestored traffic nodes.

TRFSBS logical unit number of prestored SBS tariff.

TRFTBL logical unit number of prestored traffic table.

TRFVOL logical unit number of prestored file of nodes arranged from busiest to least busy.

TRFWU logical unit number of prestored Western Union tariff.
* TRRCKT /CSTMXD/ 1*4
  in optimal SBS mixed network, the number of terrestrial circuits

* TRRCST /CSTMXD/ R*4
  in optimal SBS mixed network, the total terrestrial cost

* TRRMLG /CSTMXD/ I*4
  in optimal SBS mixed network, the number of miles of terrestrial circuits

* TSBBKP (3) /SBS/ I*4
  CAU/NAC breakpoints in table of CAU rates for SBS tariffs in current problem

* TSBCAB (3,13) /SBS/ R*4
  base cost for each CAU/NAC breakpoint for each tariff in current SBS problem

* TSBCAI (3,13) /SBS/ R*4
  incremental cost for each CAU/NAC breakpoint for each tariff in current SBS problem

* TSBCAM (3) /INPSBS/ I*4
  CAU per NAC breakpoints used in description of monthly charge for CAUs. TSBCAM (3) - MAXINT. The logic for operating in this table is the same as for TATMIL described above.

* TSBCMN (13) /SBS/ R*4
  minimum CAU charge per NAC for each SBS tariff in current problem

* TSBFTU (13) /SBS/ R*4
  FTU charge for each SBS tariff in current problem

* TSBNAC (13) /SBS/ R*4
  NAC charge for each SBS tariff in current problem

* TSBSCU (13) /SBS/ R*4
  SCU rate for each SBS tariff in current problem
* TTYP (13) /RUN/ I*4

tariff type of each tariff in current problem

1 ATT
2 WU
3 SBS

* TWU (3,13) /WU/ R*4

Western Union channel charge for each tariff is current problem at each category

* UNET /HEADRS/ R*8

header of user defined network file

* UTAR /HEADRS/ R*8

header of user defined tariff file

* V (600) /RNODES/ I*4

vertical coordinates of rate centers in current problem

* VC (600) /INPNET/ I*4

vertical coordinates of each node specified when building a network

* VCIRTS (600) /INPNET/ I*4

number of addition traffic from CODEL to CODER when building a new network

* VCPFTU /SBS/ I*4

number of voice circuits per FTU for SBS tariffs

* WUPRS (2080, 13) /WU/ I*2

array of connections and rate categories for each Western Union tariff in current problem

0 no connection between this city pair
1 long haul
2 medium haul
3 short haul

see NWUTRI for further description

* WURTCT (65) /WU/ I*2

pointer to node in RTCNTR array of each satellite access city in current problem
5.0 CNDC PROGRAM DATA BASE

5.1 INTRODUCTION

Within the CNDC Model the Data Base Management System refers to the collection of files, prestored and user defined, on which is stored network and tariff descriptions; the program required to build and maintain the prestored data files; and the subprograms within the CNDC Model which retrieve data from and put data into the data base files. This section describes the files which make up the data base.

The modules within the CNDC program belong to one or two systems. One set of modules performs the data base management function and the other performs the optimization function. Figure 5-1 illustrates how these two systems interact with the data base. Those modules which perform the data base management function create and maintain the data base files and retrieve from these files that data which is required to perform the optimization function.

The files which make up the data base are of two kinds: the prestored files and the user defined files. The prestored files are created and maintained by a separate program, PRSTRD. Currently, Sonalysts has the responsibility to maintain the integrity of these files. The model does not provide the user with direct access to any of these prestored files. The prestored data base is described in section 5.2.

The files within the user defined data base can be referenced, created, deleted, listed by the user in any INPUT session (see sections 3.3 - 3.5). These files contain the descriptions of user defined networks, user defined tariffs, and an execution file. The user defined data base is described in section 5.3:
Figure 5-1. Communications Network Design and Costing Model
5.2 PRESTORED DATA BASE

The files which contain the prestored data are created and maintained by Sonalysts personnel only. These files define the set of prestored nodes (SMSA - Standard Metropolitan Statistical Areas), the traffic between every pair of nodes in the prestored set, the ATT, Western Union, and SBS tariffs, and the directories to the user defined network and tariff files. The eight files which form the prestored data base are described in the following sections.

5.2.1 Network

The prestored traffic table is to contain up to 350 unique traffic nodes, each node designated by a unique four letter alphabetic code name. The list of traffic nodes provided to Sonalysts contained 316 nodes. The file TRFNDF, known internally to CNDC as TRFNDS, contains the four letter code, vertical and horizontal coordinates, and SMSA descriptor of each of these 316 nodes. The 316 node description records are arranged alphabetically by the four character code. These 316 records are preceded by two additional records. The first identifies the file as the file of prestored traffic nodes. The second indicates the number of nodes defined. The file format follows. The data type of each field is in parentheses following the field name.

```
TRAFFIC NODES $\bullet\bullet\bullet$ (A16)
NREC (I3)
NREC OCCURRENCES
CODE (A4) VCOORD (I5) HCOORD (I5) SMSA (A36)
```

where

- **NREC**: number of nodes in the file of prestored traffic nodes
- **CODE**: unique four character code of the node
- **VCOORD**: vertical coordinate of the node
- **HCOORD**: horizontal coordinate of the node
This file has been created and will, if required, be updated by Sonalysts personnel. It will not be available to the user through the INPUT function for edit or deletion.

5.2.2 Traffic Table

The directional traffic between any pair of nodes in the prestored set is represented as a percentage of the total traffic between all prestored traffic nodes. This directional traffic is described by an ordered pair of nodes, the first node identifying the origin of the traffic and the second node the termination. The file TOUT, known internally to CNDC as TRFTBL, contains the traffic values. Although it may contain up to 350 X 350 traffic values, it currently contains the 316 X 316 array of traffic values for the 316 nodes identified in the traffic nodes file. The file contains 316 records with 350 values in each record. Each value is written in I10 format. To get the percent represented by each value, it is necessary to multiply each value by 10⁻¹⁰. The 316 records of traffic values is preceded by two additional records. The first identifies this file as the file of prestored traffic levels. The second indicates the number of traffic nodes and the power to which ten must be raised (currently -10) to yield the factor by which each value must be raised to get percent traffic level. The file format follows:

```
TRAFFIC TABLE (A16)
NNODES (I3), POWER (I3)

NNODES OCCURRENCES

{ TLEVEL1 (I10), ..., TLEVEL350 (I10) }

...
...
```

where

- **NNODES**: number of nodes in prestored network
- **POWER**: power to which 10 must be raised to yield the factor by which each value in the table must be multiplied to yield a normalized matrix
- **TLEVEL**: traffic level
5.2.3 Traffic by Volume

It was desirable to be able to specify a network by all those prestored nodes which constituted a given percentage of the total traffic. To make this possible, a file was created which lists the prestored nodes from busiest to least busy. With each node specification is the cumulative percentage which represents that portion of the total traffic contained in a network with all nodes up to a given node. The format of the file VLST, known internally to CNDC as TRFVOL, follows:

```
| TRAFFIC VOLUMES (A16) |
| NNODES (I3) |
| NNODES |
| CODE (A4) POS (I4) PCNT (F11.7) |

where

NNODES number of nodes in prestored network
CODE unique for character code of each prestored node
POS position of node in prestored network file
PCNT cumulative percent of traffic represented by all nodes up to and including specified node
```

5.2.4 ATT Tariff

The prestored ATT tariff contains a list of the category A rate centers and the rates for series 2000/series 3000 channels between any two cities for the three schedules: schedule 1: both cities are category A rate centers; schedule 2: only one of the pair of cities is a category A rate center; and schedule 3: neither of the pair of cities is a category A rate center. Each of the category A rate centers is identified by its unique four character code. Each of the three tariff schedules is defined by a table of rates which are a function of mileage. The table is organized by mileage breakpoints. Associated with each breakpoint are a base charge plus an increment charge for every mile in excess of the mileage breakpoint. The file TRFATT contains the data required to define the ATT tariff structure. The first
record, the header record, indicates that this file describes the prestored ATT tariff. The second record indicates the number of records required to list the category A rate centers and rate schedule. The file format follows:

```
ATT TARIFF (A16)
NREC (I3) NCATA (I3)

 NREC
   OCCURRENCES
       CODE1 (A5) .... CODE16 (A5)

 .........

 NMLG (I3)

 `NMLG
   OCCURRENCES
       MLG (I10) BASE1 (F8.2) INC1 (F6.2) BASE2 (F8.2)
              INC2 (F6.2) BASE3 (F8.2) INC3 (F6.2)

 .......
```

where

- **NREC**: number of records following which contain the list of ATT category A rate centers
- **NCATA**: number of category A rate centers in list
- **CODE**: unique four character code for each category A rate center in the file
- **NMLG**: number of mileage breakpoints in ATT rate table
- **BASE1**: base charge for each mileage breakpoint for schedule 1
- **BASE2**: base charge for each mileage breakpoint for schedule 2
- **BASE3**: base charge for each mileage breakpoint for schedule 3
- **INC1**: increment charge for each mileage breakpoint for schedule 1
- **INC2**: increment charge for each mileage breakpoint for schedule 2
- **INC3**: increment charge for each mileage breakpoint for schedule 3

This file will, if required, be modified by Sonalysts personnel only. It is not available to the user through the INPUT function for edit or deletion.
5.2.5 WU Tariff

The prestored Western Union Tariff contains a list of the satellite access city pairs along with the tariff category (1: long haul, 2: medium haul, 3: short haul) associated with each pair. This list will be followed by the channel charge for each of the three categories. The satellite access city pairs are specified as a pair of four character codes arranged alphabetically by the left most member and then within the left member alphabetically by the right most member. The file `TRFWU` contains the above data in the following format:

```
WU TARIFF     (A16)
NPAIRS (I3)
NPAIRS OCCURRENCES
  CODEL (A5) CODER (A5) CAT (I2)
  ......
  CHGLH (F8.2)
  CHGMH (F8.2)
  CHGSH (F8.2)
```

where

- **NPAIRS** number of satellite access city pairs
- **CODEL** unique four character code of member of satellite access city pair
- **CODER** unique four character code of member of satellite access city pair
- **CAT** tariff category of access city pair
  - 1 long haul
  - 2 medium haul
  - 3 short haul
- **CHGLH** long haul channel charge
- **CHGMH** medium haul channel charge
- **CHGSH** short haul channel charge

This file will be modified, if required, by Sonalysts personnel only. It will not be available to the user through the INPUT function for edit or deletion.
5.2.6 SBS Tariff

The prestored SBS tariff contains the SBS monthly rates for network access centers (NAC), supplemental capacity units (SCU), CNS-A full time transmission units (FTU), channel access units (CAU), and minimum CAU charge per NAC. The CAU charges are a function of the number of CAUs at each NAC. Associated with each CAU count breakpoint is a base charge plus an incremental charge per CAU.

The file TRFSBS contains the above data in the following format:

```
SBS TARIFF (A16)
NAC (F8.0)
SCU (F8.0)
FTU (F8.0)
CAUMIN (F8.0)
NTAB (I3)
OCURRENCES { NCAU (I10) BASE (F8.0) INC (F4.0).
...
```

where

- **NAC** charge per network access center
- **SCU** charge per supplemental capacity unit
- **FTU** charge per full time transmission unit
- **CAUMIN** minimum CAU charge per NAC
- **NTAB** number of values in CAU rate table
- **NCAU** number of CAU/NAC up to which the accompanying base and increment charges apply
- **BASE** base charge for specified number of CAUs
- **INC** incremental charge for each CAU over a specified number

This file will be modified, if required by Sonalysts personnel only. It is not available to the user through the INPUT function for edit or deletion.
5.2.7 User Defined Tariff Directory

The CNDC Model, through the INPUT module, allows for the definition of NASA tariffs based on the prestored tariffs, terrestrial and satellite. In addition, the user may store up to 13 of the NASA defined tariffs so they may be used as previously stored data in future computer runs. These 13 NASA defined tariffs will be stored in files named TAR001-TAR013.

The INPUT module must be able to keep track of what NASA tariffs have been defined and what files they are located in. To this end, the file TRFDIR contains a directory of the current set of NASA defined tariffs. The file format follows:

```
| TARIFF DIRECTORY   (A16) |
| NFIL (I3)            |
| NFIL                 |
| OCCURRENCES          |
| FILENAME (A6)        |
| TYPE (I3)            |
```

where

- NFIL number of NASA defined tariff files in list below
- FILENAME name of file containing or available for a NASA defined tariff
- TYPE tariff type
  - 0 file available for definition
  - 1 ATT type tariff
  - 2 WU type tariff
  - 3 SBS type tariff

Currently, NFIL has the value of 13 and FILENAME can have the values TAR001 to TAR013. TYPE indicates if the associated file is currently in use or available for use. If this file is currently in use, TYPE also indicates the type of tariff it defines.
5.2.8 User Defined Network Directory

The CNDC Model, through the INPUT module, allows for the definition of networks. In addition, the user may store up to 20 of these network definitions so they may be used in future computer runs. These 20 network definitions are stored in files named NETO01-NETO20.

The INPUT module must be able to keep track of what networks have been defined and files they are located in. To this end, the file NETDIR contains a directory of the current set of network definitions. The file format follows:

\[
\begin{align*}
\text{NETWORK DIRECTORY (A16)} \\
\text{NFIL (I3)} \\
\{ \\
\text{FILENAME (A6)} \\
\text{STATUS (I3)} \\
\text{NFIL OCCURRENCES} \\
\ldots
\end{align*}
\]

where

- \text{NFIL} \quad \text{number of network definition file names in list below}
- \text{FILENAME} \quad \text{name of file which contains or may contain a network definition}
- \text{STATUS} \quad \text{status of network}
  
  \begin{align*}
  &0 \quad \text{available for definition} \\
  &1 \quad \text{currently defined}
  \end{align*}

Currently, NFIL has the value of 20 and FILENAME may have values NETO01 to NETO20.
5.3 USER DEFINED DATA BASE

The user has the capability, using the INPUT module, to define tariffs and network. These user defined network and tariff files constitute the user defined data base. These files are built by the user through an interactive process with the model. The user must also specify his program execution runstream. This execution control file, also built interactively through INPUT, is a final file maintained a part of the user defined data base.

5.3.1 User Defined Networks

The user supplied input to CNDC includes the NASA defined tariffs and a network configuration. The user must specify which, if any, of the prestored traffic nodes he wants to include in the run. In addition, he must define the total traffic for the stored nodes, expressed as voice circuits. The user also may include in his network, traffic not specified in the prestored traffic table. He does this by defining additional nodes to be included in his network and indicating traffic between these additional nodes and other nodes in the network. In addition the user may specify additional traffic for prestored nodal pairs. The traffic he specifies must be expressed as number of voice circuits. Each additional node must be specified by its unique four character code and must be accompanied by its vertical and horizontal coordinates. The user will define this network through the INPUT module and create a file named NETOn where n can assume a value between 01 and 20. The format of the network definition files follows:

```
NETWORK NETn (A16)
  NNODES (13)
    CODE (A4) VCOORD (I5) HCOORD (I5)
  OCCURRENCES
    .
    .
    .
  TRFLVL (III)
  NTRFC (13)
    CODEL (A4) CODER (A5) VCIRTS (I11)
  OCCURRENCES
    .
    .
    .
```
where

**NNODES**  number of nodes in network

**CODE** unique 4 character code for each node in the network

**VCOORD** vertical coordinate of each node in the network

**HCOORD** horizontal coordinate of each node in the network

**TRFLVL** traffic level of prestored set expressed in voice circuits

**NTRFC** number of traffic table additions to follow

**CODEL** four character code of node where traffic originates

**CODER** four character code of node where traffic terminates

**VCIRTS** amount of additional traffic from CODEL to CODER in voice circuits

These files can be created, deleted, or listed by the user through the INPUT module.

### 5.3.2 User Defined Tariffs

A NASA defined tariff can be defined based on the existing prestored licensed common carrier and specialized common carrier tariffs, both terrestrial and satellite. (Currently, these prestored tariffs are ATT, WU, and SBS). The model permits the storing in the data base of up to thirteen of these NASA defined tariffs. Following is a description of the structure of the data files which contain these tariffs.

#### 5.3.2.1 NASA Defined ATT Tariff

A NASA terrestrial tariff can be defined using the ATT terrestrial tariff as a point of departure. The user may redefine which cities are to be considered category A and may alter any of the mileage increment charges of the three schedules.
In specifying the category A cities, the user may add new cities to the category A list (defining each new city by its unique 4 character code) and may declare that a city that is category A for ATT will not be category A for this tariff.

The user may not alter the ATT channel charge mileage breakpoints. He may, however, alter the increment charges associated with each breakpoint.

The user will define this tariff using the input program and create a file named TARn where n can assume the values 001 to 013 as selected by the user. As the user specifies category A city additions and deletion, INPUT will arrange the city codes specified in alphabetic order with a -1 following deletions and a +1 following additions.

After the user has altered the increment charges, INPUT will recalculate the base charges to be consistent with these new increment charges. The file format follows:

```
TARIFF NETn (A16)
NADJ (I3)
OCCURRENCES
  CODE (A4) ADJ (I3)
    ... ...
NMLG (I3)
OCCURRENCES
  MLG (I10) BAS1 (F8.2) INC1 (F6.2) BAS2 (F8.2) INC2 (F6.2) BAS3 (F8.2) INC3 (F8.2)
```

where

- **NADJ** number of records following which describe adjustments to category A status of a rate center.
- **CODE** unique four character code of rate center which is being added to or deleted from the list of category A rate centers.
- **ADJ** code which indicates if this rate center is being added to or deleted from list of category A rate centers: 1 additions, -1 deletions.
- **NMLG** number of mileage breakpoints.
BAS1, base charge for each mileage breakpoint for schedules one, two, and three respectively
BAS2
BAS3
INC1, increment charge for each mileage breakpoint for the three schedules
INC2,
INC3
MLG mileage breakpoint

5.3.2.2 NASA Defined Western Union Tariff

A NASA satellite tariff can be defined using the Western Union satellite tariff a point of departure. The user may define satellite access city pair additions and deletions and may alter the channel charge for any of the three charge categories.

In specifying a new access city pair, the user may create a new satellite access city and may also link up two existing access cities which are not currently linked. When specifying a new linkage, the user may also define the tariff category associated with this pair (1: long haul, 2: medium haul, and 3: short haul). As stated above, the user may also delete from his set of access city pairs a link that exists in the Western Union tariff. He may also keep the link but change its rate.

The user will define this tariff using the input program and define a file name TARn where n can assume the values 001 to 013. As the user defines access city pair additions and deletions by specifying the four character code of the two cities, INPUT will arrange the two cities in alphabetic order by code. Following the code pair will be a 0 to indicate deletion of a link or a 1, 2, or 3 to indicate rate category of a desired linkage. Once all city pair additions and deletion have been specified, INPUT will arrange them within the file in alphabetic order by the left most code and then, within the left code, alphabetically by the right most code. The file format follows:
5.3.2.3 NASA Defined SBS Tariff

A NASA satellite tariff can be defined using the SBS satellite tariff as a point of departure. The user may change any of the rates specified in the tariff. When altering the CAU change, the user may not alter the CAU count breakpoint. He may however alter the increment charge.

The user will define this tariff using the INPUT module and create a file named TARn where n can assume the values 001 to 013 as selected by the user. After the user has finished altering the CAU increment rates, INPUT
will adjust the associated base rates. The user may specify NAC locations for the SBS mixed problem. The file format follows:

\[
\text{TARIFF NETn (A16)} \\
\text{NREC (I3) NENTRY (I3)} \\
\begin{align*}
\text{LOC1 (A5)...LOCn (A5)} \\
\text{NAC (F8.0)} \\
\text{SCU (F8.0)} \\
\text{FTU (F8.0)} \\
\text{CAUMIN (F8.0)} \\
\text{NTAB (I3)} \\
\text{NCAU (I10) BASE (F8.0) INC (F4.0)}
\end{align*}
\]

where

- **NREC**: number of records following that contain the NAC locations
- **NENTRY**: number of NACs specified
- **LOC1...LOCn**: four character code of rate center to contain NAC
- **NAC**: charge per Network Access Center
- **SCU**: charge per Supplemental Capacity Unit
- **FTU**: charge per Full Time Transmission Unit
- **CAUMIN**: minimum CAU charge per NAC
- **NTAB**: number of entries in following CAU rate table
- **NCAU**: number of CAU/NAC up to which the accompanying base and increment rate apply
- **BASE**: base charge for specified number of CAUs
- **INC**: increment charge for each CAU over a specified number
5.3.3 Execution Control File

The user must specify his optimization execution runstream using the INPUT modules EXCON option. At this time he will specify the runid, type of run (terrestrial only, Western Union Only, SBS only, mixed terrestrial - Western Union, or mixed SBS), name of the user defined network, any alteration to the traffic level specified in the network, the number of tariffs to be compared, the names of the tariffs (user defined and/or prestored), and the output tables desired.

In addition the user may desire to piggyback several of these problem definitions in one runstream. EXCON creates an execution control file, EXECUTE, to contain this description. The file format follows:

EXECUTION FILE (A16)
RUNID (A8)
NPROB (I3)
OUT1 (I2) OUT2 (I2) OUT3 (I2) OUT4 (I2) OUT5 (I2) OUT6 (I2)
NETFIL (A6)
LEVEL (I11)
NFIL (I3)
TARFIL (A6)

where
RUNID alphanumeric descriptor of this run
NPROB number of problems defined
OUT1... flag indicating if the associated output table is desired
OUT6 0 suppress
1 include
NETFIL name of user defined network definition
LEVEL traffic level to override that specified
If LEVEL = 0, the level specified in NETFIL will be used
NFIL number of tariffs to be compared
TARFIL    tariff to use
****    terminate each problem definition
## Appendix A - Module Cross Reference

This section contains a cross reference listing of the modules that make up the CNDC Model. The modules are listed in alphabetic order. By each module name are two lists: the first of other modules called by this module and the second of all modules which call this module.

