MODELLING AND OPTIMISATION
OF MIXED VARIABLE PROCESSES
MODELLING AND DYNAMIC OPTIMISATION
OF MIXED VARIABLE PROCESSES
USING HYBRID SIMULATION TECHNIQUES

By
KURT ANTON STRÖBELE, B.Sc. (Eng)

A Thesis
Submitted to the School of Graduate Studies
in Partial Fulfillment of the Requirements
for the Degree
Doctor of Philosophy.

McMaster University
May, 1973

© Kurt Anton Strobele 1974
DOCTOR OF PHILOSOPHY (1973)
(MECHANICAL ENGINEERING)

McMASTER UNIVERSITY
Hamilton, Ontario

TITLE: Modelling and Dynamic Optimisation of Mixed Variable Processes Using Hybrid Simulation Techniques

AUTHOR: Kurt Anton Ströbele, B. Sc. (Eng.) (Witwatersrand University)

SUPERVISORS: Dr. M.C. de Malherbe, Prof. R. Newcombe.

NUMBER OF PAGES: 287, G-36

(iii)
TO MY PARENTS
ABSTRACT

An approach to study control problems in complex systems of interacting, mixed variable processes is presented. Discrete event simulation is combined with continuous simulation to provide a versatile method for system modelling and analysis. Provisions are made to include a human model into the model of a real system which allows an accurate reproduction of systems in which human control is an integral part. A special model display unit was built to provide feedback to the model operator, which can be readily connected to a hybrid computer. The display unit permits operation of the model by process people with no detailed knowledge of simulation which is an added advantage.

A copper smelter system was modelled and simulated on a hybrid computer which includes discrete as well as continuous processes and requires human control to integrate the process to achieve a maximum output. The system thus represents a good example for a variety of similar systems encountered in industry. Semi-empirical models of smelting processes were developed based on existing knowledge of the processes. Using the display unit the evolutionary development necessary for some models was possible.

The simulation model was calibrated and validated to accurately reproduce the real system behaviour. Based on simulation experiments an improved operating technique for controlling aspects of the smelting process was tested and an increase
in production was predicted. This operating technique was tested in plant trials and adopted while the predicted production increase was verified.

The validated model was then used to develop heuristic decision and forecasting algorithms for automatic optimal control of model operations. Based on the realistic model different decision strategies could be tested by observing their success in controlling the simulation model. The control algorithms were refined to give a satisfactory control of the simulation model, without the need of human intervention. The smelter model with the heuristic control can thus serve as a useful tool in developing an on-line control system for the real system, which could be used as a decision making aid to operating personnel.
ACKNOWLEDGEMENTS

The sincerest thanks are expressed to my wife Marianne for her encouragement, help and patience during the many months required for the preparation of this thesis.

The help and direction of my supervisor, Professor M.C. de Malherbe, and acting supervisor, Professor R. Newcombe, not only in this work, but throughout my graduate work at McMaster University is sincerely appreciated.

The generous support of the National Research Council, where this project originated and where most of the experimental work was done, and in particular the assistance provided there by Dr. U.P. Graefe and Mr. L.K. Nenonen in this study is gratefully acknowledged.

The cooperation of Noranda Mines in providing data for the model and implementing a plant test, and the assistance provided by personnel without which this study would not have been possible is gratefully acknowledged.

Thanks are expressed to Drs. J.L. Duncan, R.M. Korol and Dr. R.L. Judd for their help as members of the supervisory committee and to other members of staff for their valuable assistance throughout the course of this work.

Finally, the financial assistance by McMaster University and by the National Research Council throughout the course of my studies in the form of Scholarships is appreciated.
# TABLE OF CONTENTS

ACKNOWLEDGEMENTS

LIST OF FIGURES

LIST OF TABLES

## CHAPTER 1.0  INTRODUCTION

1.1 PROBLEM FORMULATION  
1  

1.2 SYSTEM SELECTION  
3  

1.3 DESCRIPTION OF SYSTEM  
4

## CHAPTER 2.0  LITERATURE REVIEW AND REVIEW OF THEOREY

2.1 MODELLING OF COPPER SMELTER OPERATIONS  
10  

2.2 MODELS OF OTHER INDUSTRIAL PROCESSES  
12  

2.3 MODELLING OF SYSTEMS  
15  

2.4 SIMULATION  
18  

2.5 MODEL DEVELOPMENT  
21

## CHAPTER 3.0  SYSTEM ANALYSIS

3.1 COPPER REFINING PROCESSES  
26  

3.2 MATERIAL TRANSPORTATION BY CRANES  
32  

3.3 SMELTER MODEL  
35
<table>
<thead>
<tr>
<th>CHAPTER 4.0</th>
<th>SMELTER MODEL DEVELOPMENT</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1</td>
<td>DEFINITION AND OVERVIEW OF THE SYSTEM MODEL</td>
<td>36</td>
</tr>
<tr>
<td>4.2</td>
<td>CONTROL OF THE SIMULATION MODEL</td>
<td>48</td>
</tr>
<tr>
<td>4.3</td>
<td>CRANE SUBMODELS</td>
<td>59</td>
</tr>
<tr>
<td>4.4</td>
<td>REVERBERATORY FURNACE SUBMODELS</td>
<td>65</td>
</tr>
<tr>
<td>4.5</td>
<td>CONVERTER FURNACE SUBMODELS</td>
<td>71</td>
</tr>
<tr>
<td>4.6</td>
<td>ANODE FURNACE SUBMODELS</td>
<td>77</td>
</tr>
<tr>
<td>4.7</td>
<td>MISCELLANEOUS SUBMODELS</td>
<td>79</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CHAPTER 5.0</th>
<th>MODEL VALIDATION</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1</td>
<td>VALIDATION OF SIMULATION MODELS</td>
<td>82</td>
</tr>
<tr>
<td>5.2</td>
<td>VERIFICATION OF THE MODEL</td>
<td>88</td>
</tr>
<tr>
<td>5.3</td>
<td>CALIBRATION OF THE MODEL</td>
<td>91</td>
</tr>
<tr>
<td>5.4</td>
<td>VALIDATION OF THE OVERALL MODEL PERFORMANCE</td>
<td>93</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CHAPTER 6.0</th>
<th>MODEL ANALYSIS AND RESULTS BY MANUAL CONTROL</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.1</td>
<td>APPLICATIONS OF THE MODEL</td>
<td>102</td>
</tr>
<tr>
<td>6.2</td>
<td>MEASURE OF PERFORMANCE OF THE SYSTEM</td>
<td>104</td>
</tr>
<tr>
<td>6.3</td>
<td>MODEL RESPONSE TO &quot;NO-MATTE TRANSFER&quot; POLICY</td>
<td>109</td>
</tr>
<tr>
<td>6.4</td>
<td>SYSTEM RESPONSE TO &quot;NO MATTE TRANSFER&quot; POLICY</td>
<td>117</td>
</tr>
<tr>
<td>6.5</td>
<td>DISCUSSION OF RESULTS</td>
<td>122</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CHAPTER 7.0</th>
<th>DEVELOPMENT OF AUTOMATIC MODEL CONTROL</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.1</td>
<td>DEFINITION OF PROBLEM AND OBJECTIVES</td>
<td>125</td>
</tr>
<tr>
<td>7.2</td>
<td>HEURISTIC APPROACH TO DECISION MAKING PROCESS</td>
<td>127</td>
</tr>
<tr>
<td>Chapter/Appendix</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>-----------------</td>
<td>----------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>7.3</td>
<td>CONTROL OF SMELTER ACTIVITIES</td>
<td>129</td>
</tr>
<tr>
<td>7.4</td>
<td>FORECASTING EXPECTED PROCESS STATES</td>
<td>144</td>
</tr>
<tr>
<td>7.5</td>
<td>RESULTS OF AUTOMATIC MODEL CONTROL</td>
<td>147</td>
</tr>
<tr>
<td><strong>Chapter 8.0</strong></td>
<td>CONCLUSIONS</td>
<td>151</td>
</tr>
<tr>
<td>REFERENCES</td>
<td></td>
<td>154</td>
</tr>
<tr>
<td>APPENDICES</td>
<td>DOCUMENTATION OF SMELTER MODEL</td>
<td></td>
</tr>
<tr>
<td>A</td>
<td>OVERALL SIMULATION CONTROL</td>
<td>161</td>
</tr>
<tr>
<td>B</td>
<td>CRANE SUBMODELS</td>
<td>187</td>
</tr>
<tr>
<td>C</td>
<td>REVERBERATORY SUBMODELS</td>
<td>216</td>
</tr>
<tr>
<td>D</td>
<td>CONVERTER SUBMODELS</td>
<td>238</td>
</tr>
<tr>
<td>E</td>
<td>ANODE FURNACE AND MISCELLANEOUS SUBMODELS</td>
<td>254</td>
</tr>
<tr>
<td>F</td>
<td>PROGRAMME REFERENCE LISTS</td>
<td>263</td>
</tr>
<tr>
<td>G</td>
<td>PROGRAMME LISTING</td>
<td>287</td>
</tr>
<tr>
<td>FIGURE</td>
<td>DESCRIPTION</td>
<td>PAGE</td>
</tr>
<tr>
<td>--------</td>
<td>--------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>1.3.1</td>
<td>SMELTER LAYOUT</td>
<td>8</td>
</tr>
<tr>
<td>1.3.2</td>
<td>MATERIAL FLOW CHART</td>
<td>9</td>
</tr>
<tr>
<td>3.1.1</td>
<td>TYPICAL REVERBERATORY FURNACE</td>
<td>30</td>
</tr>
<tr>
<td>3.1.2</td>
<td>COPPER CONVERTER FURNACE</td>
<td>31</td>
</tr>
<tr>
<td>4.1.1</td>
<td>COPPER SMELTING PROCESSES WITHIN SYSTEM BOUNDARY</td>
<td>41</td>
</tr>
<tr>
<td>4.1.2</td>
<td>SMELTER DISPLAY UNIT</td>
<td>42</td>
</tr>
<tr>
<td>4.1.3</td>
<td>SMELTER DISPLAY UNIT: INFORMATION FORMAT</td>
<td>43</td>
</tr>
<tr>
<td>4.1.4</td>
<td>ANALOG COMPUTER AND DISPLAY UNIT</td>
<td>44</td>
</tr>
<tr>
<td>4.1.5</td>
<td>REPRESENTATION OF OVERALL MODEL SHOWING MATERIAL AND INFORMATION FLOW</td>
<td>45</td>
</tr>
<tr>
<td>4.1.6</td>
<td>OPERATOR AND MODEL INTERACTION VIA VARIOUS INPUT AND OUTPUT DEVICES</td>
<td>46</td>
</tr>
<tr>
<td>4.2.1</td>
<td>DISCRETE EVENT SIMULATION</td>
<td>55</td>
</tr>
<tr>
<td>4.2.2</td>
<td>MIXED VARIABLE PROCESS SIMULATION</td>
<td>56</td>
</tr>
<tr>
<td>4.2.3</td>
<td>SIMPLIFIED INFORMATION FLOW IN THE SIMULATION MODEL</td>
<td>57</td>
</tr>
<tr>
<td>4.2.4</td>
<td>INFORMATION FLOW OF MODEL</td>
<td>58</td>
</tr>
<tr>
<td>4.3.1</td>
<td>SCHEMATIC REPRESENTATION OF CRANE MODEL</td>
<td>64</td>
</tr>
<tr>
<td>4.4.1</td>
<td>SCHEMATIC REPRESENTATION OF THE REVERBERATORY FURNACE MODEL</td>
<td>70</td>
</tr>
<tr>
<td>FIGURE</td>
<td>DESCRIPTION</td>
<td>PAGE</td>
</tr>
<tr>
<td>----------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>4.5.1</td>
<td>CONVERTER MODEL SHOWING BATCH PROCESSING OF MATTE</td>
<td>76</td>
</tr>
<tr>
<td>5.4.1</td>
<td>COMPARISON OF REAL AND SIMULATED SHIFT OPERATION (TRANSFER)</td>
<td>98</td>
</tr>
<tr>
<td>5.4.2</td>
<td>COMPARISON OF REAL AND SIMULATED SHIFT OPERATION (NO TRANSFER)</td>
<td>99</td>
</tr>
<tr>
<td>5.4.3</td>
<td>SIMULATED MATTE THROUGHPUT COMPARED TO ACTUAL THROUGHPUT (MARCH 1972)</td>
<td>100</td>
</tr>
<tr>
<td>5.4.4</td>
<td>SIMULATED MATTE THROUGHPUT COMPARED TO ACTUAL THROUGHPUT (MARCH 1973)</td>
<td>101</td>
</tr>
<tr>
<td>6.3.1</td>
<td>MATTE THROUGHPUT FOR TWO DIFFERENT POLICIES (1972)</td>
<td>111</td>
</tr>
<tr>
<td>6.3.2</td>
<td>MATTE THROUGHPUT FOR TWO DIFFERENT POLICIES (1973)</td>
<td>112</td>
</tr>
<tr>
<td>6.3.3</td>
<td>COMPARISON OF PARTICULAR 24-HOUR SIMULATION RUNS FOR MATTE TRANSFER AND NO TRANSFER POLICIES</td>
<td>113</td>
</tr>
<tr>
<td>6.3.4</td>
<td>SIMULATION REPORT AT THE START OF A SIMULATION RUN</td>
<td>114</td>
</tr>
<tr>
<td>6.3.5</td>
<td>SIMULATION REPORT AFTER SIMULATING THREE 8-HOUR SHIFTS</td>
<td>115</td>
</tr>
<tr>
<td>6.3.6</td>
<td>24 HOUR RECORD OF CONVERTER ACTIVITIES FOR TWO DIFFERENT POLICIES</td>
<td>116</td>
</tr>
<tr>
<td>6.4.1</td>
<td>MATTE THROUGHPUT BEFORE AND AFTER POLICY CHANGE BASED ON AVERAGE OF TRANSFER POLICY</td>
<td>119</td>
</tr>
<tr>
<td>6.4.2</td>
<td>COPPER PRODUCTION BEFORE AND AFTER POLICY CHANGE BASED ON AVERAGE OF TRANSFER POLICY</td>
<td>120</td>
</tr>
<tr>
<td>6.4.3</td>
<td>ACTUAL MATTE THROUGHPUT IN 1973 BEFORE AND AFTER POLICY CHANGE</td>
<td>121</td>
</tr>
<tr>
<td>7.3.1</td>
<td>GENERAL FLOWCHART FOR SUBMODULE 9</td>
<td>141</td>
</tr>
<tr>
<td>FIGURE 7.4.1</td>
<td>GENERAL FLOWCHART FOR SUBMODULES ELEVEN AND TWELVE</td>
<td>PAGE 146</td>
</tr>
<tr>
<td>FIGURE A-1</td>
<td>GENERAL FLOWCHART OF SUBROUTINE GASP</td>
<td>PAGE 174</td>
</tr>
<tr>
<td>FIGURE A-2</td>
<td>DETAILED FLOWCHART OF SUBROUTINE GASP</td>
<td>PAGE 175</td>
</tr>
<tr>
<td>FIGURE A-3</td>
<td>TIME CIRCUIT LINKING DISCRETE DIGITAL TO CONTINUOUS ANALOG SIMULATION</td>
<td>PAGE 177</td>
</tr>
<tr>
<td>FIGURE A-4</td>
<td>TIME DISPLAY CIRCUIT</td>
<td>PAGE 178</td>
</tr>
<tr>
<td>FIGURE A-5</td>
<td>FLOWCHART FOR SUBROUTINE SETIME</td>
<td>PAGE 179</td>
</tr>
<tr>
<td>FIGURE A-6</td>
<td>FLOWCHART FOR SUBROUTINE TSTFL</td>
<td>PAGE 180</td>
</tr>
<tr>
<td>FIGURE A-7</td>
<td>FLOWCHART FOR SUBROUTINE RD</td>
<td>PAGE 181</td>
</tr>
<tr>
<td>FIGURE A-8</td>
<td>HYBRID COMPUTING COMPONENTS</td>
<td>PAGE 182</td>
</tr>
<tr>
<td>FIGURE B-1</td>
<td>FLOWCHART OF SUBROUTINE CRA1</td>
<td>PAGE 204</td>
</tr>
<tr>
<td>FIGURE B-2</td>
<td>FLOWCHART OF SUBROUTINE CRA2</td>
<td>PAGE 204</td>
</tr>
<tr>
<td>FIGURE B-3</td>
<td>FLOWCHART OF SUBROUTINE CRA3</td>
<td>PAGE 205</td>
</tr>
<tr>
<td>FIGURE B-4</td>
<td>ANALOG CIRCUIT FOR CRANE MODELS</td>
<td>PAGE 206</td>
</tr>
<tr>
<td>FIGURE B-5</td>
<td>FLOWCHART OF SUBROUTINE GOWT</td>
<td>PAGE 207</td>
</tr>
<tr>
<td>FIGURE B-6</td>
<td>FLOWCHART OF SUBROUTINE CFIE</td>
<td>PAGE 208</td>
</tr>
<tr>
<td>FIGURE B-7</td>
<td>FLOWCHART OF SUBROUTINE GETEV</td>
<td>PAGE 209</td>
</tr>
<tr>
<td>FIGURE B-8</td>
<td>FEATURES OF ANALOG CRANE MODELS TO PREVENT COLLISIONS</td>
<td>PAGE 210</td>
</tr>
<tr>
<td>FIGURE C-1</td>
<td>DISTRIBUTION OF MATTE TAPPING DURATION FOR MATTE TUNNEL 1</td>
<td>PAGE 232</td>
</tr>
<tr>
<td>FIGURE C-2</td>
<td>DISTRIBUTION OF MATTE TAPPING DURATION FOR MATTE TUNNEL 2</td>
<td>PAGE 233</td>
</tr>
<tr>
<td>FIGURE D-1</td>
<td>ANALOG CIRCUIT FOR CONVERTER MODELS</td>
<td>PAGE 251</td>
</tr>
<tr>
<td>TABLE</td>
<td>DESCRIPTION</td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td>4.1.1</td>
<td>CRITICAL CONTROL FUNCTIONS OF A HUMAN OPERATOR IN THE RUNNING OF THE SMELTER</td>
<td></td>
</tr>
<tr>
<td>7.3.1</td>
<td>LIST OF CODED VALUES OF VARIABLE ISET</td>
<td></td>
</tr>
<tr>
<td>7.5.1</td>
<td>RESULTS OF SIMULATED SMELTER OPERATIONS BY AUTOMATIC CONTROL</td>
<td></td>
</tr>
<tr>
<td>A-1</td>
<td>LIST OF FILE NAMES OF THE PROGRAMME AS REQUIRED FOR THE MEMORY OVERLAY</td>
<td></td>
</tr>
<tr>
<td>A-2</td>
<td>INPUT COMMANDS</td>
<td></td>
</tr>
<tr>
<td>A-3</td>
<td>DEVICE AND MATERIAL CODES FOR COMMAND INPUT</td>
<td></td>
</tr>
<tr>
<td>B-1</td>
<td>CRANE MODEL VARIABLES</td>
<td></td>
</tr>
<tr>
<td>B-2</td>
<td>CRANE EVENT VARIABLES</td>
<td></td>
</tr>
<tr>
<td>B-3</td>
<td>TYPICAL SIMULATION REPORT OF CRANE STATUS</td>
<td></td>
</tr>
<tr>
<td>B-4</td>
<td>TYPICAL SIMULATION REPORT ON EVENT FILING ARRAY AND RELATED VARIABLES</td>
<td></td>
</tr>
<tr>
<td>B-5</td>
<td>FIXED CRANE VARIABLES DURING A SIMULATION RUN</td>
<td></td>
</tr>
<tr>
<td>C-1</td>
<td>LIST OF REVERBERATORY FURNACE AND MATTE TUNNEL EVENTS</td>
<td></td>
</tr>
<tr>
<td>C-2</td>
<td>DESCRIPTION OF REVERBERATORY FURNACE VARIABLES</td>
<td></td>
</tr>
<tr>
<td>C-3</td>
<td>TYPICAL SIMULATION REPORT ON REVERBERATORY AND MATTE TUNNEL SUBMODELS</td>
<td></td>
</tr>
<tr>
<td>D-1</td>
<td>DESCRIPTION OF CONVERTER MODEL VARIABLE ARRAYS</td>
<td></td>
</tr>
<tr>
<td>D-2</td>
<td>TYPICAL SIMULATION REPORT OF CONVERTER FURNACES</td>
<td></td>
</tr>
<tr>
<td>E-1</td>
<td>TYPICAL SIMULATION REPORT OF ANODE FURNACE AND MISCELLANEOUS SUBMODELS</td>
<td></td>
</tr>
</tbody>
</table>
1.0 INTRODUCTION

1.1 PROBLEM FORMULATION

In an industrial environment where a large number of processes interact it is of particular interest to control the overall system in an optimum fashion. In many instances it is possible to control individual units, processes or subsystems in a satisfactory way but the control of an integrated system of processes presents problems. The overall control of a plant is often the responsibility of a foreman who attempts to coordinate processes to maintain a high level of production. The system controller is usually faced with irregular process behaviour, equipment failure and the general problem of achieving the best output within given constraints. This is especially true in situations where the production capacity of an originally small system was increased gradually by expanding capacities of critical processes until a rather complex system emerged. This problem can also arise when units or processes in a system are replaced by newer methods that upset the overall operation and performance. Similar problems are, however, also found in newly developed plants where certain aspects of process interaction had not or could not be accounted for in the design phase. Examples of both situations are frequent and one of each will be discussed in the text.