<table>
<thead>
<tr>
<th>Module</th>
<th>Calls</th>
<th>Called By</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACC</td>
<td>none</td>
<td>none</td>
</tr>
<tr>
<td>ALDONE</td>
<td>none</td>
<td>SBSMIX</td>
</tr>
<tr>
<td>ATCOST</td>
<td>none</td>
<td>COSTNG DEFTER</td>
</tr>
<tr>
<td>ATTLST</td>
<td>none</td>
<td>CRATT CRATTB</td>
</tr>
<tr>
<td>BSTRCT</td>
<td>PTRPRS</td>
<td>INPUT INPUTB</td>
</tr>
<tr>
<td></td>
<td>TRFSKP</td>
<td></td>
</tr>
<tr>
<td>CATEG</td>
<td>PTRPRS</td>
<td>COSTNG OUTNU</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CNDC</td>
<td>COSTNG</td>
<td>none</td>
</tr>
<tr>
<td></td>
<td>INIT</td>
<td></td>
</tr>
<tr>
<td></td>
<td>INPUT</td>
<td></td>
</tr>
<tr>
<td></td>
<td>NETWRK</td>
<td></td>
</tr>
<tr>
<td></td>
<td>OUTPUT</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SBSMIX</td>
<td></td>
</tr>
<tr>
<td>CNDCB</td>
<td>COSTNG</td>
<td>none</td>
</tr>
<tr>
<td></td>
<td>INIT</td>
<td></td>
</tr>
<tr>
<td></td>
<td>INPUTB</td>
<td></td>
</tr>
<tr>
<td></td>
<td>NETWRK</td>
<td></td>
</tr>
<tr>
<td></td>
<td>OUTPUT</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SBSMIX</td>
<td></td>
</tr>
<tr>
<td>COSTNG</td>
<td>ATCOST</td>
<td>CNDC</td>
</tr>
<tr>
<td></td>
<td>CATEG</td>
<td>CNDCB</td>
</tr>
<tr>
<td></td>
<td>LISTED</td>
<td></td>
</tr>
<tr>
<td></td>
<td>MILES</td>
<td></td>
</tr>
<tr>
<td></td>
<td>ROWTOT</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SBCOST</td>
<td></td>
</tr>
<tr>
<td></td>
<td>WUCOST</td>
<td></td>
</tr>
<tr>
<td>Module</td>
<td>Calls</td>
<td>Called By</td>
</tr>
<tr>
<td>---------</td>
<td>------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>CRATT</td>
<td>ATTLST</td>
<td>CREATTT</td>
</tr>
<tr>
<td>CRATTB</td>
<td>ATTLST</td>
<td>TARB</td>
</tr>
<tr>
<td>CREATE</td>
<td>DEFNOD, FILDEF, GETPER, GETTFC, MNMPPTR, NCREAT, RVIFC, SMSA, USRNET</td>
<td>NETCON</td>
</tr>
<tr>
<td>CREATTT</td>
<td>CRATT, CRSSBS, CRWU, FILDEF, TCREAT</td>
<td>TARCON</td>
</tr>
<tr>
<td>CRSSBS</td>
<td>none</td>
<td>CREATTT</td>
</tr>
<tr>
<td>CRSSBSB</td>
<td>none</td>
<td>TARB</td>
</tr>
<tr>
<td>CRWU</td>
<td>WUPAIR</td>
<td>CREATTT</td>
</tr>
<tr>
<td>CRWUB</td>
<td>WUPAIR</td>
<td>TARB</td>
</tr>
<tr>
<td>DECIDE</td>
<td>none</td>
<td>SBSMIX</td>
</tr>
<tr>
<td>DEFNOD</td>
<td>SORT1</td>
<td>NCREAT, CREATE, NETB</td>
</tr>
<tr>
<td>DEFPRE</td>
<td>none</td>
<td>INIT</td>
</tr>
<tr>
<td>DEFTER</td>
<td>ATCOST, LISTED, MILES, MTPY, NETWRK, TRSTOT</td>
<td>SBSMIX</td>
</tr>
<tr>
<td>DELETE</td>
<td>FILDEF, NDELET</td>
<td>NETCON</td>
</tr>
<tr>
<td>DELETT</td>
<td>FILDEF, TDELET</td>
<td>TARCON</td>
</tr>
<tr>
<td>DIRGEN</td>
<td>none</td>
<td>EXCON, INIT</td>
</tr>
<tr>
<td>Module</td>
<td>Calls</td>
<td>Called By</td>
</tr>
<tr>
<td>---------</td>
<td>-------------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>EXCON</td>
<td>DIRGEN</td>
<td>INPUT</td>
</tr>
<tr>
<td></td>
<td>XCREAT</td>
<td></td>
</tr>
<tr>
<td></td>
<td>XLIST</td>
<td></td>
</tr>
<tr>
<td>EXCONB</td>
<td>EXCREA</td>
<td>INPUTB</td>
</tr>
<tr>
<td></td>
<td>FILDEF</td>
<td></td>
</tr>
<tr>
<td>EXCREA</td>
<td>none</td>
<td>XCREAT</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EXCONB</td>
</tr>
<tr>
<td>FACSIZ</td>
<td>none</td>
<td>OUT3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>OUT2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>OUT2SB</td>
</tr>
<tr>
<td></td>
<td></td>
<td>OUT3SB</td>
</tr>
<tr>
<td>FILDEF</td>
<td>none</td>
<td>CREATE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CREATT</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DELETE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DELETT</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LIST</td>
</tr>
<tr>
<td></td>
<td></td>
<td>LISTT</td>
</tr>
<tr>
<td></td>
<td></td>
<td>XCREAT</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EXCONB</td>
</tr>
<tr>
<td></td>
<td></td>
<td>NETB</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TARB</td>
</tr>
<tr>
<td>GETPER</td>
<td>none</td>
<td>CREATE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>NETB</td>
</tr>
<tr>
<td>GETRTE</td>
<td>none</td>
<td>OUT2SB</td>
</tr>
<tr>
<td></td>
<td></td>
<td>OUT3SB</td>
</tr>
<tr>
<td>GETTFC</td>
<td>TRFSKP</td>
<td>CREATE</td>
</tr>
<tr>
<td></td>
<td>WRTTRF</td>
<td>NETB</td>
</tr>
<tr>
<td>IBLATT</td>
<td>MNMPTTR</td>
<td>IBLTAR</td>
</tr>
<tr>
<td></td>
<td>TRFSKP</td>
<td></td>
</tr>
<tr>
<td>IBLLST</td>
<td>none</td>
<td>IBLNET</td>
</tr>
<tr>
<td>. IBLNET</td>
<td>IBLLST</td>
<td>INIT</td>
</tr>
<tr>
<td></td>
<td>IBLNOD</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IBLTRF</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IBLWHO</td>
<td></td>
</tr>
<tr>
<td>IBLNOD</td>
<td>none</td>
<td>IBLNET</td>
</tr>
<tr>
<td>IBLSBS</td>
<td>USRNET</td>
<td>IBLTAR</td>
</tr>
<tr>
<td>Module</td>
<td>Calls</td>
<td>Called By</td>
</tr>
<tr>
<td>-----------</td>
<td>-----------</td>
<td>-----------</td>
</tr>
<tr>
<td>IBLSB1</td>
<td>TRFSKP</td>
<td>IBLTAR</td>
</tr>
<tr>
<td>IBLTAR</td>
<td>IBLATT</td>
<td>INIT</td>
</tr>
<tr>
<td></td>
<td>IBLSBBS</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IBLSB1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IBLWU1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IBLWU2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>IBLWU3</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>IBLTRF</td>
<td>MNMPTTR</td>
<td>IBLNET</td>
</tr>
<tr>
<td></td>
<td>TRFSKP</td>
<td></td>
</tr>
<tr>
<td></td>
<td>WRTTRF</td>
<td></td>
</tr>
<tr>
<td>IBLWHO</td>
<td>none</td>
<td>IBLNET</td>
</tr>
<tr>
<td>IBLWU1</td>
<td>MNMPTTR</td>
<td>IBLTAR</td>
</tr>
<tr>
<td>IBLWU2</td>
<td>MNMPTTR</td>
<td>IBLTAR</td>
</tr>
<tr>
<td>IBLWU3</td>
<td>MNMPTTR</td>
<td>IBLTAR</td>
</tr>
<tr>
<td></td>
<td>PTRPRS</td>
<td></td>
</tr>
<tr>
<td>INDX</td>
<td>none</td>
<td>SRTNAC</td>
</tr>
<tr>
<td>INIT</td>
<td>DEFPRE</td>
<td>CNDC</td>
</tr>
<tr>
<td></td>
<td>DIRGEN</td>
<td>CNDCB</td>
</tr>
<tr>
<td></td>
<td>IBLNET</td>
<td>IBLTAR</td>
</tr>
<tr>
<td>INPUT</td>
<td>BSTRCT</td>
<td>CNDC</td>
</tr>
<tr>
<td></td>
<td>EXCON</td>
<td></td>
</tr>
<tr>
<td></td>
<td>NETCON</td>
<td>IBLTAR</td>
</tr>
<tr>
<td></td>
<td>TARCON</td>
<td></td>
</tr>
<tr>
<td>INPUTB</td>
<td>BSTRCT</td>
<td>CNDCB</td>
</tr>
<tr>
<td></td>
<td>EXCONB</td>
<td></td>
</tr>
<tr>
<td></td>
<td>NCLOSE</td>
<td>IBLTAR</td>
</tr>
<tr>
<td></td>
<td>NETB</td>
<td></td>
</tr>
<tr>
<td></td>
<td>NINIT</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TARB</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TCLOSE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TINIT</td>
<td></td>
</tr>
<tr>
<td>INTCHR</td>
<td>none</td>
<td>NCREAT</td>
</tr>
<tr>
<td></td>
<td></td>
<td>OUT2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>OUT2SB</td>
</tr>
<tr>
<td></td>
<td></td>
<td>OUT3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>OUT3SB</td>
</tr>
<tr>
<td>ISNAC</td>
<td>none</td>
<td>PRTITN</td>
</tr>
</tbody>
</table>

A-4
<table>
<thead>
<tr>
<th>Module</th>
<th>Calls</th>
<th>Called By</th>
</tr>
</thead>
<tbody>
<tr>
<td>LIST</td>
<td>FILDEF</td>
<td>NETCON</td>
</tr>
<tr>
<td></td>
<td>NLSTDR</td>
<td></td>
</tr>
<tr>
<td></td>
<td>NLSTFL</td>
<td></td>
</tr>
<tr>
<td>LISTED</td>
<td>none</td>
<td>COSTNG</td>
</tr>
<tr>
<td></td>
<td></td>
<td>OUTATT</td>
</tr>
<tr>
<td></td>
<td></td>
<td>OUT2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DEFTER</td>
</tr>
<tr>
<td></td>
<td></td>
<td>OUT2SB</td>
</tr>
<tr>
<td></td>
<td></td>
<td>OUT3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>OUT3SB</td>
</tr>
<tr>
<td>LISTT</td>
<td>FILDEF</td>
<td>TARCON</td>
</tr>
<tr>
<td></td>
<td>TLSTDR</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TLSTFL</td>
<td></td>
</tr>
<tr>
<td>MILES</td>
<td>none</td>
<td>COSTNG</td>
</tr>
<tr>
<td></td>
<td></td>
<td>OUT2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>OUT4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DEFTER</td>
</tr>
<tr>
<td></td>
<td></td>
<td>OUT2SB</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TOTCST</td>
</tr>
<tr>
<td>MUMPTR</td>
<td>none</td>
<td>IBLATT</td>
</tr>
<tr>
<td></td>
<td></td>
<td>IBLTRF</td>
</tr>
<tr>
<td></td>
<td></td>
<td>IBLWU1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>IBLWU2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>IBLWU3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SMSA</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CREATE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>NCREAT</td>
</tr>
<tr>
<td></td>
<td></td>
<td>RVTCF</td>
</tr>
<tr>
<td></td>
<td></td>
<td>USRNET</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SBSMIX</td>
</tr>
<tr>
<td>MTPLY</td>
<td>none</td>
<td>OUT2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>OUT2SB</td>
</tr>
<tr>
<td></td>
<td></td>
<td>OUT3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>OUT3SB</td>
</tr>
<tr>
<td>MTPY</td>
<td>none</td>
<td>OUT2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>OUT4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DEFTER</td>
</tr>
<tr>
<td></td>
<td></td>
<td>OUT2SB</td>
</tr>
<tr>
<td></td>
<td></td>
<td>OUT3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>OUT3SB</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PRTSBS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TOTCST</td>
</tr>
<tr>
<td>Module</td>
<td>Calls</td>
<td>Called By</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>NCLOSE</td>
<td>none</td>
<td>NETCON</td>
</tr>
<tr>
<td></td>
<td></td>
<td>NCREAT</td>
</tr>
<tr>
<td></td>
<td></td>
<td>NDELET</td>
</tr>
<tr>
<td></td>
<td></td>
<td>INPUTB</td>
</tr>
<tr>
<td>NCREAT</td>
<td>INTCHR</td>
<td>CREATE</td>
</tr>
<tr>
<td></td>
<td>MNMPTR</td>
<td>NETB</td>
</tr>
<tr>
<td></td>
<td>NCLOSE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TRESKIP</td>
<td></td>
</tr>
<tr>
<td></td>
<td>WRTTRF</td>
<td></td>
</tr>
<tr>
<td>NDELET</td>
<td>NCLOSE</td>
<td>DELETE</td>
</tr>
<tr>
<td>NETB</td>
<td>DEFNOD</td>
<td>INPUTB</td>
</tr>
<tr>
<td></td>
<td>FILDEF</td>
<td></td>
</tr>
<tr>
<td></td>
<td>GETPER</td>
<td></td>
</tr>
<tr>
<td></td>
<td>GETTFC</td>
<td></td>
</tr>
<tr>
<td></td>
<td>NCREAT</td>
<td></td>
</tr>
<tr>
<td></td>
<td>NLSTDR</td>
<td></td>
</tr>
<tr>
<td></td>
<td>NLSTFL</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SMSA</td>
<td></td>
</tr>
<tr>
<td></td>
<td>USRNET</td>
<td></td>
</tr>
<tr>
<td>NETCON</td>
<td>CREATE</td>
<td>INPUT</td>
</tr>
<tr>
<td></td>
<td>DELETE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>LIST</td>
<td></td>
</tr>
<tr>
<td></td>
<td>NCLOSE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>NINIT</td>
<td></td>
</tr>
<tr>
<td>NETWRK</td>
<td>none</td>
<td>CNDC</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DEFTER</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CNDCB</td>
</tr>
<tr>
<td>NINIT</td>
<td>none</td>
<td>NETCON</td>
</tr>
<tr>
<td></td>
<td></td>
<td>INPUTB</td>
</tr>
<tr>
<td>NLSTDR</td>
<td>none</td>
<td>LIST</td>
</tr>
<tr>
<td></td>
<td></td>
<td>XCREAT</td>
</tr>
<tr>
<td></td>
<td></td>
<td>NETB</td>
</tr>
<tr>
<td>NLSTFL</td>
<td>none</td>
<td>LIST</td>
</tr>
<tr>
<td></td>
<td></td>
<td>NETB</td>
</tr>
<tr>
<td>OUTATT</td>
<td>LISTED</td>
<td>OUT1</td>
</tr>
<tr>
<td>OUTPUT</td>
<td>OUT1</td>
<td>CNDC</td>
</tr>
<tr>
<td></td>
<td>OUT2</td>
<td>CNDCB</td>
</tr>
<tr>
<td></td>
<td>OUT3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>OUT4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>WRTTRF</td>
<td></td>
</tr>
<tr>
<td>Module</td>
<td>Calls</td>
<td>Called By</td>
</tr>
<tr>
<td>---------</td>
<td>---------------</td>
<td>-----------</td>
</tr>
<tr>
<td>OUTSBS</td>
<td>none</td>
<td>OUT1</td>
</tr>
<tr>
<td>OUTWU</td>
<td>CATEG</td>
<td>OUT1</td>
</tr>
<tr>
<td>OUT1</td>
<td>OUTATT, OUTSBS, OUTWU, SMSA</td>
<td>OUTPUT</td>
</tr>
<tr>
<td>OUT2</td>
<td>FACSIZ, INTCHR, LISTED, MILES, MTPY, MTPY, OUT2SB, PRTSBS</td>
<td>OUTPUT</td>
</tr>
<tr>
<td>OUT2SB</td>
<td>MILES, GETRTE, FACSIZ, MTPY, MTPY, MTPLY, INTCHR</td>
<td>OUT2</td>
</tr>
<tr>
<td>OUT3</td>
<td>FACSIZ, INTCHR, LISTED, MTPY, MTPY, OUT2SB, PRTSBS</td>
<td>OUTPUT</td>
</tr>
<tr>
<td>OUT3SB</td>
<td>FACSIZ, GETRTE, INTCHR, LISTED, MTPY, MTPY</td>
<td>OUT3</td>
</tr>
<tr>
<td>OUT4</td>
<td>MILES, MTPY, SUMEQP</td>
<td>OUTPUT</td>
</tr>
<tr>
<td>PRTITN</td>
<td>ISNAC</td>
<td>SBSMIX</td>
</tr>
<tr>
<td>PRTSBS</td>
<td>MTPY</td>
<td>OUT2, OUT3</td>
</tr>
<tr>
<td>Module</td>
<td>Calls</td>
<td>Called By</td>
</tr>
<tr>
<td>-----------</td>
<td>------------</td>
<td>--------------------</td>
</tr>
<tr>
<td>PTRPR5</td>
<td>none</td>
<td>IBLWU3 PTRPR5 TRFSKP CATEG OUT2 WUPAIR</td>
</tr>
<tr>
<td>ROWTOT</td>
<td>none</td>
<td>COSTNG</td>
</tr>
<tr>
<td>RVTFC</td>
<td>MNMPTTR</td>
<td>CREATE</td>
</tr>
<tr>
<td>SBCOST</td>
<td>none</td>
<td>COSTNG SBSC</td>
</tr>
<tr>
<td>SBSC</td>
<td>SBCOST</td>
<td>SBSCMIX</td>
</tr>
<tr>
<td>SBSMIX</td>
<td>ALDONE DECIDE DEFTER MNMPTTR PRTIIN SBSC SRTNAC TOTCST</td>
<td>CNDC CNDCB</td>
</tr>
<tr>
<td>SMSA</td>
<td>none</td>
<td>CREATE OUT1 NETB</td>
</tr>
<tr>
<td>SORTR</td>
<td>none</td>
<td>SRTNAC</td>
</tr>
<tr>
<td>SORT1</td>
<td>none</td>
<td>NCREAT TCRATT DEFNOD TCRSBS</td>
</tr>
<tr>
<td>SORTID</td>
<td>none</td>
<td>NCREAT TCREWU</td>
</tr>
<tr>
<td>SORT2</td>
<td>none</td>
<td>NCREAT TCREWU</td>
</tr>
<tr>
<td>SRTNAC</td>
<td>SORTR INDX</td>
<td>SBSMIX</td>
</tr>
<tr>
<td>SUMEQP</td>
<td>none</td>
<td>OUT4</td>
</tr>
<tr>
<td>TARB</td>
<td>CRATTB CRSBSB CRWUB FILDEF TCREAT</td>
<td>INPUTB</td>
</tr>
<tr>
<td>Module</td>
<td>Calls</td>
<td>Called By</td>
</tr>
<tr>
<td>----------</td>
<td>---------------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>TARCON</td>
<td>CREAT TDELETT LISTT TLSTTWU TINIT</td>
<td>INPUT</td>
</tr>
<tr>
<td>TARTYP</td>
<td>none</td>
<td>OUT1</td>
</tr>
<tr>
<td>TCLOSE</td>
<td>none</td>
<td>TARCON TCREAT TDELETT INPUTB</td>
</tr>
<tr>
<td>TCRATT</td>
<td>SORT1</td>
<td>TCREAT</td>
</tr>
<tr>
<td>TCREAT</td>
<td>TCLOSE TCRATT TCRSBS TCRWU</td>
<td>CREAT TARB</td>
</tr>
<tr>
<td>TCRSBS</td>
<td>SORT1</td>
<td>TCREAT</td>
</tr>
<tr>
<td>TCREWU</td>
<td>SORT2</td>
<td>TCREAT</td>
</tr>
<tr>
<td>TDELET</td>
<td>TCLOSE</td>
<td>DELETT</td>
</tr>
<tr>
<td>TINIT</td>
<td>none</td>
<td>TARCON INPUTB</td>
</tr>
<tr>
<td>TLSTAT</td>
<td>none</td>
<td>TLSTFL</td>
</tr>
<tr>
<td>TLSTDR</td>
<td>none</td>
<td>LISTT XCREAT</td>
</tr>
<tr>
<td>TLSTFL</td>
<td>TLSTAT TLSTSB TLSTWU</td>
<td>LISTT</td>
</tr>
<tr>
<td>TLSTSB</td>
<td>none</td>
<td>TLSTFL</td>
</tr>
<tr>
<td>TLSTWU</td>
<td>none</td>
<td>TLSTFL</td>
</tr>
<tr>
<td>TOTCST</td>
<td>MILES MTPY TRSTOT</td>
<td>SBSMIX</td>
</tr>
<tr>
<td>TRFSKP</td>
<td>none</td>
<td>IBLATT IBLTRF IBLSB1 NCREAT GETTFC</td>
</tr>
<tr>
<td>Module</td>
<td>Calls</td>
<td>Called By</td>
</tr>
<tr>
<td>-----------</td>
<td>---------</td>
<td>--------------------</td>
</tr>
<tr>
<td>1RSTOT</td>
<td>none</td>
<td>TOTCST</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DEFTER</td>
</tr>
<tr>
<td>USRNET</td>
<td>NMMPTR</td>
<td>CREATE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>IBLSBS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>NETB</td>
</tr>
<tr>
<td>WRTTRF</td>
<td>none</td>
<td>IBLTRF</td>
</tr>
<tr>
<td></td>
<td></td>
<td>NCREAT</td>
</tr>
<tr>
<td></td>
<td></td>
<td>OUTPUT</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GETTC</td>
</tr>
<tr>
<td>WUCOST</td>
<td>none</td>
<td>COSTNG</td>
</tr>
<tr>
<td>WUPAIR</td>
<td>PTRPRS</td>
<td>CRWU</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CRWUB</td>
</tr>
<tr>
<td>XCREAT</td>
<td>EXCREA</td>
<td>EXCON</td>
</tr>
<tr>
<td></td>
<td>FILDEF</td>
<td></td>
</tr>
<tr>
<td></td>
<td>NLSTDR</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TLSTDR</td>
<td></td>
</tr>
<tr>
<td>XLIST</td>
<td>none</td>
<td>EXCON</td>
</tr>
</tbody>
</table>
Following is a list of all the error messages within the model. The fatal errors cause model execution to be terminated. Fatal errors may indicate that files within the database have errors within them. They may also indicate that code within the model has been altered incompletely. Fatal errors require the intervention of a programmer and will not occur under normal conditions.