The object of this thesis is to develop an approach to analyse, study and improve the operation of such systems.

Mathematical models of real systems produce exact results provided there are no discrepancies between the real system behaviour and the mathematical
relationships representing it. To produce an accurate model of a complex system a simulation model will allow the best representation of the system especially in situations where exact mathematical relationships of process behaviour do not exist. A simulation approach is therefore used in this study.

In general industrial operations consist of continuous dynamic processes, i.e. processes in which the state is time dependent, and varies continuously over a period of time. In addition to the continuous processes, logical relationships exist that require actions at discrete time intervals. A hybrid simulation approach which allows accurate modelling of both discrete as well as continuous processes will be employed.

Most simulation models are off-line models which generate a simulation run independently producing a final output of statistics and other data. To include the important function of human operators in a system an interactive simulation approach will be taken. This allows the inclusion of human decisions and control functions in a model thereby coming very close to reproduce operations of a real system.
1.2 SYSTEM SELECTION

In this thesis an application of the approach of hybrid simulation of mixed
variable processes to a particular, real system is presented to show and verify the
technique.

Copper smelter operations form a dynamic system, where both discrete as well
as continuous variables govern the system state and serve as a typical example for many
industrial processes. Steel and paper making processes are other examples where this
approach is applicable. Based on preliminary results of the present work, a similar
study was started to study process control in steel making. * In general the technique
shown here is applicable to any complex system. Particular benefits are achieved when
studying systems in which some form of control is exercised by human operators.

Through the co-operation of the Control Systems and Analysis Laboratories of
National Research Council of Canada and Noranda Mines Ltd., the necessary
information was obtained on the copper smelting and refining processes of a particular
smelter for which a model was then developed.

In addition to regular operating records and other process data that were made
available by the smelter, several field studies were necessary to analyse processes and
obtain all the data required for developing and validating the model.

* Apart from crane operations every process involved and the appearance of the
overall system is entirely different to the copper smelter system. However, the
simulation technique developed here could be used with no significant modifications.
1.3 DESCRIPTION OF SYSTEM

It is important to have a clear and thorough understanding of the operation of a system to develop a realistic and accurate model.

It will therefore be useful to describe briefly the processes that are involved in copper smelting. Further details where necessary will appear in Chapter 4.

The following terms are commonly used in connection with copper smelting and refining and are listed here with a brief description:

<table>
<thead>
<tr>
<th>Fluxing ore</th>
<th>Silicon oxides (SiO₂)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Concentrates</td>
<td>Concentrated copper sulphide ores</td>
</tr>
<tr>
<td>Calcine</td>
<td>Concentrates after blending and roasting</td>
</tr>
<tr>
<td>Matte</td>
<td>Liquid ore in the reverberatory furnace (eutectic mixture of FeS and Cu₂S)</td>
</tr>
<tr>
<td>Slag</td>
<td>Liquid mixture of fayalite (2FeO·SiO₂) and gangue</td>
</tr>
<tr>
<td>White Metal</td>
<td>Almost pure liquid copper sulphide (Cu₂S)</td>
</tr>
<tr>
<td>Blister Copper</td>
<td>Almost pure liquid copper</td>
</tr>
<tr>
<td>Flue Dust</td>
<td>Precipitates rich in copper from the flue</td>
</tr>
<tr>
<td>Converter aisle</td>
<td>The aisle along which the different smelter furnaces are positioned which are serviced by cranes moving along its length (sometimes used to describe the smelter system)</td>
</tr>
<tr>
<td>Roasters</td>
<td>Roasting furnace in which excess sulphur and humidity can be removed from concentrates</td>
</tr>
<tr>
<td>Reverberatory Furnace</td>
<td>Furnace for smelting calcine or concentrated ores.</td>
</tr>
<tr>
<td>Converter Furnace</td>
<td>Pierce-Smith Converter furnace for converting matte by oxidation/reduction reactions.</td>
</tr>
</tbody>
</table>
Anode Furnace - Holding furnace for pyrometallurgical refining of blister copper.
Anode wheel - Casting machine for refined liquid copper.
Pigs - Lumps of solid blister copper cast in sand moulds.
Ladles - Large steel containers for transporting material.
Skulls - Solidified crust of matte or copper in a ladle.

(Liquid phases are given for some materials, as they occur in that form in the smelter).

Figure 1.3.1 is the floor plan of the smelter and shows the location of different furnaces. Smelting and refining of copper can be described by four consecutive pyrometallurgical processes. The first stage considered in the smelter system is that of smelting ore in one of three reverberatory furnaces. Roasting of concentrates and electrolytic refining of the copper precedes and succeeds the above four stages. These processes are not considered as they have no direct bearing on the smelter system operation that is considered here. (This is described in Chapter 4).

There are two hot charged reverberatory furnaces No's. 1 & 2 and one wet charged furnace. Whereas a hot charged furnace is fed with calcine from roasters, the wet or green charged furnace accepts concentrates at ambient temperatures with a certain moisture content.

Matte, molten material from the reverberatory furnaces, is transported in ladles by one of three overhead gantry cranes to the converters.

Both stages 2 & 3 of the smelting process take place in the converters which require frequent services by cranes during a converting cycle. Blister copper, the
final product from the converter process, is removed from converters and transferred to an anode furnace or cast into approximately 5 ton blocks called pigs on the smelter floor, designated as the pig bay. The fourth stage of refining takes place in the anode furnace. Fig. 1.3.2 shows the general material flow for the smelting process from the smelting of raw material to the production of copper.

At any time there are no less than three reverberatory furnaces, four converters, two anode furnaces in operation while the remaining converter or anode furnace is equipped with a new brick lining. The 15 circled numbers referring to material movements by cranes show the variety and number of crane services that are required.

It was not attempted to draw the complete material flow of the actual smelter as it would serve no other purpose than showing its complexity which can be appreciated. At any time only two cranes operate and are busy between 50%-90% of the time.

The crane at the north end is not used on a regular basis unless a breakdown of one of the other two occurs, or if a task of longer duration along the north end is to be performed. Its presence further south causes mutual interference and increases the danger of collisions at speeds of over 300 ft/min. with loads of molten materials at high temperatures. The crane on the north end also has a lower travelling speed which is an added disadvantage.

Activities in the converter aisle are controlled by a converter foreman who attempts to process a certain number of ladles of matte during an 8-hour shift. Operating personnel at converters, anode furnaces, cranes etc. act according to the foreman’s overall plan or schedule although detailed decisions are made by
individuals. An independent crew controls the reverberatory furnace operation attempting to supply matte uniformly to the converter aisle at the required rate.

The description of the smelter system above excludes some details that add to its complexity. Some of these will be described in Chapter 4 and the Appendices.

Nevertheless, it is evident that efficient control of the overall smelter is a difficult task. The duration of smelting processes in converters and the availability of material from the reverberatory furnaces is not deterministic, that is, no exact forecasting of endpoints of processes is possible. Invariably during several hours of production some processes operate below their capacity or are even idle at times. A buildup of crane service requirements will cause delays if the cranes cannot cope with the tasks fast enough. Under these conditions scheduling of processes is extremely difficult for a foreman, although the degree of perfection attained through many years of experience is remarkable.

The system presented is a typical situation where performance improvement through better control of the system can be studied on a model of the system.

Furthermore, the analysis can provide insight into a variety of other areas such as the effects of expansions to the smelter or the introduction of new processes. This aspect will be discussed in Chapter 6 together with other applications of the simulation model.
FIG. 2    MATERIAL FLOW CHART
2.0 LITERATURE REVIEW AND REVIEW OF THEORY

2.1 MODELLING OF COPPER SMELTER OPERATIONS

Control problems in many copper smelters have prompted several researchers to analyse and study these problems. There are numerous examples of studies directed at isolated processes of the copper smelting process, but only a limited number are reported in which the overall smelting process was studied.

Templeton and Hankley (T-1) viewed the converter aisle as a total system consisting of typically one reverberatory furnace, three converter furnaces, one anode furnace and one crane. Both discrete and continuous decision variables are included in the model and approaches to system optimisation are presented. However, for the given size of the system the problem becomes enormous and solutions for the entire problem are impossible. Sectioning of the problem produced a number of smaller sub-problems that could be solved more readily. A suggestion is made to combine the subsystems to produce results for the total system which is no longer guaranteed to be an optimal solution. Although the results of the simplified models are not directly applicable to a real situation the work represents a valuable contribution to the study of better control and optimisation procedures for the converter aisle problem.

Davies, (D-1) and Davies and Thixton (D-2) employed an empirical approach to study converter aisle operations. They report several improvements of smelter operations after implementing the results obtained from digital simulation models based on the GPSS simulation language (G-1).

Two models were developed, a macro and a micro model. While the latter
was used to simulate smelting operations over a short period of time in great detail, the
other provided also long term effects of variations of input variables such as converter
overhauls and changes to plant layout and equipment. The simulation model was based
on operating rules and statistical data from actual plant performance.

Foreman (F-1) reports of a large scale attempt of on-line process control of
a converter furnace operation with plans to extend the programme to reverberatory
combustion and overall smelter control.

At the time of reporting the on-line control of the converter, was run in
parallel to manual control, while refinement of the converter model was anticipated.
Results showed that predictions of the on-line model at the time did not match actual
operating figures close enough, suggesting that more accurate on-line measurements
would have to be obtained with the possible inclusion of further process variables.
While this approach seemed very promising the entire programme was called off after
a 10 month trial period.

The above studies are the most closely related to the work to be described
here. Far more attention in research has been given to individual processes in copper
smelting than to their integrated behaviour which is the aim of this work. When the
development of submodels for particular processes is presented in Chapter 4 a literature
review of work related to these processes will be given then.