Nonfatal errors occur during initialization of a problem during optimization. They indicate that specifications within a problem are invalid, perhaps because the user is accessing a user defined file which is no longer valid because of tariff changes. Again, these errors should not occur under normal conditions.

**FATAL ERRORS**

<table>
<thead>
<tr>
<th>Number</th>
<th>Module</th>
<th>Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>1010</td>
<td>INIT</td>
<td>Execution Control file header not as expected</td>
</tr>
<tr>
<td>1020</td>
<td>IBLLST</td>
<td>file header of prestored ATT tariff file not as expected</td>
</tr>
<tr>
<td>1030</td>
<td>IBLNET</td>
<td>file header of user defined network not as expected</td>
</tr>
<tr>
<td>1040</td>
<td>IBLTRF</td>
<td>file header of prestored traffic table not as expected</td>
</tr>
<tr>
<td>1050</td>
<td>IBLATT</td>
<td>header record of tariff file not as expected</td>
</tr>
<tr>
<td>1060</td>
<td>IBLATT</td>
<td>header of prestored ATT file not as expected</td>
</tr>
<tr>
<td>1070</td>
<td>IBLWU1</td>
<td>header record of prestored WU tariff not as expected</td>
</tr>
<tr>
<td>1080</td>
<td>IBLWU2</td>
<td>file header of user defined WU tariff not as expected</td>
</tr>
<tr>
<td>1090</td>
<td>IBLWU3</td>
<td>file header of prestored WU tariff not as expected</td>
</tr>
<tr>
<td>Number</td>
<td>Module</td>
<td>Message</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td>---------</td>
</tr>
<tr>
<td>1100</td>
<td>IBLWU3</td>
<td>file header of user defined WU tariff not as expected</td>
</tr>
<tr>
<td>1110</td>
<td>IBLSB1</td>
<td>file header of prestored SBS tariff not as expected</td>
</tr>
<tr>
<td>1120</td>
<td>IBLSBS</td>
<td>file header of user defined SBS tariff not as expected</td>
</tr>
<tr>
<td>1130</td>
<td>IBLSBS</td>
<td>file header of prestored SBS tariff not as expected</td>
</tr>
<tr>
<td>2010</td>
<td>BSTRCT</td>
<td>header from traffic nodes not as expected</td>
</tr>
<tr>
<td>2020</td>
<td>BSTRCT</td>
<td>header from prestored ATT tariff not as expected</td>
</tr>
<tr>
<td>2030</td>
<td>BSTRCT</td>
<td>ATT tariff levels exceeds array size in model</td>
</tr>
<tr>
<td>2040</td>
<td>BSTRCT</td>
<td>header from prestored SBS tariff not as expected</td>
</tr>
<tr>
<td>2050</td>
<td>BSTRCT</td>
<td>SBS tariff levels exceeds array size in model</td>
</tr>
<tr>
<td>2060</td>
<td>BSTRCT</td>
<td>header from prestored WU tariff not as expected</td>
</tr>
<tr>
<td>2070</td>
<td>BSTRCT</td>
<td>number of items in prestored network array exceeds number allowed</td>
</tr>
<tr>
<td>2080</td>
<td>BSTRCT</td>
<td>number of items in prestored category A list exceeds number allowed</td>
</tr>
<tr>
<td>2090</td>
<td>BSTRCT</td>
<td>number of items required for WU access cities exceeds number allowed</td>
</tr>
<tr>
<td>3010</td>
<td>NINIT</td>
<td>file header of network directory not as expected</td>
</tr>
<tr>
<td>3020</td>
<td>NINIT</td>
<td>number of files in directories exceeds the capacity of arrays</td>
</tr>
<tr>
<td>3030</td>
<td>GETTFC</td>
<td>file header of prestored traffic table not as expected</td>
</tr>
<tr>
<td>3040</td>
<td>GETPER</td>
<td>file header of TRFVOL file not as expected</td>
</tr>
<tr>
<td>3050</td>
<td>GETPER</td>
<td>desired percent not found in table</td>
</tr>
<tr>
<td>3060</td>
<td>DEFNOD</td>
<td>number of user defined nodes not number put into array</td>
</tr>
<tr>
<td>3070</td>
<td>NCREAT</td>
<td>file specified for network file already defined</td>
</tr>
<tr>
<td>Number</td>
<td>Module</td>
<td>Message</td>
</tr>
<tr>
<td>--------</td>
<td>----------</td>
<td>-------------------------------------------------------------------------</td>
</tr>
<tr>
<td>3080</td>
<td>NCREAT</td>
<td>file specified for network file not in directory</td>
</tr>
<tr>
<td>3090</td>
<td>NDELET</td>
<td>file name to delete not in directory</td>
</tr>
<tr>
<td>3100</td>
<td>NLSTFL</td>
<td>file specified to be printed not in directory</td>
</tr>
<tr>
<td>3110</td>
<td>NLSTFL</td>
<td>header record of file not as expected</td>
</tr>
<tr>
<td>4010</td>
<td>TINIT</td>
<td>file header of tariff directory not as expected</td>
</tr>
<tr>
<td>4020</td>
<td>TINIT</td>
<td>number of files in directory exceeds the capacity of array in model</td>
</tr>
<tr>
<td>4030</td>
<td>TCREAT</td>
<td>file name to create not in directory</td>
</tr>
<tr>
<td>4040</td>
<td>TCREAT</td>
<td>file specified already created</td>
</tr>
<tr>
<td>4050</td>
<td>TCREAT</td>
<td>file type to be created invalid</td>
</tr>
<tr>
<td>4060</td>
<td>TDELET</td>
<td>file to be deleted not in directory</td>
</tr>
<tr>
<td>4070</td>
<td>TLSTFL</td>
<td>file to list not in directory</td>
</tr>
<tr>
<td>4080</td>
<td>TLSTFL</td>
<td>file to be listed empty</td>
</tr>
<tr>
<td>4090</td>
<td>TLSTFL</td>
<td>file type of tariff to be listed is invalid</td>
</tr>
<tr>
<td>4100</td>
<td>TLSTAT</td>
<td>file header of ATT tariff invalid</td>
</tr>
<tr>
<td>4110</td>
<td>TLSTWU</td>
<td>file header of WU tariff invalid</td>
</tr>
<tr>
<td>4120</td>
<td>TLSTSB</td>
<td>file header of SBS tariff invalid</td>
</tr>
<tr>
<td>5000</td>
<td>ACC</td>
<td>input arguments out of range</td>
</tr>
<tr>
<td>5010</td>
<td>ATCOST</td>
<td>input arguments out of range</td>
</tr>
<tr>
<td>5020</td>
<td>CATEG</td>
<td>input arguments out of range</td>
</tr>
<tr>
<td>5030</td>
<td>LISTED</td>
<td>input arguments out of range</td>
</tr>
<tr>
<td>5040</td>
<td>MILES</td>
<td>input arguments out of range</td>
</tr>
<tr>
<td>5050</td>
<td>PTRPRS</td>
<td>input arguments out of range</td>
</tr>
<tr>
<td>5060</td>
<td>ROWTOT</td>
<td>input arguments out of range</td>
</tr>
<tr>
<td>5070</td>
<td>SBCOST</td>
<td>input arguments out of range</td>
</tr>
<tr>
<td>5080</td>
<td>TARTYP</td>
<td>input arguments out of range</td>
</tr>
<tr>
<td>Number</td>
<td>Module</td>
<td>Message</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td>---------</td>
</tr>
<tr>
<td>5090</td>
<td>DEFPRE</td>
<td>file header of prestored network file not as expected</td>
</tr>
<tr>
<td>5100</td>
<td>DIRGEN</td>
<td>file header from tariff directory not as expected</td>
</tr>
<tr>
<td>5110</td>
<td>DIRGEN</td>
<td>file header from network directory not as expected</td>
</tr>
<tr>
<td>5120</td>
<td>FACSIZ</td>
<td>number of input voice circuits out of range</td>
</tr>
<tr>
<td>5130</td>
<td>RVTFI</td>
<td>nodes specified don't exist in user defined network</td>
</tr>
<tr>
<td>5140</td>
<td>WUCOST</td>
<td>input arguments out of range</td>
</tr>
<tr>
<td>5150</td>
<td>FILDEF</td>
<td>directory requested is invalid</td>
</tr>
<tr>
<td>B1000</td>
<td>INPUTB</td>
<td>invalid option code</td>
</tr>
<tr>
<td>B2010</td>
<td>NETB</td>
<td>invalid network file name</td>
</tr>
<tr>
<td>B2020</td>
<td>NETB</td>
<td>network file already defined</td>
</tr>
<tr>
<td>B2030</td>
<td>NETB</td>
<td>invalid network option</td>
</tr>
<tr>
<td>B2040</td>
<td>NETB</td>
<td>invalid prestored city code</td>
</tr>
<tr>
<td>B2050</td>
<td>NETB</td>
<td>exceeded number of prestored nodes</td>
</tr>
<tr>
<td>B2060</td>
<td>NETB</td>
<td>invalid percent for selecting prestored nodes</td>
</tr>
<tr>
<td>B2070</td>
<td>NETB</td>
<td>invalid traffic level</td>
</tr>
<tr>
<td>B2080</td>
<td>NETB</td>
<td>maximum additional cities exceeded</td>
</tr>
<tr>
<td>B2090</td>
<td>NETB</td>
<td>invalid additional traffic volume</td>
</tr>
<tr>
<td>22100</td>
<td>NETB</td>
<td>additional traffic entry city not in network</td>
</tr>
<tr>
<td>B2110</td>
<td>NETB</td>
<td>traffic table spool option invalid</td>
</tr>
<tr>
<td>B2120</td>
<td>NETB</td>
<td>invalid additional city coordinate</td>
</tr>
<tr>
<td>B3000</td>
<td>TARBB</td>
<td>invalid tariff filename</td>
</tr>
<tr>
<td>B3010</td>
<td>TARBB</td>
<td>tariff file already defined</td>
</tr>
<tr>
<td>B3020</td>
<td>TARBB</td>
<td>invalid tariff file type</td>
</tr>
<tr>
<td>B3100</td>
<td>CRATTB</td>
<td>invalid listed city addition/deletion</td>
</tr>
<tr>
<td>Number</td>
<td>Module</td>
<td>Message</td>
</tr>
<tr>
<td>--------</td>
<td>----------</td>
<td>---------------------------------------------------</td>
</tr>
<tr>
<td>B3110</td>
<td>CRATTB</td>
<td>listed city addition/deletion duplication</td>
</tr>
<tr>
<td>B4000</td>
<td>EXCONB</td>
<td>negative number of restarts specified</td>
</tr>
<tr>
<td>B4010</td>
<td>EXCONB</td>
<td>invalid option</td>
</tr>
<tr>
<td>B4020</td>
<td>EXCONB</td>
<td>undefined network file</td>
</tr>
<tr>
<td>B4030</td>
<td>EXCONB</td>
<td>undefined tariff file</td>
</tr>
<tr>
<td>B4040</td>
<td>EXCONB</td>
<td>tariff not ATT-type</td>
</tr>
<tr>
<td>B4050</td>
<td>EXCONB</td>
<td>invalid file type</td>
</tr>
<tr>
<td>B4060</td>
<td>EXCONB</td>
<td>negative number of tariff files specified</td>
</tr>
<tr>
<td>B4070</td>
<td>EXCONB</td>
<td>no valid tariff files specified</td>
</tr>
<tr>
<td>Number</td>
<td>Module</td>
<td>Message</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td>---------</td>
</tr>
<tr>
<td>1510</td>
<td>INIT</td>
<td>error encountered in building tariff input for specified problem</td>
</tr>
<tr>
<td>1520</td>
<td>INIT</td>
<td>number of tariffs specified in execution control file out of range</td>
</tr>
<tr>
<td>1530</td>
<td>INIT</td>
<td>error encountered in building data structures for specified case - run option out of range</td>
</tr>
<tr>
<td>1540</td>
<td>INIT</td>
<td>error encountered in building data structures for specified case - run option out of range</td>
</tr>
<tr>
<td>1550</td>
<td>INIT</td>
<td>error encountered in input for specified case - proceeding to next case</td>
</tr>
<tr>
<td>1560</td>
<td>IBLNET</td>
<td>network file requested is not in directory</td>
</tr>
<tr>
<td>1570</td>
<td>IBLNET</td>
<td>network file requested is not defined</td>
</tr>
<tr>
<td>1580</td>
<td>IBLTAR</td>
<td>requested file not defined</td>
</tr>
<tr>
<td>1590</td>
<td>IBLTAR</td>
<td>status of file out of range</td>
</tr>
<tr>
<td>1600</td>
<td>IBLTAR</td>
<td>mismatch between run type and tariff file type</td>
</tr>
<tr>
<td>1610</td>
<td>IBLTAR</td>
<td>more than two tariff files specified for mixed run</td>
</tr>
<tr>
<td>1620</td>
<td>IBLTAR</td>
<td>tariff option out of range</td>
</tr>
<tr>
<td>1630</td>
<td>IBLTAR</td>
<td>for mixed mode there should be two tariffs defined - one for as ATT tariff, the other a satellite tariff</td>
</tr>
<tr>
<td>1640</td>
<td>IBLTAR</td>
<td>number of breakpoint specified in ATT tariff exceeds array capacity in model</td>
</tr>
<tr>
<td>1650</td>
<td>IBLTAR</td>
<td>additional traffic specified for a node not in the user defined network</td>
</tr>
<tr>
<td>1660</td>
<td>IBLTAR</td>
<td>number of mileage breakpoints specified in tariff file not as expected</td>
</tr>
<tr>
<td>1670</td>
<td>IBLTAR</td>
<td>mileage breakpoint specified in tariff file not as expected</td>
</tr>
<tr>
<td>1680</td>
<td>IBLWU3</td>
<td>number of satellite access for the problem exceeds array capacity for this model</td>
</tr>
<tr>
<td>Number</td>
<td>Module</td>
<td>Message</td>
</tr>
<tr>
<td>--------</td>
<td>---------</td>
<td>---------------------------------------------------------</td>
</tr>
<tr>
<td>1690</td>
<td>IBLSB1</td>
<td>number of entries in CAU cost table greater than capacity of arrays in model</td>
</tr>
<tr>
<td>1700</td>
<td>IBLSBS</td>
<td>number of breakpoints in SBS table not as expected</td>
</tr>
<tr>
<td>1710</td>
<td>IBLSBS</td>
<td>breakpoint values in SBS table not as in prestored tariff</td>
</tr>
<tr>
<td>3510</td>
<td>NDELET</td>
<td>file name to delete has already been deleted</td>
</tr>
<tr>
<td>3520</td>
<td>NLSTFL</td>
<td>file specified to be printed not defined</td>
</tr>
<tr>
<td>4510</td>
<td>TDELET</td>
<td>file name to delete has already been deleted</td>
</tr>
<tr>
<td>6000</td>
<td>OUT1</td>
<td>tariff in problem has invalid type</td>
</tr>
<tr>
<td>B2500</td>
<td>NETB</td>
<td>prestored node duplication</td>
</tr>
<tr>
<td>B2510</td>
<td>NETB</td>
<td>additional city a prestored node</td>
</tr>
<tr>
<td>B2520</td>
<td>NETB</td>
<td>additional city duplication</td>
</tr>
<tr>
<td>B2530</td>
<td>NETB</td>
<td>additional traffic pair duplication</td>
</tr>
<tr>
<td>B3500</td>
<td>CRATTB</td>
<td>city to be added already listed</td>
</tr>
<tr>
<td>B3510</td>
<td>CRATTB</td>
<td>city to be deleted not a listed city</td>
</tr>
<tr>
<td>B3600</td>
<td>CRWUB</td>
<td>access pair category change erroneous</td>
</tr>
<tr>
<td>B3610</td>
<td>CRWUB</td>
<td>access pair to be deleted not in list</td>
</tr>
<tr>
<td>B3620</td>
<td>CRWUB</td>
<td>category out of range</td>
</tr>
<tr>
<td>B3630</td>
<td>CRWUB</td>
<td>access pair duplication</td>
</tr>
<tr>
<td>B4500</td>
<td>EXCONB</td>
<td>maximum problems exceeded</td>
</tr>
<tr>
<td>B4510</td>
<td>EXCONB</td>
<td>maximum tariffs exceeded</td>
</tr>
<tr>
<td>B4520</td>
<td>EXCONB</td>
<td>tariff file duplication</td>
</tr>
<tr>
<td>B4530</td>
<td>EXCONB</td>
<td>undefined tariff file</td>
</tr>
<tr>
<td>B4540</td>
<td>EXCONB</td>
<td>invalid file type</td>
</tr>
<tr>
<td>B4550</td>
<td>EXCONB</td>
<td>invalid output table number</td>
</tr>
</tbody>
</table>
Appendix C - Prestored ATT Tariff

This appendix defines the American Telephone and Telegraph Company tariff as defined in F.C.C. No. 260 (effective March 3, 1982). This tariff is referred to as the prestored ATT tariff in the CNDC model.

Per airline mile, per month

(a) Schedule I (applies between a pair of Category "A" rate centers)

<table>
<thead>
<tr>
<th>Mileage</th>
<th>Charge</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$ 73.56</td>
</tr>
<tr>
<td>2-14</td>
<td>$ 73.56 + $2.59 for each mile over 1 mile</td>
</tr>
<tr>
<td>15</td>
<td>$ 109.82</td>
</tr>
<tr>
<td>16-24</td>
<td>$ 109.82 + $2.16 for each mile over 15 miles</td>
</tr>
<tr>
<td>25</td>
<td>$ 131.42</td>
</tr>
<tr>
<td>26-39</td>
<td>$ 131.42 + $1.62 for each mile over 25 miles</td>
</tr>
<tr>
<td>40</td>
<td>$ 155.72</td>
</tr>
<tr>
<td>41-59</td>
<td>$ 155.72 + $1.62 for each mile over 40 miles</td>
</tr>
<tr>
<td>60</td>
<td>$ 188.12</td>
</tr>
<tr>
<td>61-79</td>
<td>$ 188.12 + $1.62 for each mile over 60 miles</td>
</tr>
<tr>
<td>80</td>
<td>$ 220.52</td>
</tr>
<tr>
<td>81-89</td>
<td>$ 220.52 + $1.62 for each mile over 80 miles</td>
</tr>
<tr>
<td>100</td>
<td>$ 252.92</td>
</tr>
<tr>
<td>101-999</td>
<td>$ 252.92 + $.94 for each mile over 100 miles</td>
</tr>
<tr>
<td>1000</td>
<td>$1,098.92</td>
</tr>
<tr>
<td>Over 1000</td>
<td>$1,098.92 + $.58 for each mile over 1000 miles</td>
</tr>
</tbody>
</table>

(b) Schedule II (applies between a pair of rate centers where one rate center is in Category "A" and the other rate center of the same pair of rate centers is in Category "B")

<table>
<thead>
<tr>
<th>Mileage</th>
<th>Charge</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$ 75.00</td>
</tr>
<tr>
<td>2-14</td>
<td>$ 75.00 + $4.77 for each mile over 1 mile</td>
</tr>
<tr>
<td>15</td>
<td>$ 141.78</td>
</tr>
<tr>
<td>16-24</td>
<td>$ 141.78 + $4.47 for each mile over 15 miles</td>
</tr>
<tr>
<td>25</td>
<td>$ 186.48</td>
</tr>
<tr>
<td>26-39</td>
<td>$ 186.48 + $2.89 for each mile over 25 miles</td>
</tr>
<tr>
<td>40</td>
<td>$ 229.83</td>
</tr>
<tr>
<td>41-59</td>
<td>$ 229.83 + $1.95 for each mile over 40 miles</td>
</tr>
<tr>
<td>60</td>
<td>$ 268.83</td>
</tr>
<tr>
<td>61-79</td>
<td>$ 268.83 + $1.95 for each mile over 60 miles</td>
</tr>
<tr>
<td>80</td>
<td>$ 307.83</td>
</tr>
<tr>
<td>81-89</td>
<td>$ 307.83 + $1.95 for each mile over 80 miles</td>
</tr>
<tr>
<td>100</td>
<td>$ 346.83</td>
</tr>
<tr>
<td>101-999</td>
<td>$ 346.83 + $.94 for each mile over 100 miles</td>
</tr>
<tr>
<td>1000</td>
<td>$1,192.83</td>
</tr>
<tr>
<td>Over 1000</td>
<td>$1,192.83 + $.58 for each mile over 1000 miles</td>
</tr>
</tbody>
</table>
(c) Schedule III (applies between a pair of Category "B" rate centers).