Because of the general nature of the problem some work in other areas will
be discussed in the next section before the major difference between previous work and
this study is given.
2.2. MODELS OF OTHER INDUSTRIAL PROCESSES

The steelmaking and petrochemical industries have used computer techniques for process control extensively. Because a high degree of success has been achieved in some cases selected studies will be reported here. It would, however, be beyond the scope of this thesis to go into the detailed applications, but those studies should be mentioned that are more closely related to the work presented.

Schueger and Tiskus (5-1) have classified automatic control of steelmaking processes according to their level of sophistication. In the basic application the control is indirect and a process model provides information on the process only, leaving control to the operator. On the other end of the scale the most complex applications involve direct control of process variables to achieve optimum performance. Numerous applications of process control of Blast Furnace, Basic Oxygen furnace (L-1) and Electric Furnace operation are found with some applications in continuous casting. Probably the most significant contribution of computerised process control is in the area of hot rolling of steel. Control is exercised over the entire process from the time hot billets enter the mill to weighing the finished coils.

Although success was not overwhelming in early applications the reliability of computer controlled systems has improved steadily and their use is becoming very common. However, even in these applications only individual units are controlled and the authors predict that "the next few years will see a concerted effort to utilise the
computer to coordinate the production from many individual processes to maximise the profitability of the entire plant.

In the chemical industries some application of this kind exists, which indicates potential increases of the overall productivity of the plant from 2%–6%.

Turning away from on-site applications of industrial process control, there are several theoretical studies worth mentioning.

Ashlour and Bindingnavle (A-1) formulated a soaking pit/rolling mill complex as a queuing system and produced a discrete event simulation model. The model is able to predict improvements by increasing the capacity of the pit model, while the system performance is measured using several different criteria.

In a similar model (A-2) of an entirely different system the authors investigated job-shop scheduling of a hypothetical machine shop. Effects of different loading rules of the system performance are discussed.

Cox and Ralston (C-1) report of a shop control system implemented in a manufacturing environment with the primary objective of obtaining optimum control over the level of work in progress. A major problem, the lack of full understanding of scheduling and priority rules are pointed out which are a prerequisite for an accurate control system.

A survey of system control studies shows that it is difficult to successfully implement a control system in a real environment without pilot studies. It is for this reason that in most cases the introduction of on-line process controls is centered around particular processes and will not be extended until these control systems are perfected. The overall control of the smelter system described by Foreman (F-1) probably failed for this reason. In the steel industry individual
processes are by now understood and measurements are accurate enough that reliable control systems could be developed for them. However, the integration of these still lies ahead.

It is difficult or sometimes impossible to achieve reliable measurement of state variables under the harsh conditions encountered in practice. This difficulty combined with the lack of knowledge of some detailed process dynamics has so far prevented the development of reliable process models. In a recent report (0-1) on Operations, Analysis, Process Design and Process Control in the Mineral Recovery and Metal Production Industries, the urgent need for accurate models of processes and for reliable sensors for continuous process state measurement was stressed. The highest priority was assigned to these areas for future research.

The aim of this study will thus be the development of the most accurate possible model for the complete smelter system based only on the existing knowledge of the processes involved. In this way the problems of process measurement can be avoided.

While a simulation approach will be used it will differ from those described in Section 2.1 in three main respects:

1. The use of a hybrid simulation model which allows accurate and efficient representation of both discrete as well as continuous processes. This approach has so far not been used.

2. The inclusion of a human decision model in the overall smelter model which does not exist in the work of Davies (D-2).

3. An attempt is made to develop automatic heuristic control procedures for the model with the aim of overall system optimisation, in contrast to the approach taken by Tempelton (T-1).
2.3 MODELLING OF SYSTEMS

In the previous section reports of different approaches to analyse systems are discussed, all of which were aimed at the overall improvements or optimisation of performance.

The term "system" has so far been used freely and should be defined. In the context of an industrial environment a system can be defined as being a combination of a number of processes, some or all of which might interact. Associated with a system is a clearly defined boundary to it. Outside influences on the system are effected through variations in defined continuous or discrete decision variables. Internal variables or state variables define the status of the system at any time, that is they define the output of the system at a given time.

To distinguish between a physical system and a modelled system, which in turn is a system of its own, the modelled system will be referred to as the model. Subsections of the model will be referred to as submodels.

Careful analysis of the smelter system showed that the problem is too complex for an analytical solution. If the problem was reduced to a smaller one for which a feasible solution by techniques such as dynamic programming could be obtained, then the results would no longer be applicable to the real system. (Ref T-1) Drastic assumptions to simplify the system behaviour would on the other hand not produce a model that gives a detailed representation of the actual system.

In general the aims are to formulate a problem so that it can be solved analytically. Usually the more realistic the model becomes the more difficult
it is to obtain an analytical solution. In this particular case the difficulty of modelling processes that are not all deterministic adds to the problem of developing a realistic model. Based on studies of the smelter operations it was decided that a simulation model of the entire smelting system would be the best tool for obtaining useful information about the actual system. Usually control problems in a system originate at a particular location or a process. Attempts to solve the problem are often directed at the area where a defect or problem shows up, thereby neglecting interferences from seemingly unrelated processes. Emphasis in this work is placed on studying complete systems rather than isolating a process from a larger system which might not be the cause of the problem that is investigated.

Mathematical models may be classified in several fashions.

A model can be deterministic or stochastic. In a stochastic model of a physical system variables are random functions of time whereas by definition the value of a variable in a deterministic model will be unique at any instant.

Another classification distinguishes between static or dynamic models. In a static model the state of the system does not vary with time whereas in a dynamic model time drives the system through different states.

In dynamic models the state of the models may change continuously or at discrete time intervals leading to another means of classifying models, namely discrete or continuous models. Discrete models are represented by difference equations or by logical state transition equations, whereas continuous models are based on differential equations. The time rate of change of variables is given continuously by these relationships.

A further means of classifying models is to specify the degree of the response
of a system to changes in the input. A linear model of a system is one in which change in input produces an equivalent change in the output.

By stating the means used to derive the model behaviour a further classification into empirical or mechanistic models is possible.

The model of the smelter system developed here is a mixture of some of the idealised modes above and could be said to contain mixed variable, dynamic, semi-empirical, non-linear models.
2.4 SIMULATION

Simulation is the imitation of the behaviour of a real system through the use of a model. A more precise definition is given by Sayre and Crosson (5-6) by distinguishing it from mathematical models: "A simulation model is a symbolic (as opposed to a physical or material) representation of a phenomenon or system, yet in contradistinction to mathematical models, the symbols of simulation are not all manipulated by means of a well-formed discipline, such as algebra, the integral calculus, numerical analysis or mathematical logic." Simulation models will have many different appearances but the approach in general to develop them can be summarised as suggested by Ackoff and Sasieni (A-3):

1. Problem definition.
2. System Analysis
3. Model construction
4. Model Validation.

A stage beyond this point is the analysis of the model and implementation of results in the real system.

The nature of the smelter system required special attention in order to simulate it effectively. Usually dynamic, continuous models are easily modelled on an analog computer, while discrete processes lend themselves to be dealt with on digital computing equipment.

Shapiro (5-2) analyses the problem of choosing a simulation system and discusses the pros and cons of purely digital, purely analog and of hybrid systems.
The conclusion, that there are particular problems that fit each type of computing system is obvious. It is more difficult to recognise them as such. Svorcek (5-4) reports of a particular selection of a simulation tool for dynamic simulations. He points out the particular advantages of a hybrid system for real-time-interactive simulation studies.

By the time the second stage in developing a simulation model, the system analysis stage, is reached, a thorough understanding of the processes will allow a better decision to be made regarding the choice of a simulation system.

Hybrid simulation was chosen for this particular study and is recommended for studies of similar systems for the following reasons:

1. Possibility of real-time simulation.
2. Both discrete and continuous processes can be modelled efficiently and accurately.
3. Interactive simulation or man-machine communication is easily facilitated on a hybrid system. Input through a typewriter and output via lineprinter and a number of peripheral devices is easily facilitated. In addition special display units can be readily connected to display continuous as well as discrete variable states.
4. Parallel processes can be simulated easily on the analog computer, which is a very important aspect in modelling complex systems.
5. In addition to these points a hybrid system naturally has the advantage of not suffering from short-comings of either an
analog or a digital system if the other makes up for it. (e.g. the lack of memory in analog computation is overcome in a hybrid system - or the tedious task of initialising an analog circuit is removed in a hybrid system by automatic setup programs from the digital computer.)

Hybrid simulation certainly has its disadvantages. Program preparation and setup time is longer on the hybrid system compared to the fast use of many of the available digital computer languages.

Even more time consuming and complicated is the debugging of a hybrid computer program. It is not as easy to pinpoint a problem since processes operate in parallel, and errors can occur in the analog or digital system or in the conversion of data between them.

In this application the advantages outweighed the disadvantages simply because the only disadvantages that arose, occurred in the initial stages when the model was built, and thus occurred only once. The advantages however can be appreciated every time the simulation model is used and in part added to its successful application.
2.5 MODEL DEVELOPMENT

In the previous two sections some background on simulation and on classification of models was given.

This chapter deals with the approach taken to obtain all necessary information about the system and how the information was analysed to produce a model of the system. Whereas a wealth of literature on modelling and simulation of systems is available (S-3, M-1, A-3, F-3, M-2, C-3, M-6) it is difficult to extract from their useful guidelines on how to go about simulating such systems in practice. For this reason a short summary of the approach to develop the simulation model is given here.

The obvious first step after recognizing and defining the problem of a complex system is to start on a detailed literature review. Apart from finding out about other work done in the particular area, the review will provide insight into the processes involved and give a better appreciation of the problems that are likely to be encountered. Very useful are publications or reports on the particular system itself if these are available.

At this stage a field study of the physical system will prove very useful. It will give the analyst an opportunity to talk to people involved in the system to obtain first hand information on the operations. In trying to get some information on problem areas he will invariably get a number of suggestions as how to solve them.
If results of the simulation model are to be interpreted by other people it is important to determine what information they will require from it so that it can be provided.

With this background it will be possible to set up a preliminary model and decide on the variables that need to be included to represent the system satisfactorily. The next step is to establish how the model can be best simulated. The reasons for choosing a hybrid simulation model were given in the previous section and need not be repeated here. A very important decision, however, is the choice of a simulation language.

Problem oriented languages were developed in the past 15 years when the use of digital simulation became more predominant. The task of modeling details of systems is tedious if a procedural language such as FORTRAN is used.