<table>
<thead>
<tr>
<th>Mileage</th>
<th>Charge</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$76.43</td>
</tr>
<tr>
<td>2-14</td>
<td>$76.43 + $6.35 for each mile over 1 mile</td>
</tr>
<tr>
<td>15</td>
<td>$165.33</td>
</tr>
<tr>
<td>16-24</td>
<td>$165.33 + $5.48 for each mile over 15 miles</td>
</tr>
<tr>
<td>25</td>
<td>$220.13</td>
</tr>
<tr>
<td>26-39</td>
<td>$220.13 + $4.03 for each mile over 25 miles</td>
</tr>
<tr>
<td>40</td>
<td>$280.58</td>
</tr>
<tr>
<td>41-59</td>
<td>$280.58 + $3.03 for each mile over 40 miles</td>
</tr>
<tr>
<td>60</td>
<td>$341.18</td>
</tr>
<tr>
<td>61-79</td>
<td>$341.18 + $2.31 for each mile over 60 miles</td>
</tr>
<tr>
<td>80</td>
<td>$387.38</td>
</tr>
<tr>
<td>81-89</td>
<td>$387.38 + $1.95 for each mile over 80 miles</td>
</tr>
<tr>
<td>100</td>
<td>$426.38</td>
</tr>
<tr>
<td>101-999</td>
<td>$426.38 + $ .97 for each mile over 100 miles</td>
</tr>
<tr>
<td>1000</td>
<td>$1,299.38</td>
</tr>
<tr>
<td>Over 1000</td>
<td>$1,299.38 + $ .58 for each mile over 1000 miles</td>
</tr>
</tbody>
</table>
Following is a directory of those rate centers classified as "Category A", that is, those rate centers to which schedule I and II rates apply within the American Telephone and Telegraph Company as defined in F.C.C. No. 260 (effective July 16, 1983). Next to each rate center is the four character code used to describe it in the CNDC model. In those cases where the rate center is an SMSA, that is, it is in the prestored network, the four character code is the SMSA code used. When the rate center has a city name in parenthesis next to it, the rate center and its paired city are considered as one SMSA in the prestored network and will be viewed as one in the ATT tariff modules.
<table>
<thead>
<tr>
<th>RATE CENTER</th>
<th>CODE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alabama</td>
<td></td>
</tr>
<tr>
<td>Anniston</td>
<td>ANAL</td>
</tr>
<tr>
<td>Birmingham</td>
<td>BIAL</td>
</tr>
<tr>
<td>Decatur</td>
<td>DEAL</td>
</tr>
<tr>
<td>Huntsville</td>
<td>HUAL</td>
</tr>
<tr>
<td>Mobile</td>
<td>MOAL</td>
</tr>
<tr>
<td>Montgomery</td>
<td>MNAL</td>
</tr>
<tr>
<td>Troy</td>
<td>TRAL</td>
</tr>
<tr>
<td>Arizona</td>
<td></td>
</tr>
<tr>
<td>Flagstaff</td>
<td>FLAR</td>
</tr>
<tr>
<td>Phoenix</td>
<td>PHAZ</td>
</tr>
<tr>
<td>Prescott</td>
<td>PRAR</td>
</tr>
<tr>
<td>Tucson</td>
<td>TUAR</td>
</tr>
<tr>
<td>Yuma</td>
<td>YUAR</td>
</tr>
<tr>
<td>Arkansas</td>
<td></td>
</tr>
<tr>
<td>Fayetteville</td>
<td>FAAR</td>
</tr>
<tr>
<td>Forrest City</td>
<td>FOAR</td>
</tr>
<tr>
<td>Hot Springs</td>
<td>HOAR</td>
</tr>
<tr>
<td>Jonesboro</td>
<td>JOAR</td>
</tr>
<tr>
<td>Little Rock</td>
<td>LIAR</td>
</tr>
<tr>
<td>Pine Bluff</td>
<td>PIAR</td>
</tr>
<tr>
<td>Searcy</td>
<td>SEAR</td>
</tr>
<tr>
<td>California</td>
<td></td>
</tr>
<tr>
<td>Anaheim</td>
<td>ANCA</td>
</tr>
<tr>
<td>Bakersfield</td>
<td>BACA</td>
</tr>
<tr>
<td>Chico</td>
<td>CHCA</td>
</tr>
<tr>
<td>Eureka</td>
<td>EUCA</td>
</tr>
<tr>
<td>Fresno</td>
<td>FRCA</td>
</tr>
<tr>
<td>Garoena</td>
<td>GACA</td>
</tr>
<tr>
<td>Hayward</td>
<td>HACA</td>
</tr>
<tr>
<td>Long Beach (Los Angeles)</td>
<td>LOCA</td>
</tr>
<tr>
<td>Los Angeles</td>
<td>LOCA</td>
</tr>
<tr>
<td>Oakland (San Francisco)</td>
<td>SFCA</td>
</tr>
<tr>
<td>Redwood City</td>
<td>RCAC</td>
</tr>
<tr>
<td>Sacramento</td>
<td>SACA</td>
</tr>
<tr>
<td>Salinas</td>
<td>SLCA</td>
</tr>
<tr>
<td>San Bernardina (Riverside)</td>
<td>RICA</td>
</tr>
<tr>
<td>San Diego</td>
<td>SNCA</td>
</tr>
<tr>
<td>San Francisco</td>
<td>SFCA</td>
</tr>
<tr>
<td>San Jose</td>
<td>SJCA</td>
</tr>
<tr>
<td>San Luis Obispo</td>
<td>SUCA</td>
</tr>
<tr>
<td>Santa Monica</td>
<td>STCA</td>
</tr>
<tr>
<td>Santa Rosa</td>
<td>SRCA</td>
</tr>
<tr>
<td>Stockton</td>
<td>SOCA</td>
</tr>
<tr>
<td>Sunnyvale</td>
<td>SYCA</td>
</tr>
<tr>
<td>Ukiah</td>
<td>UKCA</td>
</tr>
<tr>
<td>Van Nuys</td>
<td>VNCA</td>
</tr>
<tr>
<td>RATE CENTER</td>
<td>CODE</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>------</td>
</tr>
<tr>
<td>Colorado</td>
<td></td>
</tr>
<tr>
<td>Colorado Springs</td>
<td>COCO</td>
</tr>
<tr>
<td>Denver</td>
<td>DECO</td>
</tr>
<tr>
<td>Fort Collins</td>
<td>FOCO</td>
</tr>
<tr>
<td>Fort Morgan</td>
<td>FRCO</td>
</tr>
<tr>
<td>Glenwood Springs</td>
<td>GLCO</td>
</tr>
<tr>
<td>Grand Junction</td>
<td>GACO</td>
</tr>
<tr>
<td>Greeley</td>
<td>CRCO</td>
</tr>
<tr>
<td>Montrose</td>
<td>MOCO</td>
</tr>
<tr>
<td>Pueblo</td>
<td>PUCO</td>
</tr>
<tr>
<td>Connecticut</td>
<td></td>
</tr>
<tr>
<td>Bethany</td>
<td>BECT</td>
</tr>
<tr>
<td>Bloomfield</td>
<td>BLCT</td>
</tr>
<tr>
<td>Bridgeport</td>
<td>BRCT</td>
</tr>
<tr>
<td>Brookfield</td>
<td>BOCT</td>
</tr>
<tr>
<td>East Hartford</td>
<td>EACT</td>
</tr>
<tr>
<td>Groton</td>
<td>GRCT</td>
</tr>
<tr>
<td>Hamden</td>
<td>HMCT</td>
</tr>
<tr>
<td>Hartford</td>
<td>HACT</td>
</tr>
<tr>
<td>New Haven</td>
<td>NWCT</td>
</tr>
<tr>
<td>New London</td>
<td>NLCT</td>
</tr>
<tr>
<td>North Haven</td>
<td>NOCT</td>
</tr>
<tr>
<td>Orange</td>
<td>ORCT</td>
</tr>
<tr>
<td>Stamford</td>
<td>STCT</td>
</tr>
<tr>
<td>Stratford</td>
<td>SACT</td>
</tr>
<tr>
<td>West Hartford</td>
<td>WECT</td>
</tr>
<tr>
<td>West Haven</td>
<td>WSCT</td>
</tr>
<tr>
<td>Wethersfield</td>
<td>WTCT</td>
</tr>
<tr>
<td>Delaware</td>
<td></td>
</tr>
<tr>
<td>Wilmington</td>
<td>WIDE</td>
</tr>
<tr>
<td>District of Columbia</td>
<td>WADC</td>
</tr>
<tr>
<td>Florida</td>
<td></td>
</tr>
<tr>
<td>Chipley</td>
<td>CHFL</td>
</tr>
<tr>
<td>Clearwater</td>
<td>CLFL</td>
</tr>
<tr>
<td>Cocoa (Melbourne)</td>
<td>MEFL</td>
</tr>
<tr>
<td>Crestview</td>
<td>CRFL</td>
</tr>
<tr>
<td>Daytona Beach</td>
<td>DAFL</td>
</tr>
<tr>
<td>Fort Lauderdale</td>
<td>FOFL</td>
</tr>
<tr>
<td>Fort Meyers</td>
<td>FRFL</td>
</tr>
<tr>
<td>Fort Pierce</td>
<td>FPFL</td>
</tr>
<tr>
<td>Fort Walton Beach</td>
<td>FTFL</td>
</tr>
<tr>
<td>Gainesville</td>
<td>GAFL</td>
</tr>
<tr>
<td>Jacksonville</td>
<td>JAFL</td>
</tr>
<tr>
<td>Key West</td>
<td>KEFL</td>
</tr>
<tr>
<td>Lake City</td>
<td>LKFL</td>
</tr>
<tr>
<td>Miami</td>
<td>MIFL</td>
</tr>
<tr>
<td>RATE CENTER</td>
<td>CODE</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>------</td>
</tr>
<tr>
<td><strong>Florida (Continued)</strong></td>
<td></td>
</tr>
<tr>
<td>Ocala</td>
<td>OCFL</td>
</tr>
<tr>
<td>Orlando</td>
<td>ORFL</td>
</tr>
<tr>
<td>Panama City</td>
<td>PAFL</td>
</tr>
<tr>
<td>Pensacola</td>
<td>PEFL</td>
</tr>
<tr>
<td>St. Petersburg (Tampa)</td>
<td>TMFL</td>
</tr>
<tr>
<td>Sarasota</td>
<td>SAFL</td>
</tr>
<tr>
<td>Tallahassee</td>
<td>TAFL</td>
</tr>
<tr>
<td>Tampa</td>
<td>TMFL</td>
</tr>
<tr>
<td>West Palm Beach</td>
<td>WEFL</td>
</tr>
<tr>
<td>Winter Garden</td>
<td>WIFL</td>
</tr>
<tr>
<td>Winter Haven (Lakeland)</td>
<td>LAFL</td>
</tr>
<tr>
<td><strong>Georgia</strong></td>
<td></td>
</tr>
<tr>
<td>Albany</td>
<td>ALGA</td>
</tr>
<tr>
<td>Atlanta</td>
<td>ATGA</td>
</tr>
<tr>
<td>Augusta</td>
<td>AUGA</td>
</tr>
<tr>
<td>Brunswick</td>
<td>BRGA</td>
</tr>
<tr>
<td>Columbus</td>
<td>COGA</td>
</tr>
<tr>
<td>Conyers</td>
<td>CNGA</td>
</tr>
<tr>
<td>Dublin</td>
<td>DUGA</td>
</tr>
<tr>
<td>Fitzgerald</td>
<td>FIGA</td>
</tr>
<tr>
<td>Macon</td>
<td>MAGA</td>
</tr>
<tr>
<td>Rome</td>
<td>ROGA</td>
</tr>
<tr>
<td>Savannah</td>
<td>SAGA</td>
</tr>
<tr>
<td>Thomasville</td>
<td>THGA</td>
</tr>
<tr>
<td>Waycross</td>
<td>WAGA</td>
</tr>
<tr>
<td><strong>Idaho</strong></td>
<td></td>
</tr>
<tr>
<td>Boise</td>
<td>BOID</td>
</tr>
<tr>
<td>Pocatello</td>
<td>POID</td>
</tr>
<tr>
<td>Twin Falls</td>
<td>TWID</td>
</tr>
<tr>
<td><strong>Illinois</strong></td>
<td></td>
</tr>
<tr>
<td>Centralia</td>
<td>CEIL</td>
</tr>
<tr>
<td>Champaign-Urbana</td>
<td>CHIL</td>
</tr>
<tr>
<td>Chicago</td>
<td>CIL</td>
</tr>
<tr>
<td>Collinsville</td>
<td>COIL</td>
</tr>
<tr>
<td>De Kalb</td>
<td>DKIL</td>
</tr>
<tr>
<td>Hinsdale</td>
<td>HIIL</td>
</tr>
<tr>
<td>Joliet</td>
<td>JOIL</td>
</tr>
<tr>
<td>Marion</td>
<td>MAIL</td>
</tr>
<tr>
<td>Mattoon</td>
<td>MTIL</td>
</tr>
<tr>
<td>Newark</td>
<td>NEIL</td>
</tr>
<tr>
<td>Northbrook</td>
<td>NOIL</td>
</tr>
<tr>
<td>Peoria</td>
<td>PEIL</td>
</tr>
<tr>
<td>Rockford</td>
<td>ROIL</td>
</tr>
<tr>
<td>Rock Island (Davenport)</td>
<td>DAIL</td>
</tr>
<tr>
<td>Springfield</td>
<td>SPIL</td>
</tr>
<tr>
<td>Woodstock</td>
<td>WOIL</td>
</tr>
<tr>
<td>RATE CENTER</td>
<td>CODE</td>
</tr>
<tr>
<td>-------------</td>
<td>------</td>
</tr>
<tr>
<td><strong>Indiana</strong></td>
<td></td>
</tr>
<tr>
<td>Bloomington</td>
<td>BLIN</td>
</tr>
<tr>
<td>Evansville</td>
<td>EVIN</td>
</tr>
<tr>
<td>Fort Wayne</td>
<td>FOIN</td>
</tr>
<tr>
<td>Indianapolis</td>
<td>ININ</td>
</tr>
<tr>
<td>Muncie</td>
<td>MUIN</td>
</tr>
<tr>
<td>New Albany</td>
<td>NEIN</td>
</tr>
<tr>
<td>South Bend</td>
<td>SOIN</td>
</tr>
<tr>
<td>Terre Haute</td>
<td>TEIN</td>
</tr>
<tr>
<td><strong>Iowa</strong></td>
<td></td>
</tr>
<tr>
<td>Boone</td>
<td>BOIA</td>
</tr>
<tr>
<td>Burlington</td>
<td>BUIA</td>
</tr>
<tr>
<td>Cedar Rapids</td>
<td>CEIA</td>
</tr>
<tr>
<td>Davenport</td>
<td>DAIA</td>
</tr>
<tr>
<td>Dubuque</td>
<td>DUIA</td>
</tr>
<tr>
<td>Iowa City</td>
<td>IOIW</td>
</tr>
<tr>
<td>Sioux City</td>
<td>SINE</td>
</tr>
<tr>
<td>Waterloo</td>
<td>WAIA</td>
</tr>
<tr>
<td><strong>Kansas</strong></td>
<td></td>
</tr>
<tr>
<td>Dodge City</td>
<td>DOKS</td>
</tr>
<tr>
<td>Hutchinson</td>
<td>HUKS</td>
</tr>
<tr>
<td>Kansas City</td>
<td>KAMO</td>
</tr>
<tr>
<td>Manhattan</td>
<td>MAKS</td>
</tr>
<tr>
<td>Salina</td>
<td>SAKS</td>
</tr>
<tr>
<td>Topeka</td>
<td>TOKS</td>
</tr>
<tr>
<td>Wichita</td>
<td>WIKS</td>
</tr>
<tr>
<td><strong>Kentucky</strong></td>
<td></td>
</tr>
<tr>
<td>Danville</td>
<td>DAKY</td>
</tr>
<tr>
<td>Frankfort</td>
<td>FRKY</td>
</tr>
<tr>
<td>Lousiville</td>
<td>LOKY</td>
</tr>
<tr>
<td>Madisonville</td>
<td>MAKY</td>
</tr>
<tr>
<td>Paducah</td>
<td>PAKY</td>
</tr>
<tr>
<td>Winchester</td>
<td>WIKY</td>
</tr>
<tr>
<td><strong>Louisiana</strong></td>
<td></td>
</tr>
<tr>
<td>Alexandria</td>
<td>ALLA</td>
</tr>
<tr>
<td>Baton Rouge</td>
<td>BALA</td>
</tr>
<tr>
<td>Lafayette</td>
<td>LALA</td>
</tr>
<tr>
<td>Lake Charles</td>
<td>LKLA</td>
</tr>
<tr>
<td>Monroe</td>
<td>MOLA</td>
</tr>
<tr>
<td>New Orleans</td>
<td>NELA</td>
</tr>
<tr>
<td>Shreveport</td>
<td>SHLA</td>
</tr>
<tr>
<td><strong>Maine</strong></td>
<td></td>
</tr>
<tr>
<td>Augusta</td>
<td>AUME</td>
</tr>
<tr>
<td>Lewiston</td>
<td>LEME</td>
</tr>
<tr>
<td>Portland</td>
<td>POME</td>
</tr>
<tr>
<td>RATE CENTER</td>
<td>CODE</td>
</tr>
<tr>
<td>-----------------------</td>
<td>--------</td>
</tr>
<tr>
<td>Maryland</td>
<td></td>
</tr>
<tr>
<td>Baltimore</td>
<td>BAMD</td>
</tr>
<tr>
<td>Washington</td>
<td>WADC</td>
</tr>
<tr>
<td>Massachusetts</td>
<td></td>
</tr>
<tr>
<td>Boston</td>
<td>BOMA</td>
</tr>
<tr>
<td>Brockton</td>
<td>BRMA</td>
</tr>
<tr>
<td>Cambridge</td>
<td>CAMA</td>
</tr>
<tr>
<td>Fall River</td>
<td>FAMA</td>
</tr>
<tr>
<td>Framingham</td>
<td>FRMA</td>
</tr>
<tr>
<td>Lawrence</td>
<td>LAMA</td>
</tr>
<tr>
<td>Springfield</td>
<td>SPCT</td>
</tr>
<tr>
<td>Worcester</td>
<td>WOMA</td>
</tr>
<tr>
<td>Michigan</td>
<td></td>
</tr>
<tr>
<td>Detroit</td>
<td>DEMI</td>
</tr>
<tr>
<td>Flint</td>
<td>FLMI</td>
</tr>
<tr>
<td>Grand Rapids</td>
<td>GRMI</td>
</tr>
<tr>
<td>Houghton</td>
<td>HOMI</td>
</tr>
<tr>
<td>Iron Mountain</td>
<td>IRMI</td>
</tr>
<tr>
<td>Jackson</td>
<td>JAMI</td>
</tr>
<tr>
<td>Kalamazoo</td>
<td>KAMI</td>
</tr>
<tr>
<td>Lansing</td>
<td>LAMI</td>
</tr>
<tr>
<td>Petoskey</td>
<td>PEMI</td>
</tr>
<tr>
<td>Plymouth</td>
<td>PLMI</td>
</tr>
<tr>
<td>Pontiac</td>
<td>POMI</td>
</tr>
<tr>
<td>Saginaw</td>
<td>SAMI</td>
</tr>
<tr>
<td>Sault Ste. Marie</td>
<td>SUMI</td>
</tr>
<tr>
<td>Traverse City</td>
<td>TRMI</td>
</tr>
<tr>
<td>Minnesota</td>
<td></td>
</tr>
<tr>
<td>Duluth</td>
<td>DUMN</td>
</tr>
<tr>
<td>Minneapolis</td>
<td>MIMN</td>
</tr>
<tr>
<td>St. Cloud</td>
<td>STMN</td>
</tr>
<tr>
<td>St. Paul (Minneapolis)</td>
<td>MIMN</td>
</tr>
<tr>
<td>Virginia</td>
<td>VIMN</td>
</tr>
<tr>
<td>Wadena</td>
<td>WAMN</td>
</tr>
<tr>
<td>Willmar</td>
<td>WIMN</td>
</tr>
<tr>
<td>Mississippi</td>
<td></td>
</tr>
<tr>
<td>Biloxi</td>
<td>BIMS</td>
</tr>
<tr>
<td>Columbus</td>
<td>COMS</td>
</tr>
<tr>
<td>Greenville</td>
<td>GRMS</td>
</tr>
<tr>
<td>Greenwood</td>
<td>GEMS</td>
</tr>
<tr>
<td>Gulfport (Biloxi)</td>
<td>BIMS</td>
</tr>
<tr>
<td>Hattiesburg</td>
<td>HAMS</td>
</tr>
<tr>
<td>Jackson</td>
<td>JAMS</td>
</tr>
<tr>
<td>Laurel</td>
<td>LAMS</td>
</tr>
<tr>
<td>McComb</td>
<td>MCMS</td>
</tr>
<tr>
<td>Meridian</td>
<td>NEMS</td>
</tr>
<tr>
<td>Tupelo</td>
<td>TUMS</td>
</tr>
<tr>
<td>RATE CENTER</td>
<td>CODE</td>
</tr>
<tr>
<td>----------------</td>
<td>-------</td>
</tr>
<tr>
<td>Missouri</td>
<td></td>
</tr>
<tr>
<td>Cape Girardeau</td>
<td></td>
</tr>
<tr>
<td>Joplin</td>
<td>JOMO</td>
</tr>
<tr>
<td>Kansas City</td>
<td>KAMO</td>
</tr>
<tr>
<td>St. Joseph</td>
<td>STMO</td>
</tr>
<tr>
<td>St. Louis</td>
<td>SLMO</td>
</tr>
<tr>
<td>Sikeston</td>
<td>SIMO</td>
</tr>
<tr>
<td>Springfield</td>
<td>SPMO</td>
</tr>
<tr>
<td>Montana</td>
<td></td>
</tr>
<tr>
<td>Billings</td>
<td>BIMT</td>
</tr>
<tr>
<td>Glendive</td>
<td>GLMT</td>
</tr>
<tr>
<td>Helena</td>
<td>HEMT</td>
</tr>
<tr>
<td>Missoula</td>
<td>MIMT</td>
</tr>
<tr>
<td>Nebraska</td>
<td></td>
</tr>
<tr>
<td>Grand Island</td>
<td>GRNE</td>
</tr>
<tr>
<td>Omaha</td>
<td>OMNE</td>
</tr>
<tr>
<td>Sidney</td>
<td>SDNE</td>
</tr>
<tr>
<td>Nevada</td>
<td></td>
</tr>
<tr>
<td>Carson City</td>
<td>CANV</td>
</tr>
<tr>
<td>Las Vegas</td>
<td>LANV</td>
</tr>
<tr>
<td>Reno</td>
<td>RENV</td>
</tr>
<tr>
<td>New Hampshire</td>
<td></td>
</tr>
<tr>
<td>Concord</td>
<td>CONH</td>
</tr>
<tr>
<td>Dover (Portsmouth)</td>
<td>POHN</td>
</tr>
<tr>
<td>Manchester</td>
<td>MANH</td>
</tr>
<tr>
<td>Nashua</td>
<td>NANH</td>
</tr>
<tr>
<td>New Jersey</td>
<td></td>
</tr>
<tr>
<td>Atlantic City</td>
<td>ATNJ</td>
</tr>
<tr>
<td>Camden</td>
<td>CANJ</td>
</tr>
<tr>
<td>Hackensack</td>
<td>HANJ</td>
</tr>
<tr>
<td>Morristown</td>
<td>MONJ</td>
</tr>
<tr>
<td>Newark</td>
<td>NENJ</td>
</tr>
<tr>
<td>New Brunswick</td>
<td>NENJ</td>
</tr>
<tr>
<td>Trenton</td>
<td>TRNJ</td>
</tr>
<tr>
<td>New Mexico</td>
<td></td>
</tr>
<tr>
<td>Albuquerque</td>
<td>ALNM</td>
</tr>
<tr>
<td>Las Cruces</td>
<td>LANM</td>
</tr>
<tr>
<td>Roswell</td>
<td>RONM</td>
</tr>
<tr>
<td>Santa Fe</td>
<td>SANM</td>
</tr>
<tr>
<td>New York</td>
<td></td>
</tr>
<tr>
<td>Albany</td>
<td>ALNY</td>
</tr>
<tr>
<td>Binghamton</td>
<td>BINY</td>
</tr>
<tr>
<td>Buffalo</td>
<td>BUNY</td>
</tr>
<tr>
<td>RATE CENTER</td>
<td>CODE</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>--------</td>
</tr>
<tr>
<td>New York (Continued)</td>
<td></td>
</tr>
<tr>
<td>Huntington</td>
<td>HUNY</td>
</tr>
<tr>
<td>Nassau</td>
<td>NANY</td>
</tr>
<tr>
<td>New York City</td>
<td>NENY</td>
</tr>
<tr>
<td>Potsdam</td>
<td>PTNY</td>
</tr>
<tr>
<td>Poughkeepsie</td>
<td>PONY</td>
</tr>
<tr>
<td>Rochester</td>
<td>RONY</td>
</tr>
<tr>
<td>Syracuse</td>
<td>SYNY</td>
</tr>
<tr>
<td>Troy (Albany)</td>
<td>ALNY</td>
</tr>
<tr>
<td>Westchester</td>
<td>WENY</td>
</tr>
<tr>
<td>North Carolina</td>
<td></td>
</tr>
<tr>
<td>Asheville</td>
<td>ASNC</td>
</tr>
<tr>
<td>Charlotte</td>
<td>CHNC</td>
</tr>
<tr>
<td>Fayetteville</td>
<td>FANC</td>
</tr>
<tr>
<td>Gastonia (Charlotte)</td>
<td>CHNC</td>
</tr>
<tr>
<td>Greensboro</td>
<td>GRNC</td>
</tr>
<tr>
<td>Greenville</td>
<td>GENC</td>
</tr>
<tr>
<td>Laurinburg</td>
<td>LANC</td>
</tr>
<tr>
<td>New Bern</td>
<td>NENC</td>
</tr>
<tr>
<td>Raleigh</td>
<td>RANC</td>
</tr>
<tr>
<td>Rocky Mount</td>
<td>RONC</td>
</tr>
<tr>
<td>Wilmington</td>
<td>WINC</td>
</tr>
<tr>
<td>Winston-Salem (Greensboro)</td>
<td>GRNC</td>
</tr>
<tr>
<td>North Dakota</td>
<td></td>
</tr>
<tr>
<td>Bismark</td>
<td>BIND</td>
</tr>
<tr>
<td>Casselton</td>
<td>CAND</td>
</tr>
<tr>
<td>Dickinson</td>
<td>DIND</td>
</tr>
<tr>
<td>Fargo</td>
<td>FAND</td>
</tr>
<tr>
<td>Grand Forks</td>
<td>GRND</td>
</tr>
<tr>
<td>Ohio</td>
<td></td>
</tr>
<tr>
<td>Akron</td>
<td>AKOH</td>
</tr>
<tr>
<td>Canton</td>
<td>CAOH</td>
</tr>
<tr>
<td>Cincinnati</td>
<td>CIOH</td>
</tr>
<tr>
<td>Cleveland</td>
<td>CLOH</td>
</tr>
<tr>
<td>Columbus</td>
<td>COOH</td>
</tr>
<tr>
<td>Dayton</td>
<td>DAOH</td>
</tr>
<tr>
<td>Findley</td>
<td>FIOH</td>
</tr>
<tr>
<td>Mansfield</td>
<td>MAOH</td>
</tr>
<tr>
<td>Toledo</td>
<td>TOOH</td>
</tr>
<tr>
<td>Youngstown</td>
<td>YOOH</td>
</tr>
<tr>
<td>Oklahoma</td>
<td></td>
</tr>
<tr>
<td>Enid</td>
<td>ENOK</td>
</tr>
<tr>
<td>Lawton</td>
<td>LAOK</td>
</tr>
<tr>
<td>Muskogee</td>
<td>MUOK</td>
</tr>
<tr>
<td>Oklahoma City</td>
<td>OKOK</td>
</tr>
<tr>
<td>Tulsa</td>
<td>TUOK</td>
</tr>
</tbody>
</table>

C-10
<table>
<thead>
<tr>
<th>Rate Center</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oregon</td>
<td></td>
</tr>
<tr>
<td>Medford</td>
<td>MEOR</td>
</tr>
<tr>
<td>Pendleton</td>
<td>PEOR</td>
</tr>
<tr>
<td>Portland</td>
<td>POOR</td>
</tr>
<tr>
<td>Pennsylvania</td>
<td></td>
</tr>
<tr>
<td>Allentown</td>
<td>ALPA</td>
</tr>
<tr>
<td>Altoona</td>
<td>ATPA</td>
</tr>
<tr>
<td>Harrisburg</td>
<td>HAPA</td>
</tr>
<tr>
<td>Philadelphia</td>
<td>PHPA</td>
</tr>
<tr>
<td>Pittsburgh</td>
<td>PIPA</td>
</tr>
<tr>
<td>Pottsville</td>
<td>POPA</td>
</tr>
<tr>
<td>Reading</td>
<td>REPA</td>
</tr>
<tr>
<td>Scranton</td>
<td>SCPA</td>
</tr>
<tr>
<td>State College</td>
<td>STPD</td>
</tr>
<tr>
<td>Williamsport</td>
<td>WIPA</td>
</tr>
<tr>
<td>Rhode Island</td>
<td></td>
</tr>
<tr>
<td>Providence</td>
<td>PRRI</td>
</tr>
<tr>
<td>South Carolina</td>
<td></td>
</tr>
<tr>
<td>Charleston</td>
<td>CHSC</td>
</tr>
<tr>
<td>Columbia</td>
<td>COSC</td>
</tr>
<tr>
<td>Florence</td>
<td>FOSC</td>
</tr>
<tr>
<td>Greenville</td>
<td>GRSC</td>
</tr>
<tr>
<td>Orangeburg</td>
<td>ORSC</td>
</tr>
<tr>
<td>Spartanburg (Greenville)</td>
<td>GRSC</td>
</tr>
<tr>
<td>South Dakota</td>
<td></td>
</tr>
<tr>
<td>Aberdeen</td>
<td>ABSD</td>
</tr>
<tr>
<td>Huron</td>
<td>HUSD</td>
</tr>
<tr>
<td>Sioux Falls</td>
<td>SISD</td>
</tr>
<tr>
<td>Tennessee</td>
<td></td>
</tr>
<tr>
<td>Chattanooga</td>
<td>CHTN</td>
</tr>
<tr>
<td>Clarksville</td>
<td>CLTN</td>
</tr>
<tr>
<td>Jackson</td>
<td>JATN</td>
</tr>
<tr>
<td>Johnson City</td>
<td>JOTN</td>
</tr>
<tr>
<td>Kingsport (Johnson City)</td>
<td>JOTN</td>
</tr>
<tr>
<td>Knoxville</td>
<td>KNTN</td>
</tr>
<tr>
<td>Memphis</td>
<td>METN</td>
</tr>
<tr>
<td>Morristown</td>
<td>MOTN</td>
</tr>
<tr>
<td>Nashville</td>
<td>NATN</td>
</tr>
<tr>
<td>Texas</td>
<td></td>
</tr>
<tr>
<td>Abilene</td>
<td>ABTX</td>
</tr>
<tr>
<td>Amarillo</td>
<td>AMTX</td>
</tr>
<tr>
<td>Austin</td>
<td>AUTX</td>
</tr>
<tr>
<td>Beaumont</td>
<td>BETX</td>
</tr>
<tr>
<td>Corpus Christi</td>
<td>COTX</td>
</tr>
<tr>
<td>Dallas</td>
<td>DATX</td>
</tr>
<tr>
<td>RATE CENTER</td>
<td>CODE</td>
</tr>
<tr>
<td>------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>Texas (Continued)</td>
<td></td>
</tr>
<tr>
<td>El Paso</td>
<td>ELTX</td>
</tr>
<tr>
<td>Fort Worth (Dallas)</td>
<td>DATX</td>
</tr>
<tr>
<td>Freeport</td>
<td>FRTX</td>
</tr>
<tr>
<td>Harlingen (Brownsville)</td>
<td>BRTX</td>
</tr>
<tr>
<td>Houston</td>
<td>HOTX</td>
</tr>
<tr>
<td>Laredo</td>
<td>LATX</td>
</tr>
<tr>
<td>Longview</td>
<td>LOTX</td>
</tr>
<tr>
<td>Lubbock</td>
<td>LUTX</td>
</tr>
<tr>
<td>Midland</td>
<td>MITX</td>
</tr>
<tr>
<td>San Angelo</td>
<td>SATX</td>
</tr>
<tr>
<td>San Antonio</td>
<td>SNTX</td>
</tr>
<tr>
<td>Sweetwater</td>
<td>SWTX</td>
</tr>
<tr>
<td>Waco</td>
<td>WATX</td>
</tr>
<tr>
<td>Utah</td>
<td></td>
</tr>
<tr>
<td>Logan</td>
<td>LOUT</td>
</tr>
<tr>
<td>Ogden (Salt Lake City)</td>
<td>SAUT</td>
</tr>
<tr>
<td>Provo</td>
<td>PRUT</td>
</tr>
<tr>
<td>Salt Lake City</td>
<td>SAUT</td>
</tr>
<tr>
<td>Vermont</td>
<td></td>
</tr>
<tr>
<td>Burlington</td>
<td>BUVT</td>
</tr>
<tr>
<td>White River Junction</td>
<td>WHVT</td>
</tr>
<tr>
<td>Virginia</td>
<td></td>
</tr>
<tr>
<td>Blacksburg</td>
<td>BLVA</td>
</tr>
<tr>
<td>Leesburg</td>
<td>LEVA</td>
</tr>
<tr>
<td>Lynchburg</td>
<td>LYVA</td>
</tr>
<tr>
<td>Newport News</td>
<td>NEVA</td>
</tr>
<tr>
<td>Norfolk</td>
<td>NOVA</td>
</tr>
<tr>
<td>Petersburg</td>
<td>PEVA</td>
</tr>
<tr>
<td>Richmond</td>
<td>RIVA</td>
</tr>
<tr>
<td>Roanoke</td>
<td>ROVA</td>
</tr>
<tr>
<td>Washington</td>
<td>WADC</td>
</tr>
<tr>
<td>Washington</td>
<td></td>
</tr>
<tr>
<td>Billingham</td>
<td>BEMW</td>
</tr>
<tr>
<td>Kennewick (Richland)</td>
<td>RIWA</td>
</tr>
<tr>
<td>North Bend</td>
<td>NOWA</td>
</tr>
<tr>
<td>Seattle</td>
<td>SEWA</td>
</tr>
<tr>
<td>Spokane</td>
<td>SPWA</td>
</tr>
<tr>
<td>Yakima</td>
<td>YAWA</td>
</tr>
<tr>
<td>West Virginia</td>
<td></td>
</tr>
<tr>
<td>Beckley</td>
<td>BEWV</td>
</tr>
<tr>
<td>Charleston</td>
<td>CHNV</td>
</tr>
<tr>
<td>Clarksburg</td>
<td>CLWW</td>
</tr>
<tr>
<td>Fairmont</td>
<td>FAWV</td>
</tr>
<tr>
<td>Huntington</td>
<td>HUNV</td>
</tr>
<tr>
<td>Morgantown</td>
<td>MOWV</td>
</tr>
</tbody>
</table>

C-12
<table>
<thead>
<tr>
<th>RATE CENTER</th>
<th>CODE</th>
</tr>
</thead>
<tbody>
<tr>
<td>West Virginia (Continued)</td>
<td></td>
</tr>
<tr>
<td>Parkensburg</td>
<td>PAWV</td>
</tr>
<tr>
<td>Wheeling</td>
<td>WHWV</td>
</tr>
<tr>
<td>Wisconsin</td>
<td></td>
</tr>
<tr>
<td>Appleton</td>
<td>APWI</td>
</tr>
<tr>
<td>Dodgeville</td>
<td>DOWI</td>
</tr>
<tr>
<td>Eau Claire</td>
<td>EAWI</td>
</tr>
<tr>
<td>Green Bay</td>
<td>GRWI</td>
</tr>
<tr>
<td>La Crosse</td>
<td>LAWI</td>
</tr>
<tr>
<td>Madison</td>
<td>MAWI</td>
</tr>
<tr>
<td>Milwaukee</td>
<td>MIWI</td>
</tr>
<tr>
<td>Racine</td>
<td>RAWI</td>
</tr>
<tr>
<td>Stevens Port</td>
<td>STWI</td>
</tr>
<tr>
<td>Wyoming</td>
<td></td>
</tr>
<tr>
<td>Casper</td>
<td>CAWY</td>
</tr>
<tr>
<td>Cheyanne</td>
<td>CHMY</td>
</tr>
</tbody>
</table>
This appendix defines the Western Union Telegraph Company tariff as defined in F.C.C. No. 261 (effective May 11, 1982). This tariff is referred to as the prestored WU tariff in the CNDC model.

Channel Charges (per month per channel)

<table>
<thead>
<tr>
<th>Category I</th>
<th>Long Haul</th>
<th>$925.00</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category II</td>
<td>Medium Haul</td>
<td>$695.00</td>
</tr>
<tr>
<td>Category III</td>
<td>Short Haul</td>
<td>$580.00</td>
</tr>
</tbody>
</table>

The monthly charge per channel specified above as for the routes listed below. By each route pair is the four character code by which the pair is known in the CNDC model. These codes are the SMSA codes of the associated rate center.

Category I - LONG HAUL

<table>
<thead>
<tr>
<th>Route Pair</th>
<th>Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>Los Angeles - Atlanta</td>
<td>LOCA - ATGA</td>
</tr>
<tr>
<td>Los Angeles - Baltimore</td>
<td>LOCA - BAMD</td>
</tr>
<tr>
<td>Los Angeles - Boston</td>
<td>LOCA - BOMA</td>
</tr>
<tr>
<td>Los Angeles - Buffalo</td>
<td>LOCA - BUNY</td>
</tr>
<tr>
<td>Los Angeles - Cincinnati</td>
<td>LOCA - CIIOH</td>
</tr>
<tr>
<td>Los Angeles - Cleveland</td>
<td>LOCA - CLOH</td>
</tr>
<tr>
<td>Los Angeles - Columbus</td>
<td>LOCA - COOH</td>
</tr>
<tr>
<td>Los Angeles - Dayton</td>
<td>LOCA - DOAH</td>
</tr>
<tr>
<td>Los Angeles - Detroit</td>
<td>LOCA - DEMI</td>
</tr>
<tr>
<td>Los Angeles - New York</td>
<td>LOCA - NENY</td>
</tr>
<tr>
<td>Los Angeles - Philadelphia</td>
<td>LOCA - PHPA</td>
</tr>
<tr>
<td>Los Angeles - Pittsburgh</td>
<td>LOCA - PIPA</td>
</tr>
<tr>
<td>Los Angeles - Washington</td>
<td>LOCA - WADC</td>
</tr>
<tr>
<td>Los Angeles - Wilmington</td>
<td>LOCA - WIDE</td>
</tr>
<tr>
<td>San Francisco - Atlanta</td>
<td>SFAC - ATGA</td>
</tr>
<tr>
<td>San Francisco - Baltimore</td>
<td>SFCA - BAMD</td>
</tr>
<tr>
<td>San Francisco - Boston</td>
<td>SFCA - BOMA</td>
</tr>
<tr>
<td>San Francisco - Buffalo</td>
<td>SFCA - BUNY</td>
</tr>
<tr>
<td>San Francisco - Cincinnati</td>
<td>SFCA - CIIOH</td>
</tr>
<tr>
<td>San Francisco - Cleveland</td>
<td>SFCA - CLOH</td>
</tr>
<tr>
<td>San Francisco - Columbus</td>
<td>SFCA - COOH</td>
</tr>
<tr>
<td>Flight Route</td>
<td>Code</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>--------</td>
</tr>
<tr>
<td>San Francisco - Dayton</td>
<td>SFCA</td>
</tr>
<tr>
<td>San Francisco - Detroit</td>
<td>SFCA</td>
</tr>
<tr>
<td>San Francisco - New York</td>
<td>SFCA</td>
</tr>
<tr>
<td>San Francisco - Philadelphia</td>
<td>SFCA</td>
</tr>
<tr>
<td>San Francisco - Pittsburgh</td>
<td>SFCA</td>
</tr>
<tr>
<td>San Francisco - Washington</td>
<td>SFCA</td>
</tr>
<tr>
<td>San Francisco - Wilmington</td>
<td>SFCA</td>
</tr>
<tr>
<td>Seattle - Boston</td>
<td>SEWA</td>
</tr>
<tr>
<td>Seattle - Cleveland</td>
<td>SEWA</td>
</tr>
<tr>
<td>Seattle - Detroit</td>
<td>SEWA</td>
</tr>
<tr>
<td>Seattle - New York</td>
<td>SEWA</td>
</tr>
<tr>
<td>Seattle - Philadelphia</td>
<td>SEWA</td>
</tr>
<tr>
<td>Seattle - Pittsburgh</td>
<td>SEWA</td>
</tr>
<tr>
<td>Seattle - Washington</td>
<td>SEWA</td>
</tr>
<tr>
<td>Seattle - Detroit</td>
<td>SEWA</td>
</tr>
<tr>
<td>Seattle - Cleveland</td>
<td>SEWA</td>
</tr>
<tr>
<td>Seattle - New York</td>
<td>SEWA</td>
</tr>
<tr>
<td>Seattle - Philadelphia</td>
<td>SEWA</td>
</tr>
<tr>
<td>Seattle - Pittsburgh</td>
<td>SEWA</td>
</tr>
<tr>
<td>Seattle - Washington</td>
<td>SEWA</td>
</tr>
<tr>
<td>San Francisco - Chicago</td>
<td>SFCA</td>
</tr>
<tr>
<td>Kansas City - Omaha</td>
<td>KAMO</td>
</tr>
<tr>
<td>Kansas City - Los Angeles</td>
<td>KAMO</td>
</tr>
<tr>
<td>Kansas City - New York</td>
<td>KAMO</td>
</tr>
<tr>
<td>Kansas City - San Francisco</td>
<td>KAMO</td>
</tr>
<tr>
<td>Los Angeles - Chicago</td>
<td>LOCA</td>
</tr>
<tr>
<td>* Los Angeles - Bridgeton, MO</td>
<td>LOCA</td>
</tr>
<tr>
<td>Los Angeles - Indianapolis</td>
<td>LOCA</td>
</tr>
<tr>
<td>Los Angeles - Milwaukee</td>
<td>LOCA</td>
</tr>
<tr>
<td>Los Angeles - Minneapolis</td>
<td>LOCA</td>
</tr>
<tr>
<td>Los Angeles - St Louis</td>
<td>LOCA</td>
</tr>
<tr>
<td>Minneapolis - Boston</td>
<td>MIMN</td>
</tr>
<tr>
<td>San Francisco - Chicago</td>
<td>SFCA</td>
</tr>
<tr>
<td>San Francisco - Indianapolis</td>
<td>SFCA</td>
</tr>
<tr>
<td>San Francisco - Milwaukee</td>
<td>SFCA</td>
</tr>
</tbody>
</table>

* Bridgeton, MO will be viewed as St. Louis, MO.
Category II - (Continued)

San Francisco - Minneapolis  SFCA - MIMN  
San Francisco - St Louis  SFCA - SLMO  
Seattle - Chicago  SEWA - CIIL  
Seattle - Dallas/Ft Worth  SEWA - DATX  
Seattle - Kansas City  SEWA - KAMO  
Seattle - Milwaukee  SEWA - MIWI  
Seattle - Minneapolis  SEWA - MIMN  
Seattle - St Louis  SEWA - SLMO  

Category III - Short Haul

Atlanta - Baltimore  ATGA - BAMD  
Atlanta - Boston  ATGA - BOMA  
Atlanta - Chicago  ATGA - CIIL  
Atlanta - Cleveland  ATGA - CLOH  
Atlanta - Dallas/Ft Worth  ATGA - DATX  
Atlanta - Detroit  ATGA - DEMI  
Atlanta - Houston  ATGA - HOTX  
Atlanta - Indianapolis  ATGA - ININ  
Atlanta - Kansas City  ATGA - KAMO  
Atlanta - Milwaukee  ATGA - MIWI  
Atlanta - Minneapolis  ATGA - MIMN  
Atlanta - Philadelphia  ATGA - PHPA  
Atlanta - New York  ATGA - NENY  
Atlanta - Washington  ATGA - WADC  
Atlanta - Wilmington  ATGA - WIDE  
Boston - Chicago  BOMA - CIIL  
Boston - Cincinnati  BOMA - CIOH  
Boston - Columbus  BOMA - COOH  
Boston - Dayton  BOMA - DAOH  
Boston - Indianapolis  BOMA - ININ  
Boston - Milwaukee  BOMA - MIWI  
Boston - St Louis  BOMA - SLMO  
Chicago - Baltimore  CIIL - BAMD  
Chicago - Dallas/Ft Worth  CIIL - DATX  
Chicago - Houston  CIIL - HOTX  
Chicago - New York  CIIL - NENY  
Chicago - Philadelphia  CIIL - PHPA  
Chicago - Washington  CIIL - WADC  
Chicago - Wilmington  CIIL - WIDE  
Dallas/Ft Worth - Cincinnati  DATX - CIOH  
Dallas/Ft Worth - Cleveland  DATX - CLOH  
Dallas/Ft Worth - Columbus  DATX - COOH  
Dallas/Ft Worth - Dayton  DATX - DAOH  
Dallas/Ft Worth - Detroit  DATX - DEMI  
Dallas/Ft Worth - Indianapolis  DATX - ININ  
Dallas/Ft Worth - Milwaukee  DATX - MIWI  
Dallas/Ft Worth - Minneapolis  DATX - MIMN  
Dallas/Ft Worth - Pittsburgh  DATX - PIPA  
Dallas/Ft Worth - St Louis  DATX - SLMO  

D-3
Category III - (Continued)

Houston - Cincinnati
Houston - Indianapolis
Houston - Milwaukee
Houston - Minneapolis
Houston - St Louis
Milwaukee - Baltimore
Milwaukee - New York
Milwaukee - Philadelphia
Milwaukee - Washington
New York - Columbus
New York - Dayton
New York - Indianapolis
New York - Minneapolis
Philadelphia - Indianapolis
Philadelphia - Kansas City
St Louis - Baltimore
St Louis - New York
St Louis - Washington
St Louis - Wilmington
Seattle - Los Angeles
Seattle - San Francisco
Washington - Indianapolis
Washington - Minneapolis
HOTX - CIOH
HOTX - ININ
HOTX - MIWI
HOTX - MIMN
HOTX - SLMO
MIWI - BAMD
MIWI - NENY
MIWI - PHPA
MIWI - WADC
NENY - COOH
NENY - DAOH
NENY - ININ
NENY - MINM
PHPA - ININ
PHPA - KAMO
SLMO - BAMD
SLMO - NENY
SLMO - WADC
SLMO - WIDE
SEWA - LOCA
SEWA - SFCA
WADC - ININ
WADC - MIMN
Appendix E - Prestored SBS Tariff

This appendix defines the Satellite Business Systems tariff as set forth in F.C.C. No. 2 (effective date October 1, 1983). This tariff is referred to as the prestored SBS tariff in the CNDC model. All charges specified are monthly recurring charges.