Discrete event simulation languages make use of such terms as events, attributes, and files to describe quantities commonly encountered in most simulation models. The use of a simulation language is certainly recommended although it requires learning the new language. Any simulation model could be produced from first principles, that is writing a program in one of the procedural programming languages if there was no other possibility. The effort, however, is considerably reduced when using a simulation language where in some cases the completion of a check list is all the programming required. Some of the more widely used languages are GPSS (G-1), SIMSCRIPT (K-1) and GASP (P-1). These languages are referred to in most textbooks on simulation where useful comparisons are made that could aid in selecting a particular one. (S-3, M-1)
Tocher (T-1) and Buxton (B-1) discuss a number of simulation languages and compare them on several points.

The GASP (P-1) discrete event simulation language was chosen for the hybrid simulation study for basically three reasons:

1. It was available immediately.
2. It is a FORTRAN based language which can be accommodated on a small computer.
3. It is a flexible language that can be modified to suit a particular problem. (This may not be true for all other simulation languages).

More details of its operation will be given in Chapter 4 where the development of a discrete-continuous simulation system is described.

Having decided on an approach to simulate the system the field study has to be defined. All data necessary should be listed together with the means of obtaining it. This means that a good understanding of the processes involved is required by this stage.

Much of the required data can be extracted from regular operating data, but it will have to be supplemented by on site data collection if not sufficient data is available. The accumulated data should then be ordered since use will be made of it throughout the development of the model.

It is important to delay any attempts to start on the actual model building as long as possible. Early attempts tend to become drawbacks later when numerous changes have to be introduced that were not accounted for initially.
Even when the model development has started it will frequently be necessary to go back to the analysis stage if it is found that additional information is required. Developing a good model is a long process and should be carefully undertaken to pay off with reliable results.

The inclusion of human decisions into a simulation model of the kind described in Section 1.3 was given special attention, since the system analysis showed a significant effect of these decision variables on the overall performance of the system.

The model will gradually grow and change until an operational version is ready for testing. Modular design is always recommended in large models. Any corrections or additions can thereby be handled efficiently without upsetting the overall progress of the model.

The final version of the simulation model will then be tested for validity. A simulation can be considered valid if it satisfies preset conditions of accuracy, it can and should not duplicate a real system exactly unless the system is completely deterministic.

Maisel and Gnugnoli (M-1) suggest three ways in which validity can be assured.

1. Built in checks of validity in the simulation.
2. Expert reaction to simulated results.
3. Comparison of results with standards or operating records.

All three possibilities should be made use of to ensure proper validation of a simulation model. A final test, if this is possible in practice, would be to use the model to predict a given response to a change in input which can then be
verified by the same test on physical system.

All four means of checking the validity were used in the simulation model and are described in Chapter 5.
3.0 SYSTEM ANALYSIS

3.1. COPPER REFINING PROCESSES

Copper sulphide concentrates used by the smelter contain amounts of copper sulphide (Cu₂S), silicon oxide (SiO₂), iron oxide (Fe₂O₃), iron sulphide (FeS), and a certain percentage of moisture. Traces of silver and gold are valuable metals and are removed during the final electrolytic process, but play no important part in the smelter refining processes. The exact composition of the ore varies and depending on it, the grade of matte produced by the reverberatory furnaces will change. Fluxing ore, consisting essentially of silicon (SiO₂) is added to the concentrates before they are added to the wet charged furnace. (Furnace 13 on Figure 1.3.1)

The dry charged furnaces are fed with calcine, that is, concentrates with proportions of flux added that have passed through roasters. Calcine has usually got a lower sulphur content and contains no moisture. It is transferred to the hot charged furnace at elevated temperatures to conserve heat.

The ore to be smelted is entered along the two longest sides of the reverberatory furnaces and forms banks on either side leaving a gulley in the middle along which the molten ore runs to form a liquid bath at the east end. Heat is supplied by the burners in the west end of the furnace. The furnace is fired by oil and a forced air flow along the length of the furnace melts the material below it. A slag launder is situated between the burners and is used to return converter slag into the reverberatory furnace.
There are two tapping holes for removing two phases of molten ore from the furnace. The slag hole is situated in the east wall and is located higher than the matte hole on one of the long side walls also near the east side. Figure 3.1.1 gives a schematic representation of a reverberatory furnace with its most important features.

While the basic function of the reverberatory furnace is that of smelting the ore, a chemical reaction taking place to a limited degree is the same as the one described later in the matte phase of the converter, except that the amount of flux is limited to just remove all iron oxides.

\[ \text{SiO}_2 + 2\text{FeO} = 2\text{FeO} \cdot \text{SiO}_2 \]  \hspace{1cm} (3.1)

Matte, which is essentially a liquid eutectic mixture of copper and iron sulphides (CuS, FeS) from the hot charged furnaces is in general of higher copper grade than that from the wet charged furnace but in both cases the specific gravity is higher than that of reverberatory slag. Reverberatory slag contains fayalite, silicon oxide bonded to ferrous oxide, and most of the gangue and forms the upper layer of the liquid bath in the furnace. Slag floating on top of the matte can be tapped from the furnace through a hole located above the matte hole. The layers are not distinctly separated and when this region is near any tapping hole no material is removed until the bath level has changed sufficiently that only slag or matte can be tapped from the respective holes.

It is attempted to run the reverberatory furnaces at a fixed output and supply matte at constant intervals. Ladles are transported along a track in the two matte tunnels and are picked up by a crane at the converted aisle (Figure 1.3.1.)
As are all smelter processes, the reverberatory furnaces are run on a 24-hour basis with repairs to the brick linings being done continuously. This has only been possible since the introduction of the suspended roof, in which each brick is supported from a framework above the furnace.

Matte is picked up by a crane and poured into a converter. Figure 3.1.2 is a schematic diagram of a typical converter showing its principal components.

The reactions taking place during the matte phase have been a subject of considerable interest and the following reactions are considered possible:

\[
3\text{FeS} + 5\text{O}_2 = \text{Fe}_3\text{O}_4 + 3\text{SO}_2 \quad (3.2)
\]

\[
3\text{Fe}_3\text{O}_4 + \text{FeS} = 10\text{FeO} + 5\text{SO}_2 \quad (3.3)
\]

\[
2\text{FeS} + 3\text{O}_2 = 2\text{FeO} + 2\text{SO}_2 \quad (3.4)
\]

\[
4\text{Cu} + 2\text{FeS} + 3\text{O}_2 = 2\text{Cu}_2\text{S} + 2\text{FeO} \quad (3.5)
\]

\[
2\text{Fe} + \text{O}_2 = 2\text{FeO} \quad (3.6)
\]

\[
2\text{FeO} + \text{SiO}_2 = 2\text{FeO}_\cdot \text{SiO}_2 \quad (3.6)
\]

\[
6\text{FeO} + 3\text{O}_2 = 2\text{Fe}_3\text{O}_4 \quad (3.7)
\]

It is difficult to determine exactly which intermediate reactions take place to remove the iron sulphides in the slag stage but it is considered that the following two reactions represent the process accurately (N-2):

\[
2\text{FeS} + 3\text{O}_2 = 2\text{FeO} + 2\text{SO}_2 \quad (3.8)
\]

\[
2\text{FeO} + \text{SiO}_2 = 2\text{FeO}_\cdot \text{SiO}_2 \quad (3.9)
\]

Both reactions are exothermic and provide the necessary heat to sustain the reaction while cold flux (SiO\textsubscript{2}) is added. If not enough flux is present magnetite will be formed according to equation (3.7). Magnetite if returned in the converter slag to the reverberatory furnace settles on the floor causing an unwanted buildup of
material. The production of magnetite is thus avoided by adding enough flux in
the matte phase. During the matte phase several ladles of matte are poured into
the converter, and while correct proportions of flux are added, compressed air is
blown through the bath via tuyère holes. When most of the iron sulphides have been
oxidised they are removed by skimming converter slag from the bath. More matte
is added and the process is repeated until between twelve and eighteen ladles of
matte are contained in a converter. Transfer of material at this stage from one
converter to another is possible and will be discussed in the next section. Once
all iron sulphides are carefully oxidised (the converter bath temperature is an
indicator) the converter is said to go on a copper blow. In this phase the
remaining copper sulphide \((\text{Cu}_2\text{S})\) or white metal is oxidised according to:

\[ \text{Cu}_2\text{S} + \text{O}_2 = 2\text{Cu} + \text{SO}_2 \]  \hspace{1cm} (3.10)

This reaction is again exothermic and natural heat losses have to be supported by
artificial cooling to maintain a constant bath temperature. This is accomplished
by adding cold copper lumps of several tons weight to the bath at regular intervals
during the copper blow phase. Towards the end of this stage samples are taken to
indicate the end point when further blowing would oxidise the accumulated copper.

The copper, now called blister copper is then poured into ladles and
transported by cranes to one of the anode furnaces. Blister copper is about 99% pure and refining in the anode furnaces increases the purity up to 99.9%. Traces
of sulphur are oxidised or in case of an overblown charge the copper oxides that
were formed are reduced.

The refined copper is then cast on one of the two available anode wheels
and anode plates are shipped for electrolytic refining.
FIGURE 3.1.2  COPPER CONVERTER FURNACE
3.2. MATERIAL TRANSPORTATION BY CRANES

The reverberatory furnace smelting process described in the previous section is a continuous process, while material removal from it, matte and slag tapping occur at irregular intervals. Whenever a copper blow is completed in a converter a skull is dropped into the converter and the copper is removed from the furnace. The furnace is then almost completely empty and is said to start a new refining cycle. During a cycle a crane adds several ladles of matte and skims slag after the matte blowing process is completed. It has been the practice to schedule converters such that at any time matte is blown in three converters while the fourth is in stage three of the smelting process, the copper blow. One of the three converters in the matte blow will proceed with the copper blow around the time when the other copper blow is completed.

During the matte blow material from one converter can be transferred to another to maintain this schedule. This process is usually called "matte transfer", or "white metal transfer". The converter receiving transfer matte will then be ready to proceed with a copper blow sooner, while the supplying converter could be empty again to start a new cycle.

Scheduling of material transfers is critical since matte should only be transferred when it has been completely refined in a matte blow. To do this efficiently any two converters involved in a material transfer have to be ready to receive or supply transfer matte at more or less the same time to avoid any idle time. The availability of the receiving furnace can be delayed by adding more
All additions of matte, transfer matte, anode slag, reactor slag, copper pigs, flue dust, slag skulls and copper scrap to the converter and material removals of converter slag and blister copper are performed by the overhead cranes. Collar pulling on a converter is a very time consuming but necessary service also performed by a crane, when a built up collar of material has to be removed from the converter mouth.