<table>
<thead>
<tr>
<th>Network Access Center (each)</th>
<th>NAC</th>
<th>$17,850.00</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAC Supplemental Capacity Units (each)</td>
<td>SCU</td>
<td>$5,700.00</td>
</tr>
<tr>
<td>Full Time Transmission Units</td>
<td>FTU</td>
<td>$2,550.00</td>
</tr>
<tr>
<td>Minimum CAU Charges</td>
<td>CAU</td>
<td>$17,850.00</td>
</tr>
</tbody>
</table>

Connection Arrangement Units
(monthly at each NAC)

<table>
<thead>
<tr>
<th>NUMBER OF CAUs AT EACH NAC</th>
<th>MONTHLY RECURRING CHARGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 CAU - 150 CAUs</td>
<td>$95 per CAU</td>
</tr>
<tr>
<td>151 CAUs - 300 CAUs</td>
<td>$14,250.00 plus $90.00 per CAU in excess of 150</td>
</tr>
<tr>
<td>More than 300 CAUs</td>
<td>$27,750.00 plus $65.00 per CAU in excess of 300</td>
</tr>
</tbody>
</table>
Appendix F - Prestored Network

This appendix contains the 316 SMSAs that make up the prestored network of the CNDC model. Each SMSA is defined by the four character code by which it is known to the model, the vertical and horizontal coordinates, and the SMSA descriptor.

<table>
<thead>
<tr>
<th>Code</th>
<th>Vertical</th>
<th>Horizontal</th>
<th>City, State, Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABTX</td>
<td>8698</td>
<td>4513</td>
<td>ABILENE TX</td>
</tr>
<tr>
<td>AKOH</td>
<td>5637</td>
<td>2472</td>
<td>AKRON OH</td>
</tr>
<tr>
<td>ALGA</td>
<td>7649</td>
<td>1817</td>
<td>ALBANY GA</td>
</tr>
<tr>
<td>ALLA</td>
<td>8409</td>
<td>3168</td>
<td>ALEXANDRIA LA</td>
</tr>
<tr>
<td>ALNM</td>
<td>8549</td>
<td>5887</td>
<td>ALBUQUERQUE NM</td>
</tr>
<tr>
<td>ALNY</td>
<td>4639</td>
<td>1629</td>
<td>ALBANY-SCHENECTADY-TROY NY</td>
</tr>
<tr>
<td>ALPA</td>
<td>5166</td>
<td>1585</td>
<td>ALLENTOWN-BETHLEHEM-EASTON PA-NJ</td>
</tr>
<tr>
<td>AMTX</td>
<td>8266</td>
<td>5076</td>
<td>AMARILLO TX</td>
</tr>
<tr>
<td>ANAL</td>
<td>7406</td>
<td>2304</td>
<td>ANNISTON AL</td>
</tr>
<tr>
<td>ANCA</td>
<td>9250</td>
<td>7810</td>
<td>ANAHEIM-SANTA ANA-GARDEN GROVE CA</td>
</tr>
<tr>
<td>AWIN</td>
<td>673</td>
<td>2958</td>
<td>ANDERSON IN</td>
</tr>
<tr>
<td>ANMI</td>
<td>5602</td>
<td>2908</td>
<td>ANN ARBOR MI</td>
</tr>
<tr>
<td>ANSC</td>
<td>6961</td>
<td>1894</td>
<td>ANDERSON SC</td>
</tr>
<tr>
<td>APWI</td>
<td>5589</td>
<td>3776</td>
<td>APPLETON-OSHKOSH WI</td>
</tr>
<tr>
<td>ASCN</td>
<td>6749</td>
<td>2001</td>
<td>ASHEVILLE NC</td>
</tr>
<tr>
<td>ATGA</td>
<td>7260</td>
<td>2083</td>
<td>ATLANTA GA</td>
</tr>
<tr>
<td>ATGE</td>
<td>7130</td>
<td>1948</td>
<td>ATHENS GE</td>
</tr>
<tr>
<td>ATNJ</td>
<td>5284</td>
<td>1284</td>
<td>ATLANTIC CITY NJ</td>
</tr>
<tr>
<td>ATPA</td>
<td>5460</td>
<td>1972</td>
<td>Altoona PA</td>
</tr>
<tr>
<td>AUGA</td>
<td>7089</td>
<td>1674</td>
<td>AUGUSTA GA-SC</td>
</tr>
<tr>
<td>AUTX</td>
<td>9005</td>
<td>3996</td>
<td>AUSTIN TX</td>
</tr>
<tr>
<td>BACA</td>
<td>8947</td>
<td>8060</td>
<td>BAKERSFIELD CA</td>
</tr>
<tr>
<td>BALA</td>
<td>8476</td>
<td>2874</td>
<td>BATON ROUGE LA</td>
</tr>
<tr>
<td>BAMD</td>
<td>5510</td>
<td>1575</td>
<td>BALTIMORE MD</td>
</tr>
<tr>
<td>BAME</td>
<td>3777</td>
<td>1322</td>
<td>BANGOR ME</td>
</tr>
<tr>
<td>BAMI</td>
<td>5713</td>
<td>3124</td>
<td>BATTLE CREEK MI</td>
</tr>
<tr>
<td>BEMS</td>
<td>5850</td>
<td>3281</td>
<td>BENTON HARBOR MI</td>
</tr>
<tr>
<td>BETX</td>
<td>8777</td>
<td>3344</td>
<td>BEAUMONT-PORT ARTHUR-ORANGE TX</td>
</tr>
<tr>
<td>BEMA</td>
<td>6087</td>
<td>8933</td>
<td>BELLINGHAM WA</td>
</tr>
<tr>
<td>BIAL</td>
<td>7518</td>
<td>2446</td>
<td>BIRMINGHAM AL</td>
</tr>
<tr>
<td>BICT</td>
<td>4730</td>
<td>1394</td>
<td>BRISTOL CT</td>
</tr>
<tr>
<td>BIMS</td>
<td>8296</td>
<td>2481</td>
<td>BILOXI-GULFPORT MS</td>
</tr>
<tr>
<td>BIMT</td>
<td>6391</td>
<td>6790</td>
<td>BILLINGS MT</td>
</tr>
<tr>
<td>BIND</td>
<td>5840</td>
<td>5736</td>
<td>BISMARK ND</td>
</tr>
<tr>
<td>BINY</td>
<td>4943</td>
<td>1837</td>
<td>BINGHAMTON NY-PA</td>
</tr>
<tr>
<td>BLIL</td>
<td>6358</td>
<td>3483</td>
<td>BLOOMINGTON-NORMAL IL</td>
</tr>
<tr>
<td>BLIN</td>
<td>6417</td>
<td>2994</td>
<td>BLOOMINGTON IN</td>
</tr>
<tr>
<td>BROID</td>
<td>7096</td>
<td>7869</td>
<td>BOISE CITY ID</td>
</tr>
<tr>
<td>BOMA</td>
<td>4422</td>
<td>1249</td>
<td>BOSTON MA</td>
</tr>
<tr>
<td>BRCT</td>
<td>4841</td>
<td>1360</td>
<td>BRIDGEPORT CT</td>
</tr>
<tr>
<td>BRFL</td>
<td>8270</td>
<td>1116</td>
<td>BRANDENTON FL</td>
</tr>
</tbody>
</table>
BRMA 4465 1205 BROCKTON MA
BRTX 9820 3663 BROWNVILLE-HARLINGEN-SAN BENI TX
BRWA 6349 8940 BREMERTON WA
BUNC 6364 1588 BURLINGTON NC
BUNY 5075 2326 BUFFALO NY
BUYT 4270 1808 BURLINGTON VT
BYMI 5368 3085 BAY CITY MI
BYTX 8827 3788 BRYAN-COLLEGE STATION TX
CAOH 5676 2419 CANTON OH
CANY 6918 6297 CASPER WY
CEIA 6261 4021 CEDAR RAPIDS IA
CHCA 8057 8668 CHICO CA
CHIL 6371 3335 CHAMPAIGN-URBANA-RANTOL IL
CHNC 6657 1698 CHARLOTTE-GASTONIA NC
CHSC 7021 1281 CHARLESTON-NORTH CHARLESTON SC
CHTN 7098 2365 CHATTANOOGA TN-GA

CHVA 5919 1683 CHARLOTTESVILLE VA
CHNV 6152 2174 CHARLESTON WV
CIIL 5986 3425 CHICAGO IL
CIOH 6263 2679 CINCINNATI OH-KY
CLOH 5574 2543 CLEVELAND OH
CLTN 6988 2837 CLARKSVILLE-HOPKINSVILLE TN-KY
COCO 7556 2045 COLUMBUS GA-AL
COMO 6901 3841 COLUMBIA MO
COOH 5972 2555 COLUMBUS OH
COSC 6901 1589 COLUMBIA SC
COTX 9475 3739 CORPUS CHRISTI TX
CUMD 5650 1916 CUMBERLAND MD-WV
DACT 4829 1423 DANBURY CT
DAFL 7791 1032 DAYTONA BEACH FL
DAIA 6273 3817 DAVENPORT-ROCK ISLAND-MOLINE IA-IL
DAOH 6113 2705 DAYTON OH
DATX 8436 4034 DALLAS-FORT WORTH TX
DAYA 6270 1640 DANVILLE VA
DECO 7501 5899 DENVER-BOULDER CO
DEIA 6471 4275 DES MOINES IA
DEIL 6478 3413 DECATURE IL
DEM1 5536 2828 DETROIT MI
DUIA 6088 3925 DUBUQUE IA
DUMN 5352 4530 DULUTH-SUPERIOR MN-WI
EAWI 5698 4261 EAU CLAIRE WI
ELIN 5895 3168 ELKHART IN
ELNY 5029 1953 ELMIRA NY
ELTX 9231 5655 EL PASO TX
ENOK 7783 4505 ENID OK

ERPA 5321 2397 ERIE PA
EUOR 7128 8954 EUGENE-SPRINGFIELD OR
EVIN 6729 3019 EVANSVILLE IN-KY
FAAR 7600 3872 FAYETTEVILLE-SPRINGDALE AR
FAMA 4543 1170 FALL RIVER MA-RI
FANC 6501 1385 FAYETTEVILLE NC
FAND 5615 5182 FARGO-MOORHEAD ND-MN
FIMA 4459 1374 FITCHBURG-LEOMINSTER MA
FLAL 7344 2715 FLORENCE AL
FLMI 5461 2993 FLINT MI
FOAR 7752 3855 FORT SMITH AR-OK
FOCO 7331 5965 FORT COLLINS CO
FOFL 8282 557 FORT LAUDERDALE-HOLLYWOOD FL
FOIN 5942 2982 FORT WAYNE IN
FOSC 6744 1417 FORENC SC
FRCA 8669 8239 FRESNO CA
FRFL 8359 904 FORT MYERS FL
FTFL 8097 2097 FORT WALTON BEACH FL
GAAL 7355 2368 GADSDEN AL
GAFL 7838 1310 GAINESVILLE FL
GAIN 6017 3354 GARY-HAMMOND-EAST CHICAGO IN
GATX 8985 3397 GALVESTON-Texas CITY TX
GLNY 4515 1704 GLENS FALLS NY
GRCO 7345 5895 GREELEY CO
GRMI 5629 3261 GRAND RAPIDS MI
GRMT 6120 7281 GREAT FALLS MT
GRNC 6400 1638 GREENSBORO-Winston-Salem-High NC
GRND 5418 5297 GRAND FORKS ND-MN
GRSC 6250 1226 GREENVILLE-SPARTANBURG SC
GRWI 5512 3747 GREEN BAY WI
HACT 4687 1373 HARTFORD CT
HAMO 5555 1772 HAGERSTOWN MD
HAOH 6210 2718 HAMILTON-MIDDLETOWN OH
HAPA 5363 1733 HARRISBURG PA
HINC 6611 1833 HICKORY NC
HOTX 8938 3536 HOUSTON TX
HAUL 7267 2535 HUNTSVILLE AL
HUWV 6212 2299 HUNTINGTON-ASHLAND WV-KY
ININ 6272 2992 INDIANAPOLIS IN
IOIW 6313 3972 IOWA CITY IW
JAFL 7649 1276 JACKSONVILLE FL
JAMI 5663 3009 JACKSON MI
JAMS 8035 2880 JACKSON MS
JANC 6412 1131 JACKSONVILLE NC
JAWI 5970 3608 JANESVILLE-BELIOT WI
JENJ 5006 1409 JERSEY CITY NJ
JOJO 7421 4015 JOPLIN MO
JOPA 5422 2021 JOHNSTOWN PA
JOTN 6595 2050 JOHNSON CITY-KINGSPORT-Bristol TN-VA
KAIL 6149 3381 KANKAKEE IL
KAMI 5749 3177 KALAMAZOO-PORTAGE MI
KAMO 7027 4203 KANSAS CITY MO-KS
KEWI 5865 3526 KENOSHA WI
KITX 8832 4063 KILLEEN-TEMPLE TX
KNTH 6801 2251 KNOXVILLE TN
KOR 6135 3063 KOKOMO IN
LAFL 8084 1034 LAKELAND-WINTER HAVEN FL
<table>
<thead>
<tr>
<th>Code</th>
<th>Area Name</th>
<th>City</th>
</tr>
</thead>
<tbody>
<tr>
<td>LAIN</td>
<td>6206 3167 LAFAYETTE-WEST LAFAYETTE IN</td>
<td></td>
</tr>
<tr>
<td>LAKS</td>
<td>7098 4294 LAWRENCE KS</td>
<td></td>
</tr>
<tr>
<td>LALA</td>
<td>8587 2996 LAFAYETTE LA</td>
<td></td>
</tr>
<tr>
<td>LAMA</td>
<td>4373 1311 LAWRENCE-HAVERHILL MA-NH</td>
<td></td>
</tr>
<tr>
<td>LAMI</td>
<td>5584 3081 LANSING-EAST LANSING MI</td>
<td></td>
</tr>
<tr>
<td>LANM</td>
<td>9132 5742 LAS CRUCES NM</td>
<td></td>
</tr>
<tr>
<td>LANV</td>
<td>8665 7411 LAS VEGAS NV</td>
<td></td>
</tr>
<tr>
<td>LOAK</td>
<td>8178 4451 LANTON OK</td>
<td></td>
</tr>
<tr>
<td>LAPA</td>
<td>5348 1626 LANCASTER PA</td>
<td></td>
</tr>
<tr>
<td>LATX</td>
<td>9681 4099 LAREDO TX</td>
<td></td>
</tr>
<tr>
<td>LAWI</td>
<td>5874 4133 LA CROSSE WI</td>
<td></td>
</tr>
<tr>
<td>LEKY</td>
<td>6459 2562 LEXINGTON-FAYETTE KY</td>
<td></td>
</tr>
<tr>
<td>LEME</td>
<td>4042 1391 LEWISTON-AUBURN ME</td>
<td></td>
</tr>
<tr>
<td>LIAR</td>
<td>7721 3451 LITTLE ROCK-NORTH LITTLE ROCK AR</td>
<td></td>
</tr>
<tr>
<td>LINE</td>
<td>6832 4674 LINCOLN NE</td>
<td></td>
</tr>
<tr>
<td>LIOH</td>
<td>5921 2799 LIMA OH</td>
<td></td>
</tr>
<tr>
<td>LKLA</td>
<td>8679 3202 LAKE CHARLES LA</td>
<td></td>
</tr>
<tr>
<td>LOCA</td>
<td>9213 7878 LOS ANGELES-LONG BEACH CA</td>
<td></td>
</tr>
<tr>
<td>LOKY</td>
<td>6529 2772 LOUISVILLE KY-IN</td>
<td></td>
</tr>
<tr>
<td>LOMA</td>
<td>4399 1320 LLOWELL MA-NH</td>
<td></td>
</tr>
<tr>
<td>LONJ</td>
<td>5073 1348 LONG BRANCH-ASBURY PARK NJ</td>
<td></td>
</tr>
<tr>
<td>LOOH</td>
<td>5623 2608 LORAIN-ELYRIA OH</td>
<td></td>
</tr>
<tr>
<td>LOTX</td>
<td>8348 3660 LONGVIEW TX</td>
<td></td>
</tr>
<tr>
<td>LUTX</td>
<td>8598 4962 LUBBOCK TX</td>
<td></td>
</tr>
<tr>
<td>LYVA</td>
<td>6093 1703 LYNCHBURG VA</td>
<td></td>
</tr>
<tr>
<td>MAGA</td>
<td>7364 1865 MACON GA</td>
<td></td>
</tr>
<tr>
<td>MANH</td>
<td>4354 1388 MANCHESTER NH</td>
<td></td>
</tr>
<tr>
<td>MAOH</td>
<td>5783 2575 MANSFIELD OH</td>
<td></td>
</tr>
<tr>
<td>MAWI</td>
<td>5887 3796 MADISON WI</td>
<td></td>
</tr>
<tr>
<td>MCTX</td>
<td>9856 3764 MCALLEN-PHARR-EDINBURG TX</td>
<td></td>
</tr>
<tr>
<td>MECT</td>
<td>4740 1358 MERIDEN CT</td>
<td></td>
</tr>
<tr>
<td>MEFL</td>
<td>7925 903 MELBOURNE-TITUSVILLE-COCOA FL</td>
<td></td>
</tr>
<tr>
<td>MEOR</td>
<td>7503 8892 MEDFORD OR</td>
<td></td>
</tr>
<tr>
<td>METN</td>
<td>7471 3125 MEMPHIS TN-AR</td>
<td></td>
</tr>
<tr>
<td>MIFL</td>
<td>8351 527 MIAMI FL</td>
<td></td>
</tr>
<tr>
<td>MINN</td>
<td>5781 4525 MINNEAPOLIS-ST PAUL MN-WI</td>
<td></td>
</tr>
<tr>
<td>MITX</td>
<td>8934 4888 MIDLAND TX</td>
<td></td>
</tr>
<tr>
<td>MIWI</td>
<td>5788 3589 MILWAUKEE WI</td>
<td></td>
</tr>
<tr>
<td>MNAL</td>
<td>7692 2247 MONTGOMERY AL</td>
<td></td>
</tr>
<tr>
<td>MOAL</td>
<td>8167 2367 MOBILE AL</td>
<td></td>
</tr>
<tr>
<td>MOCA</td>
<td>8499 8473 MODESTO CA</td>
<td></td>
</tr>
<tr>
<td>MOLA</td>
<td>8148 3218 MONROE LA</td>
<td></td>
</tr>
<tr>
<td>MUNJ</td>
<td>6130 2925 MUNCIE IN</td>
<td></td>
</tr>
<tr>
<td>MUMI</td>
<td>5622 3370 MUSKEGON-NORTON SHORES-MUSKEGO MI</td>
<td></td>
</tr>
<tr>
<td>NANH</td>
<td>4394 1356 NASHUA-NH</td>
<td></td>
</tr>
<tr>
<td>NANY</td>
<td>4961 1355 NASSUA-SUFFOLK NY</td>
<td></td>
</tr>
<tr>
<td>NATN</td>
<td>7010 2710 NASHVILLE-DAVIDSON TN</td>
<td></td>
</tr>
<tr>
<td>NECT</td>
<td>4715 1373 NEW BRITAIN CT</td>
<td></td>
</tr>
<tr>
<td>NELA</td>
<td>8483 2638 NEW ORLEANS LA</td>
<td></td>
</tr>
<tr>
<td>NEMA</td>
<td>4532 1131 NEW BEDFORD MA</td>
<td></td>
</tr>
<tr>
<td>NENJ</td>
<td>5085 1434 NEW BRUNSWICK-PERTH AMBOY-SAYR NJ</td>
<td></td>
</tr>
<tr>
<td>Code</td>
<td>City</td>
<td>State</td>
</tr>
<tr>
<td>-------</td>
<td>--------------</td>
<td>---------------</td>
</tr>
<tr>
<td>NENY</td>
<td>New York NY-NJ</td>
<td>New York NY</td>
</tr>
<tr>
<td>NEOH</td>
<td>Newark OH</td>
<td>Ohio</td>
</tr>
<tr>
<td>NEVA</td>
<td>Newport VA</td>
<td>Virginia</td>
</tr>
<tr>
<td>NLCT</td>
<td>New London CT</td>
<td>Connecticut</td>
</tr>
<tr>
<td>NOCT</td>
<td>Norwalk CT</td>
<td>Connecticut</td>
</tr>
<tr>
<td>NOPA</td>
<td>Northeast PA</td>
<td>Pennsylvania</td>
</tr>
<tr>
<td>NOVA</td>
<td>Newport VA</td>
<td>Virginia</td>
</tr>
<tr>
<td>NWCT</td>
<td>New Haven CT</td>
<td>Connecticut</td>
</tr>
<tr>
<td>NWNY</td>
<td>Newburgh NY</td>
<td>New York NY</td>
</tr>
<tr>
<td>OCFL</td>
<td>Ocala FL</td>
<td>Florida</td>
</tr>
<tr>
<td>ODTX</td>
<td>Odessa TX</td>
<td>Texas</td>
</tr>
<tr>
<td>OKOK</td>
<td>Oklahoma OK</td>
<td>Oklahoma</td>
</tr>
<tr>
<td>OLWA</td>
<td>Olympia WA</td>
<td>Washington WA</td>
</tr>
<tr>
<td>OMNE</td>
<td>Omaha NE-IA</td>
<td>Nebraska</td>
</tr>
<tr>
<td>ORFL</td>
<td>Orlando FL</td>
<td>Florida</td>
</tr>
<tr>
<td>OWKY</td>
<td>Owensboro KY</td>
<td>Kentucky</td>
</tr>
<tr>
<td>OXCA</td>
<td>Oxnard CA-Ventura CA</td>
<td>California</td>
</tr>
<tr>
<td>PAFL</td>
<td>Panama City FL</td>
<td>Florida</td>
</tr>
<tr>
<td>PAMS</td>
<td>Pascagoula MS</td>
<td>Mississippi</td>
</tr>
<tr>
<td>PAMJ</td>
<td>Paterson NJ</td>
<td>New Jersey</td>
</tr>
<tr>
<td>PAW</td>
<td>Parkersburg WV-OH</td>
<td>West Virginia-OH</td>
</tr>
<tr>
<td>PEFL</td>
<td>Pensacola FL</td>
<td>Florida</td>
</tr>
<tr>
<td>PEIL</td>
<td>Peoria IL</td>
<td>Illinois</td>
</tr>
<tr>
<td>PEVA</td>
<td>Petersburg VA</td>
<td>Virginia</td>
</tr>
<tr>
<td>PHora</td>
<td>Phoenix IL</td>
<td>Arizona</td>
</tr>
<tr>
<td>PHPA</td>
<td>Philadelphia PA-NJ</td>
<td>Pennsylvania</td>
</tr>
<tr>
<td>PIAR</td>
<td>Pine Bluff AR</td>
<td>Arkansas</td>
</tr>
<tr>
<td>PIMA</td>
<td>Pittsfield MA</td>
<td>Massachusetts</td>
</tr>
<tr>
<td>PIPA</td>
<td>Pittsburgh PA</td>
<td>Pennsylvania</td>
</tr>
<tr>
<td>POME</td>
<td>Portland ME</td>
<td>Maine</td>
</tr>
<tr>
<td>PONH</td>
<td>Portsmouth-NH-ME</td>
<td>New Hampshire-Maine</td>
</tr>
<tr>
<td>PONY</td>
<td>Poughkeepsie NY</td>
<td>New York NY</td>
</tr>
<tr>
<td>POOR</td>
<td>Portland OR-CA</td>
<td>Oregon</td>
</tr>
<tr>
<td>PRRI</td>
<td>Providence RI</td>
<td>Rhode Island</td>
</tr>
<tr>
<td>PRUT</td>
<td>Provo-UT</td>
<td>Utah</td>
</tr>
<tr>
<td>PUCO</td>
<td>Pueblo CO</td>
<td>Colorado</td>
</tr>
<tr>
<td>RANC</td>
<td>Raleigh NC</td>
<td>North Carolina</td>
</tr>
<tr>
<td>RAWI</td>
<td>Racine WI</td>
<td>Wisconsin</td>
</tr>
<tr>
<td>RECA</td>
<td>Redding CA</td>
<td>California</td>
</tr>
<tr>
<td>RENV</td>
<td>Reno NV</td>
<td>Nevada</td>
</tr>
<tr>
<td>REP</td>
<td>Reading PA</td>
<td>Pennsylvania</td>
</tr>
<tr>
<td>RICA</td>
<td>Riverside CA</td>
<td>California</td>
</tr>
<tr>
<td>RIVA</td>
<td>Richmond VA</td>
<td>Virginia</td>
</tr>
<tr>
<td>RINA</td>
<td>Richland WA</td>
<td>Washington WA</td>
</tr>
<tr>
<td>ROIL</td>
<td>Rockford IL</td>
<td>Illinois</td>
</tr>
<tr>
<td>ROMN</td>
<td>Rochester MN</td>
<td>Minnesota</td>
</tr>
<tr>
<td>RONY</td>
<td>Rochester NY</td>
<td>New York NY</td>
</tr>
<tr>
<td>ROSE</td>
<td>Rock Hill SC</td>
<td>South Carolina</td>
</tr>
<tr>
<td>ROVA</td>
<td>Roanoke VA</td>
<td>Virginia</td>
</tr>
<tr>
<td>SACA</td>
<td>Sacramento CA</td>
<td>California</td>
</tr>
</tbody>
</table>
In preparation for designing the specifications for the optimization modules of the NASA LeRC least-cost network program, a study has been made of existing research on shortest path methods. This report summarizes Sonalysts review of network literature, proposes two algorithms to be implemented in the costing model, and explains the criteria that were used in their selection.

The basic algorithms underlying virtually all of the existing computer codes are very similar; two approaches were described by Dantzig [1] and Dijkstra [2] in 1959. Both of these methods are variations of the primal simplex method (see [7]). The implementation of the basic algorithm on a computer, however, can take many forms, and affects the efficiency of the algorithm tremendously. Over a decade, from 1968 to 1977, execution times to solve a problem using the same general algorithm, computer, and compiler became as much as 50 times faster [5,7]. This improvement is due to progress in the field of 'computer implementation technology' and the discovery of highly efficient ways to store and access the network data. The efficiency of a code also depends on the characteristics of the network with certain methods gaining an advantage for specific types of problems.

The NASA costing model requires the implementation of two distinct optimization algorithms, described in paragraphs 3.4.2 and 3.4.3 of the contract. We shall refer to these two requirements as problem 1 and problem 2, respectively.

Problem 1 involves finding the least-cost route from a given origin (or root) node to all other nodes in the network. (The same algorithm will be executed for each node as origin to obtain least-cost routes between each pair of nodes in the network). This problem has received considerable attention in the literature, and many different implementations have been documented, coded, and tested. Problem 2 requires the determination of the shortest paths between all pairs of nodes simultaneously. In contrast to
problem 1, this problem appears to have been solved definitively and has, therefore, received far less attention. A simple though efficient algorithm has been coded and published, and it does not appear likely that it will be improved upon. We shall discuss the variety of solutions of these two problems, propose two algorithms to be used in the costing model, and discuss the characteristics of the network that have influenced the specific choices.

The following table lists some of the factors that affect algorithm performance and our assumptions about their value in the NASA network. These assumptions have been considered throughout the literature search.

<table>
<thead>
<tr>
<th>Factor</th>
<th>Value in NASA Network</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Size of network</td>
<td>Up to 600 nodes; up to 359,400 arcs</td>
</tr>
<tr>
<td>2. Range of arc lengths</td>
<td>Unknown; expect it to be high</td>
</tr>
<tr>
<td>3. Density of network</td>
<td>Totally dense</td>
</tr>
<tr>
<td>4. Topology of network</td>
<td>Completely unstructured</td>
</tr>
<tr>
<td>5. Existence of negative arc lengths</td>
<td>No negative costs</td>
</tr>
<tr>
<td>6. Computer language (some implementation techniques exploit capabilities of assembly - language programming that are difficult or inefficient to duplicate in a higher level language.)</td>
<td>TSS/370 FORTRAN IV</td>
</tr>
<tr>
<td>7. Importance of storage requirements vs speed</td>
<td>Neither particularly critical</td>
</tr>
<tr>
<td>8. Capacities of links</td>
<td>Unlimited</td>
</tr>
</tbody>
</table>

Several articles have been written over the past twenty years that attempted to summarize and/or compare the shortest path algorithms of which the authors were aware at the time. These overviews have been very useful in comparing the techniques available and in locating additional material as well as original sources [3, 4, 5, 6, 7]. The earlier survey we studied was by Pollack and Wiebenson in 1959 [3]. Their article gives descriptions of
several methods and discusses the advantages and disadvantages of each. The authors discuss methods they attribute to Minty and, later, Ford and Fulkerson, Dantzig, and Moore among others.

A paper by Dreyfus in 1968 [4] claims the Dijkstra algorithm "outperformed all competitors." More interestingly, section 2 (pp. 401-403) discusses our problem 2 and concludes that two algorithms, both requiring $N(N-1)(N-2)$ additions and comparisons, "are easily proved, and programmed, and culminate a steady progression of successive improvements... (and hence)... there is good reason to believe that they are definitive." The amount of computation required by these methods was also considered by Hu [9] to compare favorably to that of other methods. One of these methods was coded as Algorithm 97 in the Communications of the ACM [12] and will be embellished as necessary to implement it as a submodule within the optimization module of the NASA costing model.

In 1973 a comparative study was published by Gilsinn and Witzgall [5] that summarized available methods, measured their comparative efficiency, and focused attention on the importance of implementation technology. They concluded that a code developed by Dial [13], based on an algorithm of Moore [14], and published as algorithm 360 in the Communications of the ACM, was the fastest available.

In 1979, a comparative analysis by Dial and others included results measuring the speed of the above mentioned method (referred to as S1) as well as several others. For the class of networks we are studying -- in particular, a dense network with a wide range of non-negative arc lengths -- Dial's improvement to his own code (referred to as S2) appears to be the best. It is interesting to note that the advantage of S2 over S1 appears to increase with the density of the network. Several attempts were made by the authors to improve on the code, and execution times increased in each case. The conclusion was that the overhead added in attempting to avoid unnecessary processing was greater than the attendant savings. We propose to base the submodule of the optimization module that deals with problem 1 on a code of Dial, et al.
REFERENCES


Appendix H - Procedures for Model Execution

Procedures have been created to simplify the running of the model, cleaning up after the run, and getting output printed. These procedures are invoked simply by naming the procedures. A listing of these predefined procedures is contained below.

BATRUN - execute the batch version of the model.

```
PROCDEF BATRUN
UNLOAD CNDCB
UNLOAD BLKDTA
RELEASE FT
ERASE RUNLIB
ERASE TSTLIB
ERASE TRFTBL
ERASE TRFNDS
ERASE TRFVOL
RMDS TOUT,TRFTBL
RMDS TRFNDF,TRFNDS
RMDS VLST,TRFVOL
RELEASE $$$
RELEASE RMDS
GETLIBT CNCLIB,RUNLIB
GETLIBT BATLIB,TSTLIB
CDD BATDEF
LOAD BLKDTA
CNDCB
```

CNCRUN - execute the interactive version of the model

```
PROCDEF CNCRUN
UNLOAD CNDC
UNLOAD BLKDTA
```
DBGRUN - execute the debug interactive version of the model

PROCDEF DBGRUN
UNLOAD CNDC
UNLOD BLKDTA
RELEASE FT
ERASE RUNLIB
ERASE TRFTBL
ERASE TRFNDS
ERASE TRFVOL
RMDS TOUT,TRFTBL
RMDS TRFNDF,TRFNDS
RMDS VLST,TRFVOL
RELEASE $$$
RELEASE RMDS
GETLIBT DBGLIB,RUNLIB
CDD DBGDEF
LOAD BLKDTA
CNDC
FINISH - cleanup after executing the interactive or debug interactive version of the model

PROCDEF FINISH
UNLOAD CNDC
UNLOAD BLKDTA
RELEASE FT
ERASE RUNLIB
ERASE SSSLIB
ERASE TSTLIB
ERASE TRFTBL
ERASE TRFnds
ERASE TRFVOL

LISTOUT pname - rename the CNDC output file and send it to the printer for hard copy output

. pname is the name selected by the user for renaming the output file

PROCDEF LISTOUT
PARAM PNAME
FINISH
ERASE PNAME
DDEF PNAME,VS,PNAME,RET=T
CDS CNDCOUT,PNAME
ERASE CNDCOUT
PRINT PNAME,ERASE=Y
GETLIBP libnam - copy library specified by libname into permanent storage

PROCDEF GETLIBP
PARAM LIBNAM
RMDS LIBNAM
RELEASE $$$
DDEF LIBNAM,VP,LIBNAM,OPTION=JOBLIB

GETLIBT libnam - copy library specified by libnam into temporary storage

PROCDEF GETLIBT
PARAM PNAME,TNAME
RMDS PNAME,TNAME
RELEASE $$$
RELEASE RMDS
DDEF TNAME,VP,TNAME,OPTION=JOBLIB,RET=T
Appendix I - Output Samples

The CNDC Model provides six categories of output:

1) user inputs to run

2) city pair output - exclusive of other network traffic

3) city pair output for least cost network

4) network totals

5) tariff totals

6) traffic table

The user selects which of these outputs are desired for each problem set. Figures I-1 through I-10 contain samples of all tables provided in the output.
CASE 1

TERRESTRIAL ONLY

TABLE 1 INPUT

PROBLEM 1 OF RUN OPTN1A IS ATT ONLY

NETWORK FILE NET007 CONTAINS THE FOLLOWING 6 NODES

<table>
<thead>
<tr>
<th>CODE</th>
<th>CITY NAME</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASOH</td>
<td>USER DEFINED</td>
</tr>
<tr>
<td>BAMD</td>
<td>BALTIMORE MD</td>
</tr>
<tr>
<td>CUNY</td>
<td>USER DEFINED</td>
</tr>
<tr>
<td>ELMN</td>
<td>USER DEFINED</td>
</tr>
<tr>
<td>OTNI</td>
<td>USER DEFINED</td>
</tr>
<tr>
<td>WAKA</td>
<td>USER DEFINED</td>
</tr>
</tbody>
</table>

TRAFFIC LEVEL FOR PRESTORED NODES IS 5000 VOICE CIRCUITS

Figure I-1. Network Description
### Tariffs Used

Tariff Trafitt IS PRESTORED ATT Tariff

<table>
<thead>
<tr>
<th>Schedule 1</th>
<th>Mileage</th>
<th>Charge</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1-14</td>
<td>$73.56 + $2.59 for each mile over 1 miles</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>$109.82 + $2.16 for each mile over 15 miles</td>
</tr>
<tr>
<td></td>
<td>25</td>
<td>$131.42 + $1.62 for each mile over 25 miles</td>
</tr>
<tr>
<td></td>
<td>39</td>
<td>$155.72 + $1.62 for each mile over 40 miles</td>
</tr>
<tr>
<td></td>
<td>60</td>
<td>$188.12 + $1.62 for each mile over 60 miles</td>
</tr>
<tr>
<td></td>
<td>80</td>
<td>$220.52 + $1.62 for each mile over 80 miles</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>$252.92 + $0.94 for each mile over 100 miles</td>
</tr>
<tr>
<td></td>
<td>Over 1000</td>
<td>$1098.92 + $0.58 for each mile over 1000 miles</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Schedule 2</th>
<th>Mileage</th>
<th>Charge</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1-14</td>
<td>$75.00 + $4.77 for each mile over 1 miles</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>$141.78 + $4.47 for each mile over 15 miles</td>
</tr>
<tr>
<td></td>
<td>25</td>
<td>$186.48 + $2.89 for each mile over 25 miles</td>
</tr>
<tr>
<td></td>
<td>39</td>
<td>$229.83 + $1.95 for each mile over 40 miles</td>
</tr>
<tr>
<td></td>
<td>60</td>
<td>$268.83 + $1.95 for each mile over 60 miles</td>
</tr>
<tr>
<td></td>
<td>80</td>
<td>$307.83 + $1.95 for each mile over 80 miles</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>$346.83 + $0.94 for each mile over 100 miles</td>
</tr>
<tr>
<td></td>
<td>Over 1000</td>
<td>$1192.83 + $0.58 for each mile over 1000 miles</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Schedule 3</th>
<th>Mileage</th>
<th>Charge</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1-14</td>
<td>$76.43 + $6.35 for each mile over 1 miles</td>
</tr>
<tr>
<td></td>
<td>15</td>
<td>$165.33 + $5.48 for each mile over 15 miles</td>
</tr>
<tr>
<td></td>
<td>25</td>
<td>$220.13 + $4.03 for each mile over 25 miles</td>
</tr>
<tr>
<td></td>
<td>39</td>
<td>$280.58 + $3.03 for each mile over 40 miles</td>
</tr>
<tr>
<td></td>
<td>60</td>
<td>$341.18 + $2.31 for each mile over 60 miles</td>
</tr>
<tr>
<td></td>
<td>80</td>
<td>$387.38 + $1.95 for each mile over 80 miles</td>
</tr>
<tr>
<td></td>
<td>100</td>
<td>$426.38 + $0.97 for each mile over 100 miles</td>
</tr>
<tr>
<td></td>
<td>1000</td>
<td>$1299.38 + $0.58 for each mile over 1000 miles</td>
</tr>
</tbody>
</table>

Tariff file A contains the following 1 category A rate centers.

Figure I-2. ATT Tariff Description
### TARIFS USED

TARIFF TRFWU IS THE PRESTORED WU TARIFF

Appropriate rates for this tariff are as follows:

<table>
<thead>
<tr>
<th>Type</th>
<th>Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Long Haul</td>
<td>$925.00</td>
</tr>
<tr>
<td>Medium Haul</td>
<td>$695.00</td>
</tr>
<tr>
<td>Short Haul</td>
<td>$580.00</td>
</tr>
</tbody>
</table>

There are 10 satellite access city pairs for this tariff:

<table>
<thead>
<tr>
<th>City Pair</th>
<th>Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>BAND - CIIL</td>
<td>SHORT HAUL</td>
</tr>
<tr>
<td>BAND - LOCA</td>
<td>LONG HAUL</td>
</tr>
<tr>
<td>CIIL - LOCA</td>
<td>MEDIUM HAUL</td>
</tr>
<tr>
<td>CIIL - NENY</td>
<td>SHORT HAUL</td>
</tr>
<tr>
<td>CIIL - PHPA</td>
<td>SHORT HAUL</td>
</tr>
<tr>
<td>CIIL - WADC</td>
<td>LONG HAUL</td>
</tr>
<tr>
<td>DEMI - LOCA</td>
<td>LONG HAUL</td>
</tr>
<tr>
<td>LOCA - NENY</td>
<td>LONG HAUL</td>
</tr>
<tr>
<td>LOCA - PHPA</td>
<td>LONG HAUL</td>
</tr>
<tr>
<td>LOCA - WADC</td>
<td>LONG HAUL</td>
</tr>
</tbody>
</table>

---

Figure I-3. WU Tariff Description
TARIFFS USED

TARIFF TRFSBS IS THE PRESTORED SBS TARIFF
THE APPROPRIATE RATES FOR THIS TARIFF ARE AS FOLLOWS

<table>
<thead>
<tr>
<th>Description</th>
<th>Charge</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAC CHARGE</td>
<td>$17,850.00</td>
</tr>
<tr>
<td>SCU CHARGE</td>
<td>$5,700.00</td>
</tr>
<tr>
<td>FTU CHARGE</td>
<td>$2,550.00</td>
</tr>
<tr>
<td>MINIMUM CAU CHARGE</td>
<td>$17,850.00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Number of CAUs</th>
<th>Charge</th>
</tr>
</thead>
<tbody>
<tr>
<td>Up to 150 CAUs</td>
<td>$95.00 for each CAU</td>
</tr>
<tr>
<td>Up to 300 CAUs</td>
<td>$1,425.00 + $90.00 for each CAU over 150 CAUs</td>
</tr>
<tr>
<td>Over 300 CAUs</td>
<td>$2,775.00 + $65.00 for each CAU over 300 CAUs</td>
</tr>
</tbody>
</table>

Figure I-4. SBS Tariff Description
TABLE 2

THE LEAST-COST ROUTE FROM CIIL TO BAMD IS CIIL-BAMD

THE TOTAL CIRCUIT MILEAGE FROM CIIL TO BAMD IS 604 MILES.