The reverberatory furnace receives converter slag by the cranes who also remove matte from the matte cars whenever required.

The pilot plant shown on the north end of the floor plan produces certain quantities of slag and blister copper that also require a crane to be moved.

The position marked as Future Continuous Smelting Process is reserved for a new process that will be included in the model later.

Blister copper is added to the anode furnaces while anode slag is skimmed from them before the refining process commences.

Solidified crusts in ladles are loosened by a crane in bumping the ladles against a large block, and cold copper pigs often need to be moved from the pouring bays to a pig pile.

All the listed crane tasks are performed by the two cranes near the south end of the converter aisle independently or in conjunction depending on their availability and the urgency of a task. Crane services have to be scheduled carefully especially when the times of service requirements cannot be exactly determined in advance.
The task of controlling all smelter processes in an optimum way thus presents itself to the converter foreman continuously even if a smooth schedule is drawn up, any unexpected delay, breakdown or irregularity upsets the plan and calls for a new assessment of the overall shift plan.
3.3 Smelter Model

The system analysis showed that the simulation model has to provide enough detail to accurately reproduce all activities in the smelter as seen from the converter aisle. That is all material inputs and outputs from processes together with all related activities have to be provided by the model at the same rates and subject to the same constraints as the system.

Since an off-line model cannot make use of continuously measured process state variables, the model should be based only on variables that can be readily measured. It was thus necessary to obtain a satisfactory dynamic representation of the system operation based only on present knowledge of smelter processes.

In Chapter 4 individual submodels will be described. The models were designed to produce a smelter system responsive to varying material input and output rates, while moving of materials or scheduling of processes is directly controlled by the operator of the simulation model. He thus takes the place of the smelter foreman in the actual plant.

In the real system the influence of the converter foreman on the running of the smelter is very evident and any model excluding this effect could not be expected to reproduce the system satisfactorily. The facilities necessary to include the human model will be discussed in the next chapter.
4.0 SMELTER MODEL DEVELOPMENT

4.1 DEFINITION AND OVERVIEW OF THE SYSTEM MODEL

The boundaries of the system were defined and are shown in Figure 4.1.1. By excluding processes outside the boundary the assumption is made that the state of these processes has no direct bearing on the operation of the system. Their dynamic behaviour can thus be represented by input variables, while output variables have no feedback to the system.

This is in close correspondence with reality and allows a definition of the boundaries such that their presence does not affect the operation of the model.

The smelter model was divided into several submodels as they were logically provided by the system. These submodels are:

- Reverberatory Furnaces
- Converter Furnaces
- Anode Furnaces
- Cranes
- Miscellaneous Processes and Units.
- Model of the foreman (human, for which only interactive facilities have to be provided)

Before discussing these processes and their interactions the essential feedback from the different processes to the model operator should be given some attention.

A model display unit (Figure 4.1.2) was built at the National Research...
Council which provides light indicators for various process states and position indicators for the crane models. This display unit proved to be of great value for both modelling and operating the simulation model. During the validation phase of the model, operating personnel found it very easy to understand and follow the simulation process through the feedback obtained from the display.

It consists of three servo-driven motors that position a pointer (crane) according to a voltage supplied by the analog model of the cranes. On the pointer, two light indicators provide information on the state of the crane. The rest of the display resembles the floor plan of the smelter where seven light indicators of different colour show the state of the modelled processes. Whereas the crane position is derived directly from the analog models of the cranes, the light indicators are set in the digital submodels of respective processes and serve as coded signals. By calling a subroutine LIGHT with the corresponding light number as an argument, lights are set and reset through signals from control lines. Figure 4.1.3 shows the information that each indicator provides on the state of the three types of furnaces.

A speedup factor of 20:1 was chosen for simulation and the display unit is scaled down 500:3 to allow the fastest possible simulation of the smelter which still allows the operator to follow the operations. A line printer, stripchart recorders and digital voltmeters provide all the remaining information on the model. Figure 4.1.4 shows the EAI 680 analog computer with the model display unit connected to it.

Fig. 4.1.1 only shows the fundamental processes that have to be included in the model. It does not show the individual units, and how these interact.
function of the foreman is also not included. The model that was developed for the entire smelter system is presented, again simplified, on Fig. 4.1.5. It shows the function of each of the six major submodels and how these interact. It should be pointed out that material and information flow exists between all units of each type of process. This complicates the function of the operator and leaves wide room for error and sub-optimal operation.

To be able to fulfill his control function efficiently the foreman has to obtain complete and up to date information on the progress of all units in the system as a basis on which to formulate his decisions and issue commands.

Table 4.1.1 lists the type of actions he takes if alerted by any condition in the model. Some of the actions are simple but all require planning and decision making so as to ensure an efficient operation of the overall system. So far the discussion of the model has been divorced from the implementation. The need for efficient information display of the model will, however, be related to some extent to the particular implementation. Fig. 4.1.6 shows the peripheral devices that were used to relay information from the model to the operator. Also shown are facilities that allow on-line control of the model. All of the triggers and actions listed on Table 4.1.1 are communicated by the outputs and inputs as shown.

All output signals but one are of visual form. A bell ringing on the typewriter is, however, very successfully used to alarm the operator when an important action is required and in principle any other device that can be interpreted by human senses could be used.

The display unit described above is the central output device which will be observed almost continuously by an operator. Some of the processes will be
represented by digital models, others by partly digital and partly analog models.

Analog variables if given by stripcharts, voltmeters or servo driven motors will show continuous variations. Output from digital models will change at discrete intervals when state variables and statistics are updated. At this time the coded light signals will change if the corresponding process reaches or leaves a given state. Information on any process is only provided if requested except for messages on the typewriter to alert the operator or request action from him.

As a reference to Chapter 7 where a completely automatic control of the model is described Fig. 4.1.5 also shows that the control function of the operator could be replaced by a non human control program.

Following the description of the overall model and the particular attention given to the operator submodel the next section will deal with the development of the hybrid simulation mechanism which produced a realistic operating model. The last sections in this chapter are then devoted to the development of the 5 major models and their implementation.

Because of the size of the problem not all submodels were developed by the author. The crane submodels which were to be mixed variable models, and the reverberatory furnace and matte tunnel submodels, which were to be represented by purely digital discrete event type models, are representative models for the system and were developed by the author. The mixed variable converter furnace submodel was developed by L.K. Nenonen of the N.R.C. who has had valuable previous experience in the converter control studies.
The anode furnace and miscellaneous process submodels were to be discrete event type models and were developed by Dr. U.P. Graefe also of the N.R.C. Dr. Graefe also provided a subprogram that decodes the input commands to the model and translates these into corresponding events.

After describing the overall operation of the simulation a brief description of each submodel will follow with further details on each presented in the Appendices.
FIGURE 4.1.1 COPPER SMELTING PROCESSES WITHIN SYSTEM BOUNDARY
Figure 4.1.2  Smelter Display Unit
CONVERTERS

1. Matte stage
2. Copper stage
3. At an end point, not blowing
4. Material to be added
5. Empty ladle required
6. Material to be removed

REVERBERATORY FURNACES AND MATTE TUNNELS

1. High slag level
2. High matte level
3. Matte available
4. Matte on matte car
5. Empty ladle needed
6. Full ladle of matte on floor

ANODE FURNACES

1. Blister copper required
2. Full, ready for refining
3. Refined, ready for tapping

* indicator lights

FIG. 4.1.3 SMELTER DISPLAY UNIT: INFORMATION FORMAT
FIGURE 4.1.5 Representation of Overall Model showing Material and Information Flow
FIGURE 4.1.6  Operator and Model Interaction via various Input and Output Devices
<table>
<thead>
<tr>
<th>Trigger</th>
<th>Response</th>
<th>Critical Decision</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arrival of matte at matte tunnel</td>
<td>direct to a converter</td>
<td>Converter, crane</td>
</tr>
<tr>
<td>End of matte blow in converter</td>
<td>cause skimming, new matte addition</td>
<td>Crane</td>
</tr>
<tr>
<td>Start of copper blow in converter</td>
<td>cause addition of flue dust</td>
<td>Crane</td>
</tr>
<tr>
<td>End of copper blow in converter</td>
<td>cause addition of skull, skimming of copper</td>
<td>Crane, anode furnace</td>
</tr>
<tr>
<td>Stock of skulls low</td>
<td>cause ladle bumping</td>
<td>Crane</td>
</tr>
<tr>
<td>Cu-pig required by converter</td>
<td>direct pig to converter</td>
<td>Crane</td>
</tr>
<tr>
<td>Stock of pigs low</td>
<td>plan for casting of pigs</td>
<td>Crane</td>
</tr>
<tr>
<td>Matte required by one or more</td>
<td>plan distribution of matte as it becomes available</td>
<td>Converters</td>
</tr>
<tr>
<td>Material available at pilot or prototype plant</td>
<td>direct to a converter or pig pile</td>
<td>Converter, crane</td>
</tr>
<tr>
<td>Anode furnace to start refining when full, or when decided by operator</td>
<td>cause skimming of anode slag then cause start of refining</td>
<td>Crane</td>
</tr>
<tr>
<td>Anode furnace completed refining</td>
<td>cause start of casting to fit scheduling of anode wheels</td>
<td>Crane</td>
</tr>
<tr>
<td>Converter requires collar pulling</td>
<td>cause collar pulling</td>
<td>Crane</td>
</tr>
<tr>
<td>Crane requires maintenance</td>
<td>cause start of crane service</td>
<td>Crane</td>
</tr>
<tr>
<td>Converter is to be stopped</td>
<td>cause interruption of converter operation</td>
<td>Crane</td>
</tr>
</tbody>
</table>

**TABLE 4.1.1 CRITICAL CONTROL FUNCTIONS OF A HUMAN OPERATOR IN THE RUNNING OF THE SMELTER**
4.2 CONTROL OF THE SIMULATION MODEL

The reasons for choosing a hybrid computer to perform the simulation of the smelter system were given on Section 2.4 and the decision to use the GASP discrete event simulation language is presented in Section 2.5.

In this section an overview of the hybrid simulation approach to represent the smelter model is given. While some parts of the implementation will be discussed, details of the simulation control appear in Appendix A.

As all simulation languages GASP (P-1) deals with a few defined quantities that allow a simple representation of a real system.