THE TOTAL COST OF CIRCUITS FROM CIIL TO BAMD IS $284017.51

<table>
<thead>
<tr>
<th>LINK</th>
<th>TOTAL AIRLINE MILEAGE</th>
<th>TRAFFIC VOLUME ON LINK (VC'S)</th>
<th>FACILITY SIZE</th>
<th>COST PER CIRCUIT (DOLLARS)</th>
<th>TOTAL COST OF CIRCUITS (DOLLARS)</th>
<th>TARIFF</th>
<th>SERVICE</th>
</tr>
</thead>
<tbody>
<tr>
<td>CIIL-BAMD</td>
<td>604</td>
<td>1253</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>5</td>
<td>226.67</td>
</tr>
</tbody>
</table>

COST PER VOICE CIRCUIT BASED ON 10979 VOICE CIRCUITS AT ORIGINATING NAC LOCATION

SATELLITE COST PER CIRCUIT BASED ON FOLLOWING CHARGES IN ORIGINATING CITY

<table>
<thead>
<tr>
<th>NACS</th>
<th>10 3 COST $17850.00 EACH $178500.00</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCUS</td>
<td>20 3 COST $5700.00 EACH $114000.00</td>
</tr>
<tr>
<td>FTUS</td>
<td>949 3 COST $2550.00 EACH $139950.00</td>
</tr>
<tr>
<td>9 GROUPS CAUS</td>
<td>1116 3 BASE COST $27750.00 PLUS</td>
</tr>
<tr>
<td></td>
<td>INCREMENT $65.00 EACH OVER 300 $727110.00</td>
</tr>
<tr>
<td>CAUS</td>
<td>955 3 BASE COST $27750.00 PLUS</td>
</tr>
<tr>
<td></td>
<td>INCREMENT $65.00 EACH OVER 300 $69625.00</td>
</tr>
</tbody>
</table>

TOTAL EARTH STATION CHARGES $2488585.00

Figure I-5. SBS Least Cost Route
Table 2

<table>
<thead>
<tr>
<th>LINK</th>
<th>TOTAL AIRLINE MILEAGE</th>
<th>TRAFFIC VOLUME ON LINK (VC'S)</th>
<th>FACILITY SIZE</th>
<th>COST PER CIRCUIT (Dollars)</th>
<th>TOTAL COST OF CIRCUITS (Dollars)</th>
<th>TARIFF</th>
<th>SERVICE</th>
</tr>
</thead>
<tbody>
<tr>
<td>NANY-NENY</td>
<td>20</td>
<td>21949</td>
<td>6</td>
<td>5</td>
<td>5</td>
<td>$120.62</td>
<td>2647488.38</td>
</tr>
<tr>
<td>NENY-CIIL</td>
<td>711</td>
<td>3228</td>
<td>0</td>
<td>5</td>
<td>3</td>
<td>$580.00</td>
<td>1872240.00</td>
</tr>
</tbody>
</table>

Link Summary

The least-cost route from Nany to CIIL is NANY-NENY-CIIL.

The total circuit mileage from Nany to CIIL is 731 miles.

The total cost of circuits from Nany to CIIL is $451,972,838.

Figure I-6. Least Cost Route
TABLE 3
NANY - CIIL

THE LEAST-COST ROUTE FROM NANY TO CIIL IS
NANY-NENY-CIIL

LINK SUMMARY

<table>
<thead>
<tr>
<th>LINK</th>
<th>TOTAL NETWORK TRAFFIC VOLUME ON LINK (VC'S)</th>
<th>FACILITY SIZE</th>
<th>COST PER CIRCUIT (DOLLARS)</th>
<th>TOTAL COST OF CIRCUITS (DOLLARS)</th>
<th>TARIFF</th>
<th>SERVICE</th>
</tr>
</thead>
<tbody>
<tr>
<td>NANY-NENY</td>
<td>22674</td>
<td>6 1 7</td>
<td>$120.62</td>
<td>2734937.88</td>
<td>TRFATT-1</td>
<td>TERRESTRIAL</td>
</tr>
<tr>
<td>NENY-CIIL</td>
<td>3228</td>
<td>0 5 3</td>
<td>$580.00</td>
<td>1872240.00</td>
<td>TRF4W</td>
<td>SATELLITE</td>
</tr>
</tbody>
</table>

Figure 1-7. Least Cost Network
### TABLE 4. OUTPUT NETWORK TOTALS

<table>
<thead>
<tr>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total terrestrial circuit mileage in the least-cost network</td>
<td>2466 miles</td>
</tr>
<tr>
<td>Total satellite circuit mileage in the least-cost network</td>
<td>32170 miles</td>
</tr>
<tr>
<td>Combined total circuit mileage in the least-cost network</td>
<td>34636 miles</td>
</tr>
<tr>
<td>Total number of terrestrial voice circuits in the least-cost network</td>
<td>116050 circuits</td>
</tr>
<tr>
<td>Total number of satellite voice circuits in the least-cost network</td>
<td>24960 circuits</td>
</tr>
<tr>
<td>Combined total number of voice circuits in the least-cost network</td>
<td>141010 circuits</td>
</tr>
<tr>
<td>Total cost of terrestrial circuits in the least-cost network</td>
<td>$2,419,600.00</td>
</tr>
<tr>
<td>Total cost of satellite circuits in the least-cost network</td>
<td>$1,162,768.00</td>
</tr>
<tr>
<td>Combined total cost of circuits in the least-cost network</td>
<td>$3,582,368.00</td>
</tr>
</tbody>
</table>

### SUMMARY OF SATELLITE EARTH STATION EQUIPMENT

<table>
<thead>
<tr>
<th>Equipment</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of NACS</td>
<td>25</td>
</tr>
<tr>
<td>Number of SCUs</td>
<td>45</td>
</tr>
<tr>
<td>Number of FTUs</td>
<td>3568</td>
</tr>
<tr>
<td>Number of CAs</td>
<td>24,960</td>
</tr>
</tbody>
</table>

SBS mixed algorithm to select number of NACS went through 6 iterations. The optimal solution was with 7 NAC locations. The total cost at each number of NAC locations is as follows:

<table>
<thead>
<tr>
<th>Number of NAC Locations</th>
<th>Total System Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>$5,171,828.00</td>
</tr>
<tr>
<td>3</td>
<td>$3,914,605.00</td>
</tr>
<tr>
<td>4</td>
<td>$3,773,849.00</td>
</tr>
<tr>
<td>5</td>
<td>$3,707,289.00</td>
</tr>
<tr>
<td>6</td>
<td>$3,665,544.00</td>
</tr>
<tr>
<td>7</td>
<td>$3,582,368.00</td>
</tr>
</tbody>
</table>

The 7 NAC locations are as follows:

- BAMD
- LOCA
- DEMI
- MANY
- PHPA
- WADC
- HENY

Figure 1-8. Output Network Totals
<table>
<thead>
<tr>
<th>TARIFF</th>
<th>TOTAL CIRCUIT MILEAGE</th>
<th>TOTAL VOICE CIRCUITS</th>
<th>PERCENT OF NETWORK CIRCUITS</th>
<th>TOTAL COST</th>
<th>PERCENT OF NETWORK COST</th>
<th>SERVICE</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRFATT</td>
<td>2466</td>
<td>116050</td>
<td>0.82</td>
<td>24196000.</td>
<td>0.68</td>
<td>TERRESTRIAL</td>
</tr>
<tr>
<td>TRFSBS</td>
<td>32170</td>
<td>24960</td>
<td>0.18</td>
<td>11627686.</td>
<td>0.32</td>
<td>SATELLITE</td>
</tr>
</tbody>
</table>

Figure I-9. Tariff Summary
TABLE 6 TRAFFIC TABLE

<table>
<thead>
<tr>
<th>BAMD</th>
<th>DEMI</th>
<th>LOCA</th>
<th>NANY</th>
<th>NENY</th>
<th>PHPA</th>
<th>WADC</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1230</td>
<td>751</td>
<td>1813</td>
<td>4594</td>
<td>3618</td>
<td>6204</td>
</tr>
<tr>
<td>1230</td>
<td>0</td>
<td>897</td>
<td>1067</td>
<td>2386</td>
<td>1353</td>
<td>1342</td>
</tr>
<tr>
<td>751</td>
<td>897</td>
<td>0</td>
<td>725</td>
<td>1142</td>
<td>800</td>
<td>787</td>
</tr>
<tr>
<td>1813</td>
<td>1067</td>
<td>725</td>
<td>0</td>
<td>20835</td>
<td>3475</td>
<td>1774</td>
</tr>
<tr>
<td>4594</td>
<td>2386</td>
<td>1142</td>
<td>20835</td>
<td>0</td>
<td>6380</td>
<td>4201</td>
</tr>
<tr>
<td>3618</td>
<td>1353</td>
<td>800</td>
<td>3475</td>
<td>8380</td>
<td>0</td>
<td>3131</td>
</tr>
<tr>
<td>6204</td>
<td>1342</td>
<td>787</td>
<td>1774</td>
<td>4201</td>
<td>3131</td>
<td>0</td>
</tr>
</tbody>
</table>

Figure I-10. Traffic Table
Appendix J - Common Blocks

The data structures used within the model (defined in section 4) are contained in common blocks. Below is a listing of the common blocks used within the CNDC model.

ARRYS

COMMON /ARRYS/ MXPRN,MXCTA,MXACC
INTEGER*4 MXPRN,MXCTA,MXACC

ATT

COMMON /ATT/ TATBAS,TATINC,NALT,NLEVEL,NSCHED,TATMLG,ALSTD,-
1 NPLEVL,LSTED,NINLST,TPTR
REAL TATBAS(9,3,13),TATINC(9,3,13)
INTEGER*4 NALT,NLEVEL,NSCHED,TATMLG(9),ALSTD(200),LSTED(600),-
1 NINLST(13),TPTR(13),NPLEVEL

CSTMXD

COMMON /CSTMXD/ TRRCST,STLCST,TRRMLG,TRRCKT,STLMLG,-
1 STLCKT
REAL*4 TRRCST,STLCST
INTEGER*4 TRRMLG,TRRCKT,STLMLG,STLCKT

CSTOUT

COMMON /CSTOUT/ COST,MINCP,NETOTL,OROUTE,SVC
INTEGER*4 NETOTL(600,600)
INTEGER*2 OROUTE(600,600),SVC(600,600)
REAL COST(600,600),MINCP(600,600)

CURFIL

COMMON /CURFIL/ FLNM,STATUS,NFIL
INTEGER*4 NFIL,STATUS(20)
REAL*8 FLNM(20)
HEADRS

COMMON /HEADRS/ HNET,HTAR,HNOD,HTFC,HPATT,-
1 HNATT,HPWU,HNWU,HPSBS,HNSBS,-
2 HEXEC,UNET,UTAR,HWOL
REAL*8 HNET(2),HTAR(2),HNOD(2),HTFC(2),
1 HPATT(2),HNATT,HPWU(2),HNWU,HPSBS(2),-
2 HNSBS,HEXEC(2),UNET,UTAR,HWOL(2)

INPATT

COMMON /INPATT/ NTLEV,TATMIL,INCSC,ADJ,-
1 PRCATA,DRCATA,MDIR,MPBLK,NCATA
INTEGER*4 NTLEV,TATMIL(9),ADJ(600)
REAL INCSC(9,3)
INTEGER*4 PRCATA(400),DRCATA(20),MDIR,MPBLK,NCATA

INPEXC

COMMON /INPEXC/ NETFIL,FILNAM,MAXTRF,MAXRST,RUNID,-
1 NRSTRT,NOUT,OPTN,NTRF,OUTCNT,LEVEL
REAL*8 NETFIL(10),FILNAM(130),RUNID
INTEGER*4 MAXTRF,MAXRST,NRSTRT,NOUT,-
1 OPTN(10),NTRF(10),OUTCNT(6,10),LEVEL(10)

INPNET

COMMON /INPNET/ CROPT,LVLTRF,NTRFC,MAXUSR,-
1 TOTNDS,PERCNT,VC,HC,VCIRTS
INTEGER*4 CROPT,LVLTRF,NTRFC,MAXUSR,TOTNDS,-
1 PERCNT,VC(600),HC(600),VCIRTS(600)

INPNOD

COMMON /INPNOD/ NNDS,MAXNDS,CODE,CODEL,CODER
INTEGER*4 NNDS,MAXNDS,CODE(600),CODEL(600),-
1 CODER(600)

INPSBS

COMMON /INPSBS/ NSBTAB,NAC,SCU,FTU,CAUMIN,TSBCAM,-
1 INC SB
INTEGER*4 NSBTAB,TSBCAM(3)
REAL NAC,SCU,FTU,CAUMIN,INCSB(3)
INPWU

COMMON /INPWU/ NADJ, CHGLH, CHGMH, CHGSH, CAT
INTEGER*4 NADJ, CAT(600)
REAL CHGLH, CHGMH, CHGSH

LUN

COMMON /LUN/ NETDIR, TRFATT, TRFDIR, TRFNS, TRFSBS, -
1 TRFTBL, TRFWU, EXECUTE, LUNNET, LUNTRF, DEBUG, -
2 TRFVOL, LUNSPL, LUNOUT
INTEGER NETDIR, TRFATT, TRFDIR, TRFNS, TRFSBS, -
1 TRFTBL, TRFWU, EXECUTE, LUNNET, LUNTRF, -
2 DEBUG, TRFVOL, LUNSPL, LUNOUT

NETFIL

COMMON /NETFIL/ NFILES, NNFIL, NETDEF
INTEGER*4 NNFIL, NETDEF(20)
REAL*8 NFILES(20)

PRSNDS

COMMON /PRSNDS/ NPRNDS, KDIR, KPBLK, KRNODE, PRNDS
INTEGER*4 NPRNDS, KDIR, KPBLK, KRNODE(20), PRNDS(350)

PRSWU

COMMON /PRSWU/ NACC, MAXACC, LSTACC
INTEGER*4 NACC, MAXACC, LSTACC(65)

RNODES

COMMON /RNODES/ NDIR, NPBLK, DRNODE, RTCNTR, V, H
INTEGER*4 NDIR, NPBLK, DRNODE(25), RTCNTR(600), V(600), -
1 H(600)

RUN

COMMON /RUN/ TF, FILNET, NTARIF, OPTION, NCASE, TRFLVL, OUTFLG, -
1 NNODES, TTYP
REAL*8 TF(13), FILNET
INTEGER*4 NNODES, NTARIF, OPTION, NCASE, TRFLVL, OUTFLG(6), -
1 TTYP(13)
SATMXD

COMMON /SATMXD/ SATOTL
INTEGER*4 SATOTL(600,600)

SBS

COMMON /SBS/ TSBCAI,TSBCAB,TSBCMN,TSBSCU,TSBFTU,TSBNAC,CPNA
1 MAXGRP,NSCUPN,NTAB,TSBBKP,VCPFTU,NPTAB
REAL TSBCAI(3,13),TSBCMNN(13),TSBSCU(13),-1
1 TSBCAB(3,13),TSBFTU(13),TSBNAC(13)
INTEGER*4 CPNA,MAXGRP,NSCUPN,NTAB,TSBBKP(3),-
1 VCPFTU,NPTAB

SBSEQP

COMMON /SBSEQP/ QNAC,QSCU,QFTU,QCAU
INTEGER*4 QNAC,QSCU,QFTU,QCAU

SBSMXD

COMMON /SBSMXD/ SBMX,NACNMB,NACLOC,NODPTR
INTEGER*4 SBMX,NACNMB,NACLOC(600),NODPTR(600)

SBSMX2

COMMON /SBSMX2/ SATCST,TOTSAT,SATNVC,STABLE,SYSTBL
REAL*4 SATCST(600),TOTSAT(600)
INTEGER*4 SATNVC(600)
INTEGER*2 STABLE(600)
LOGICAL SYSTBL

TARFIL

COMMON /TARFIL/ TFILES,NTFIL,TARDEF
INTEGER*4 NTFIL,TARDEF(13)
REAL*8 TFILES(13)

TRFC

COMMON /TRFC/ TRAFIC
INTEGER*4 TRAFIC(600,600)
Several of the common block variables are constant for the current version of the model, as, for example, the maximum number of prestored nodes. These constants are initialized in the Block Data Subprogram. Below is a listing of this program.

```
BLOCK DATA
*LOAD INPEXC,CNDC.COMMON
*LOAD INPNOD,CNDC.COMMON
*LOAD INPNET,CNDC.COMMON
*LOAD LUN,CNDC.COMMON
*LOAD HEADRS,CNDC.COMMON
*LOAD ATT,CNDC.COMMON
*LOAD SBS,CNDC.COMMON
*LOAD RUN,CNDC.COMMON
*LOAD WU,CNDC.COMMON
*LOAD PRSWU,CNDC.COMMON
*LOAD ARrys,CNDC.COMMON

C
C ARrys
C
DATA MXPRN/350/, MXCTA/400/, MXACC/65/
C
C INPEXC
C
DATA MXTRF/13/, MAXRST/10/, NOUT/6/
C
C INPNOD
C
DATA MAXNDS/600/
C
C INPNET
C
DATA MAXUSR/250/
C
C LOGICAL UNIT NUMBERS
C
DATA NETDIR/1/, TRFATT/2/, TRFDIR/3/, TRFNDS/4/, TRFSBS/5/,-
1 TRFTBL/6/, TRFWU/7/, EXECUTE/8/, LUNNET/30/, LUNTRF/50/,-
1 DEBUG/91/, TRFVOL/12/, LUNSPL/70/, LUNOUT/99/```
FILE HEADERS

DATA HNET/'NETWRK D' , 'IRECTORY '/
DATA HTAR/'TARIFF D', 'IRECTORY '/
DATA HNOD/'TRAFFIC ', 'NODES '/
DATA HTFC/'TRAFFIC ', 'ABLE '/
DATA HPATT/'ATT TARI', 'FF '/
DATA HNATT/'ATT-NASA'/
DATA HPUW/'WU TARI', 'F '/
DATA HNMU/'WU-NASA '/
DATA HPSBS/'SBS TARI', 'FF '/
DATA HNSBS/'SBS-NASA'/
DATA HEXEC/'EXECUTIO', 'N FILE '/
DATA UNET/'NETWORK '/
DATA HVOL/'TRAFFIC ', 'VOLUMES '/

ATT

DATA NATL/200/, NLEV/9/, NSCH/3/

SBS

DATA MAXG/100/, NSCNN/2/, VCPFTU/20/, NTAB/3/, CPNA/372/

WU

DATA NWUCAT/3/, NWUTHR/2080/, MAXSAC/65/

RUN

DATA NCA/0/

PRSWU

DATA MAXACC/65/

END
Appendix K - Directory of Files

There are many files which make up the CNDC model - source code, object code libraries, user defined data base files, prestored data base files, and command files. Below is a listing of these files and a brief description.

Source Code of Model:

SOURCE.BINPUT
SOURCE.TNTWRK
SOURCE.BCNDC
SOURCE.BOUTPUT
SOURCE.BATCH
SOURCE.BINIT
SOURCE.FSBSMXD
SOURCE.BTARCON
SOURCE.BLKDTA
SOURCE.BUTLTY
SOURCE.BNETCON
SOURCE.PBLKDTA
SOURCE.BTARIFF
SOURCE.BDRCTRY

Back up of Model Source Code

BACKUP.BCNDC
BACKUP.BINPUT
BACKUP.BNETCON
BACKUP.BTARCON
BACKUP.BOUTPUT
BACKUP.TNTWRK
BACKUP.BINIT
BACKUP.BDRCTRY
BACKUP.BTARIFF
BACKUP.BUTLTY
BACKUP.BATCH
BACKUP.FSBSMXD
BACKUP.CNCBAT

Debug version of the model:

A version of the model with debug statements is maintained to enable one to quickly locate problems if and when they develop.
DEBUG.BCNDC
DEBUG.BINPUT
DEBUG.BNETCON
DEBUG.BTARCON
DEBUG.BINIT
DEBUG.BUTLY
DEBUG.FSBSMXD
DEBUG.BOUTPUT
DEBUG.BATCH

Libraries of object code:

CNCLIB    - object code of source version
PRSLIB    - object code of program which creates and maintains the
prestored data base
DBGLIB    - object code of debug version
BATLIB    - library of modules required to run the model in batch
mode
DBTLIB    - debug version of BATLIB

Common Blocks Required by Source Code

CNDC.COMMON
PRSTRD.COMMON

Command Files Which Assemble Files Required to Execute Specific Version of
the Model:

CNCDEF
DBGDEF
BATDEF

Files that Make Up Prestored Data Base

TRFATT    ATT tariff
TRFWU     WU tariff
TRFSBS    SBS tariff
NETDIR    directory to user defined networks
TRFDIR    directory to user defined tariffs
TOUT\    traffic table
TRFNDF    prestored network
VLST      prestored network ordered by volume, most busy to least
busy
NLST      annotated listing of prestored network
TLST      annotated listing of prestored traffic table
PTRFC traffic table from which prestored traffic table constructed

PNODES SMSA file from which prestored network constructed

Files that Make Up User Defined Data Base

User Defined Networks

NET001

NET020

User Defined Tariffs

TAR001

TAR013

Execution Control File

EXECUTE

Procedures described in Appendix H are found in USERLIB(SYSPRO).
1.0 SBS Tariff Change

Change one to the manual reflects a change to the Satellite Business Systems tariff as set forth in F.C.C. No. 2, effective May 1, 1983. This tariff is described below. All charges specified are monthly recurring charges.

<table>
<thead>
<tr>
<th>Service</th>
<th>Monthly Recurring Charge</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network Access Center (each)</td>
<td>NAC $17,850.00</td>
</tr>
<tr>
<td>NAC Supplemental Capacity Units</td>
<td>SCU $5,700.00</td>
</tr>
<tr>
<td>Full Time Transmission Units</td>
<td>FTU $2,550.00</td>
</tr>
<tr>
<td>Minimum CAU Charges</td>
<td></td>
</tr>
<tr>
<td>Connection Arrangement Units</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CAU $17,850.00</td>
</tr>
</tbody>
</table>

**NUMBER OF CAUs AT EACH NAC**

<table>
<thead>
<tr>
<th>CAUs Range</th>
<th>Monthly Recurring Charge</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 CAU - 150 CAUs</td>
<td>$100 per CAU</td>
</tr>
<tr>
<td>151 CAUs - 300 CAUs</td>
<td>$15,000.00 plus $95.00 per CAU in excess of 150</td>
</tr>
<tr>
<td>More than 300 CAUs</td>
<td>$29,250.00 plus $70.00 per CAU in excess of 300</td>
</tr>
</tbody>
</table>