Events are considered to be the operations and/or processes taking place in the system at discrete intervals of time, since simulation is performed at discrete steps and not continuously on a digital computing machine. At these intervals the state of the system will change. Characteristics of an event are given by attributes associated with it. (i.e. the duration, height and comfort of a flight are considered attributes of a flight, if these characteristics were important to a customer in deciding to fly on a certain airline.)

An entity represents an object or person in a system which acts or is acted upon during a simulation. An entity is also characterised by attributes. The following examples show how a simulation is performed by moving a system from one state to another.

A ladle arrives (event) at a certain time (attribute of the event) at the
converter aisle (attribute of the event). If a crane (entity) is available (attribute of crane) and a converter (entity) requires material (attribute of converter) then the transfer (event) will be performed immediately (attribute of the event).

This example could be continued to perform all activities in the converter aisle and would represent a purely discrete simulation. Since the crane submodels are analog, continuous models, certain modifications are necessary that will be described later.

Events can have any number of different attributes, only limited by programming requirements. Events are contained in sets or files; the latter name will be used here and will refer to the area in the computer memory where a group of events are stored. The number of files are determined by the system and in each file at least one attribute is to be of the same kind, usually the first.

There is always one time file, the master file which contains all "time events" ordered chronologically. The controlling programme GASP removes and executes time events sequentially while updating the simulation time. When an event is removed from the time-file, the event code characterises the type of event it represents and execution is performed by branching to a respective subroutine or section of the programme.

After initialisation at least one time event has to exist in the time file. It will be removed by the controlling programme and executed. The execution of events can create further events which are filed in the time file. These are endogenous events in contrast to exogenous events such as commands from the model operator which will be discussed in the next section. As long as there are events in the time file the simulation continues until any preset condition (number of events, elapsed time, etc.)
for termination is reached.

In this work the basic structure of the event filing system of the GASP-
simulation language was used. It required modifications to produce an integrated
mixed variable simulation language capable of handling both discrete event-type as
well as purely analog models.

On a hybrid computer there are three modes of operation of the analog
computer, INITIAL CONDITION, HOLD and OPERATE. The OPERATE-mode refers
to the state of the analog computer when differential and logical equations are being
solved. The HOLD-mode refers to a state where all analog values are frozen. The
IC-mode is selected to set up circuits before proceeding to the OPERATE mode for
execution.

The modes of operation can be selected from the digital computer which
provides a means of synchronising digital and analog simulation models on the hybrid
computer. While discrete event simulation is not a form of real time simulation, the
analog computer solves equations at a uniform rate given by the selected time scale. This
rate determines the rate of simulation. To ensure that discrete events are not executed
consecutively while instantly updating the simulation time, a timing circuit was developed.
It is shown in Appendix A (Fig. A-3) where the detailed operation is described.

The link between the timing circuit and the digital simulation control
program is the output of comparator C39. Whenever a time event is executed the one
comparator input is set to the scaled time interval to the next time event, TREL. This occurs
in the HOLD-mode when other analog models might be given new input or target
conditions. When placed in the OPERATE mode the integrator A35 will operate until
the time to the next event has elapsed when the comparator C39 changes its
logic state which will be sensed by the digital control program. During any OPERATE
period the control programme continuously samples logic outputs of the timing circuit
and other analog circuits. The simulation is then frozen by selecting the HOLD mode
and the event is executed after the scaled time interval. The next time event is then
selected, the timing circuit having been reset and this process repeats itself.

As mentioned, during the OPERATE mode analog circuits are solved while
some of their logic outputs are continuously tested by the simulation control programme.
This is the case if any target (e.g. a crane reaching and waiting at a location as
described in the next section) of an analog model is reached. Again the simulation is frozen
by selecting the HOLD mode to reset the particular model or schedule an endogenous
event. If an endogenous event is scheduled to occur before the previous next event
the target of the timing circuit is reset. This condition can also arise when an exogenous
event, issued from an outside source is to occur before the previous next time event.
If the HOLD mode is not caused by the timing circuit itself the intermediate simulation
time is provided by the previous event time to which the rescaled value of the timing
integrator C35 is added. It can be obtained by reading the output of an analog to
digital converter.

While more detail can be found in Appendix A, Fig. 4.2.2 gives a simplified
representation of the process of discrete-continuous event simulation on a hybrid computer.

It allows for the parallel operation of several analog models which means
that the simulation control program not only executes the time events but also triggers
the start and registers the endpoints of analog models. Because these other models
can create new time events (as can be done by manual input) at times other than during
the execution of time events, it is necessary to test for new events and reset the timing
circuit whenever such a condition arises. This is done by two subroutines as explained in Appendix A, where the controlling program GASP is also flowcharted and explained in more detail.

The need for interactive control of the simulation model was pointed out in the previous section. An input instruction will either cause a model to change immediately or else it can be an event type instruction which will be placed in a file to be executed at a later time.

To use the example given earlier, the arrival of a ladle filled with material will alert the operator by a light signal. He can then direct the ladle to a destination that he selects by issuing an appropriate command to a crane. In this way human decisions control certain areas of the simulation, in the same way as a foreman controls processes and operations in the smelter.

To get the controlling program to accept and interpret an input from the typewriter a sense switch is also continuously sampled by the program in the OPERATE mode. When pressing the switch, the HOLD-mode is selected and a command translation subprogram accepts and processes any typed input. This is explained in more detail in Appendix A and was also omitted for simplification on Figure 4.2.2.

Fig. 4.2.3 attempts to show the information flow in the simulation model. Because of the two operating modes HOLD and OPERATE it is difficult to relate it to Figure 4.2.2. It might be helpful to remember that any execution in the digital section of the program (including inputs to analog models) occurs when the analog section is in the HOLD-mode. During the OPERATE mode the only activity in the digital section is that of sampling logic signals of analog models and of the sense switch. (This is not shown on Fig. 4.2.3).
The information flow does therefore not represent one path through different stations, at any time, but it rather shows several paths as they can occur at different times during simulation. The following examples relating to Fig. 4.2.3 might help to gain an understanding of the operation of the model:

1. At the start of simulation a programme section is executed which initialises all analog and as digital models, reads data cards and as well as initialising simulation variables. Control is then turned to SIMULATION CONTROL programme and simulation as shown in Fig. 4.2.2 commences.

2. If the operator presses a sense switch (No. 1) the statement ENTER appears on the typewriter. He can respond by entering a command (Format is given in Appendix A) which could be one of three types. If it is not filed in one of the files an event is executed directly which might affect any digital or analog model. (A process could be triggered to start in which case light signals might be reset i.e. information flow all the way to the display unit).

3. During any simulation run various statistics are continually created and updated for all processes of the model. These are discussed in the next sections and can be requested at any time on a single basis or as a summary of the simulation status of the entire model. A REPORT GENERATING programme produces this typeout which can be seen on Figure 6.3.4 in Chapter 6 where results of the model are presented.

4. By entering " simulation continues and control goes to the SIMULATION CONTROL section. Events are removed from files and executed. In
the case of the crane models this involves setting inputs and targets of analog models which will cause physical crane movements that can be observed on the display unit during the OPERATE mode.

5. By pressing another sense switch (No. 7) the area marked as AUTOMATIC CONTROL is activated. Its operation is explained in Chapter 7 and included here only to show that it fulfills the function of the human operator by generating internally whatever inputs might otherwise have been provided to the model.

Figure 4.2.4 shows the implementation of the simulation model on the EAI690 Hybrid System at the National Research Council of Canada.

The core requirement of about 50K memory locations for the entire model, of which about 2K are variables in DIMENSION or COMMON statements, was well in excess of the 16K provided by the EAI640 digital system. A primary overlay system was used to execute the program of the model which was split into twelve submodules. It utilizes a magnetic disc for storing sections of the program not required in core at any time during execution. This overlay system was developed by Dr. U.P. Graefe. The event structure lent itself well to split up the programme of the model in sections. No cross-reference is necessary between the submodels and the constraint that overlay modules are to be independent of each other could thus be met.
Figure 4.2.2 Mixed Variable Process Simulation
FIGURE 4.2.3  Simplified Information Flow in the Simulation Model
4.3 CRANE SUBMODELS

The main tasks performed by any one of the three overhead cranes are those of transporting ladles or other items from one point to another. Less frequently cranes are being used to "pull a collar" from a converter furnace, when by means of a special pulling hook the built-up material around the converter mouth is removed. Other tasks are the bumping of a ladle against a block to break the solidified crust inside a ladle called skull, or a floor cleaning operation by means of a shovel.

All crane operations can be reduced to two simple tasks, one of moving from one destination to another, and one of being occupied for a certain length of time at a fixed position.

Furthermore each crane executes its tasks independently as long as its occupation at any time does not conflict with that of another crane. This conflict is possible since all three cranes use the same tracks and cannot pass each other. If such a conflict occurs, i.e. if one is to move south and the other north at the same time, then preference is given to the task of higher importance or urgency. It is also possible that a sequence of tasks is performed in conjunction by two cranes.

Regular maintenance times are scheduled weekly during which time the cranes are inoperative. This does not eliminate unexpected breakdowns of varying length which still occur at irregular intervals.

In many digital simulation models of material handling systems the activity of cranes has to be accounted for. The following studies (L-2, A-1, D-2, T-1) included overhead cranes as submodels of larger discrete event simulation models.
In general the state of cranes is described by several characteristics such as:

- availability \( x_1 \)
- position \( x_2 \)
- duration of present task \( x_3 \)
- priority of present task \( x_4 \)
- \( n \)th description of state \( x_n \)

If these characteristics are combined they represent the state of the crane model:

\[
\text{state of model} = f(x_1, x_2, x_3 \ldots x_n)
\]

During the execution of a crane event the state will change according to the task that is performed, the nature of the state change being dependent on the states of all other cranes and processes.

For each problem the equations governing the transformation of state are different depending on the assumptions and rules according to which a sequence of events are to occur.

In this study the following state variables are used to describe a crane activity uniquely at any time. The characteristics are represented by variables, which are shown in Appendix B

- present position
- previous position
- present waiting time
- previous waiting time
priority of present task
priority of previous task
rank of last event in crane file
rank of present crane task
event triggered by crane
quantity of material on crane
operational state of crane
availability of crane
state of crane activity (delay)
state of task sequencing (waiting)
condition of crane (maintenance)

In the actual system a crane operator receives the order to perform a sequence of tasks which he will then execute in that order, at the end of which he will have received further instructions or else be idle. The functions to be performed by a crane model are shown on Figure 4.3.1. The model consists of a logical part representing the crane operator and a mechanical part representing the crane which executes the physical task. Below follows a description of the implementation of the crane models which represents the actual process very accurately. Crane commands are issued by the model operator and are placed sequentially in one of three crane files according to a ranking number. Execution of tasks by the crane models occurs according to the first-in-first out rule and tasks are executed continuously until no tasks remain in the respective crane file when an busy indicator light on the display unit goes off. As long as no breakdown occurs an operational light signal will light up.

Each crane model consists of a digital and an analog section which executes tasks by causing corresponding changes in other submodels. (i.e. addition of material
removal of material, etc.). The circuit diagram of the analog and flowcharts of
the digital models are given in Appendix B.

The controlling digital program GASP exercises control over all crane tasks.

The simulation is in the HOLD mode while a task is removed from the file
and the respective digital crane-submodel routine is called.

In this section of the crane model the variables that make up a crane command
are used to start the execution of the corresponding task. Interferences, priorities,
duration of the task etc. are checked to establish whether in fact the required task
can be executed immediately. If this is not the case the task is delayed while the
crane remains idle. If execution of the task is possible the analog model is set up
accordingly, that is the position of the target, direction of travel, speed of travel
and the waiting time at the destination are set on the analog components by suitable
calling programs. If another crane causes an interference and has to be moved out
of the way, then the analog and digital models of the interfering crane are set to
respond accordingly. Thereafter the controlling program returns the analog computer
to the OPERATE mode and the crane will move to its destination and perform its task
by waiting for the duration of time required for the particular task.

Time studies for all typical operations provided average times for these tasks.
A sense line will be reset at the end of the waiting period at the destination and the
computer is returned to the HOLD mode.

To complete the crane task it might in some cases be necessary to cause
further action at that time, i.e. cause another event to be executed. An example
where no such action is required would be the moving of an empty ladle from one
point to another. If however a full ladle was transported from one furnace and
poured into a converter this task would trigger the event "material was filled into converter," which will update the state of the converter. The code for this event is checked and if non-zero the respective event is executed by the controlling program. This completes one typical crane task and the next task will thereafter be executed in a similar fashion.

The active tasks of moving and waiting at a position of the cranes are thus represented by analog models. Since these can operate in parallel the execution of crane activities is very close to the real operation. During the non-operating periods when submodels are updated or new tasks selected the simulation time is frozen and does affect the crane models.

To allow for the inoperative periods in the crane models, statistics of breakdown frequencies and repair times are used. The digital models generate these periods by random sampling from frequency distributions. If a breakdown occurs the "operational" light indicator will be turned off and the crane accepts and executes no tasks until a repair period has elapsed. The same applies if the preventative maintenance is performed.

To prevent the bumping of the two faster cranes into the slower crane a delay mechanism is included in the analog models of the faster cranes that will prevent this from happening. In Chapter 7 it will be shown how a series of synchronised tasks can be performed by any two cranes. This feature allows one crane to operate in conjunction with another, rather than independently.

By issuing several commands to the cranes it is thus possible to direct material flow and thus the operations in the smelter in any desired fashion. Tasks are executed in parallel by the cranes and their movements and operational states can be observed on the model display unit.
4.4. REVERBERATORY FURNACE SUBMODELS

A schematic representation of the reverberatory furnace is given in Figure 3.1.1. In Chapter 3 the process of dry and wet reverberatory smelting was described briefly. Anderson (A-4, A-5) describes the operating practice at Noranda in detail and shows how the capacity was increased over the years.

Although production figures to day are different, the basic operations have remained the same. (e.g. change from firing with pulverised coal to oil.) While no theoretical analysis is given these two papers present a good background on reverberatory furnace operation indicating the effects many variables have on the furnace operation. McKerrow (M-3) describes a similar smelter operation at Gaspé.

Harris (H-1) has developed a mathematical model for the heat transfer mechanism in a reverberatory furnace. The model is based on an existing furnace but is idealised by simplifying assumptions. It was used to investigate the smelting capabilities of a particular reverberatory furnace and several recommendations were made for improvements. The model did not however account for metallurgical aspects of smelting which are expected to be affected by varying the smelting conditions.

While the work described by Harris provides insight into the thermal response of a furnace, the model does not represent the entire smelting operation in the reverberatory furnace. In particular the problems encountered in controlling a side charged furnace by temperature measurements are difficult. Pyrometer measurements taken at different points along the furnace read the bank temperature
which fluctuates during charging and is not a representative quantity by itself.

Allen (A-6) et al. present a preliminary report on the development of empirical mathematical models expressing aspects of reverberatory furnace operation. The aim of the study was to determine all variables affecting the performance of the furnace. It was hoped to improve the operation by eliminating fluctuations of those variables that affect the optimum smelting conditions. In contrast to the mechanistic model of Harris, Allen et al. produced a multiple regression model based on data of 86 operating days. Because several variables proved to be dependent a non-linear model was used to predict the smelting rate. (The equation for the smelting rate alone fills 12 typed pages.) Predictions of this model were close to the actual performance of the furnace, but simpler, linear models failed to produce satisfactory results.

Until a final report on this work provides evidence to the contrary it appears that accurate reverberatory furnace models have to be based on the large number of interdependent variables involved. Such a model would be very impractical for day-to-day operation because of the effort involved in measuring all variables frequently and accurately. Furthermore their evaluation necessitates a digital computing machine. Such a model would probably be useful only for a particular furnace while any other furnace might behave quite differently.

In this study it was attempted to limit control variables only to those that are available on a regular basis at the smelter. Since inputs and outputs of materials are the only quantities that directly affect the operation of the converter aisle, it was possible to develop a relatively simple empirical model based on material balances.
There are two important aspects of the reverberatory furnace that have to be represented by the model. Firstly, matte of a given grade of copper is supplied by the furnace. Since this is the primary source of material to the converter aisle it has an important influence on the overall performance of the smelter and had to be modelled as accurately as possible. Secondly the reverberatory furnaces receive slag from the converter aisle. This is a more flexible but necessary process in the smelter. Figure 4.4.1 shows the model of the reverberatory which supplies matte and produces slag. The rate equations shown are developed in Appendix C.

The reverberatory furnaces models are implemented on a purely digital basis. Since both the rate equations for slag and matte production are uniform and since the availability of matte to the converters is given by a stochastic variable, there was no need for an analog model. All three furnaces are treated in a similar fashion. A number of discrete events were programmed that simulate the behaviour of the model if they are executed in a specific sequence.

Typically the event of an arrival of an empty matte ladle will trigger an event which indicates when the ladle will have been filled again. If any event that requires an operator response occurs there will be a feedback to the operator informing him of the occurrence of such an event. This is either done by a light signal on the display board or by a typed message on the teletype writer or the line printer. The meaning of the light indicators is given on Figure 4.1.3. Light (4) indicates that matte is available, and when picked up light (5) will come on while light (4) is turned off. Light (5) is turned off only when a crane is instructed to
deposit an empty ladle at the matte tunnel. If no converter was ready to receive matte, the ladle can be placed on the floor when light (6) is turned on. Lights (1) to (3) give information on the material levels in each furnace.

When no operator or other action is necessary during a time span it is possible to bring a sequence of minor events together into one larger event. This event will cover a longer period of elapsed simulation time and economises on space and time in the programme. A small example will explain this:

1. The matte car starts moving to the tapping hole with an empty ladle
2. The matte car arrives at the tapping hole
3. Matte tapping starts
4. Matte tapping ends
5. The matte car starts moving to the converter aisle
6. The matte car arrives at the converter aisle with a filled ladle.

These six events can be combined into one event, namely event (6) which is scheduled at the time that event (1) is just about to occur. It would not be possible to combine these events if it is considered necessary for example to specify the time that matte tapping starts.

The availability of matte and slag in the furnace are the two variables that determine the input and output of the model. The availability of material is represented by the level instead of a volume measurement to conform with the actual readings in the smelter. These readings are taken at intervals during smelter operation and provide a measurement for controlling the smelting rate.

It would go too far to point out all the implications involved here, but it should be mentioned that these readings are difficult to take simply because the
separation of the slag and matte layers is not distinct. Furthermore the bath is surrounded partly by the charge banks causing the size of the bath to vary. Estimation of the total quantity of molten material in the furnace is therefore subject to some error.

All assumptions that were made together with a detailed description of all events and variables used to model the reverberatory furnaces are given in Appendix C.
FIGURE 4.4.1 Schematic Representation of the Reverberatory Furnace Model
4.5 CONVERTER FURNACE SUBMODELS

The converting process was discussed in Section 3.1 where its functions of converting matte (Cu$_2$S, FeS) to white metal (Cu$_2$S) by oxidising the iron and sulphur was described. The latter product escapes as SO$_2$ gas while the iron is removed in the slag.

Mathematical modelling of the converter process is reported by Dudgeon, Nenonen et al. (D-3), Niemi (N-3) and Nenonen (N-2). In each case models based on heat and material balances are presented. Krause et al. (K-2) have proposed a theoretical model to control the converter process by an on-line computer but describe its implementation as economically not feasible. Foreman (F-2) has reported the implementation of a similar control study based on a mathematical model of the converter processes, which did however not prove to be reliable enough for continuous on-line control because of process measurement problems.

Converter models can be based on heat and material balances to represent the chemical process of converting. For the purpose of this study the converter submodels should represent the chemical process of converting in terms of reaction rates and material inputs and outputs. This means that the model should respond to controllable variables such as matte additions, slag removals and provide end points of blowing periods based on heat and material balances. The functions of the converter model are shown on Figure 4.5.1. The reaction rate equations shown are developed in Appendix D.
The converter submodel used in the smelter model is a mixed variable model having analog as well as digital content. It was developed by L.K. Nenonen of the National Research Council.

Only a brief description of the implementation will be given here, a detailed description of all functions being given in Appendix D.

There are five converters each being represented by the same model. There are thus five identical analog models and a number of digital events representing the behaviour of converters. At any time only four of the five converters are operational, the other being down for repairs to the brick lining.

Converters are represented on the display unit by six light indicators for each of the five furnaces. The functions of the indicators are given by Figure 4.1.3.

Based on the information relayed by the indicators the model operator can control the converting process whenever his action is required.

Referring to Figure 4.1.3 light indicator (1) henceforth referred to as light, signifies that the converter is in a matte stage if the light is on. The model allows a certain number of matte ladles to be added to the converter at any time during one of eight possible matte blows. Light (4) is on as long as more matte can be added. Once the limit is reached light (4) is off while (1) stays on. In addition to the light colours of white, green and red, there are information tags as shown which remind the operator of the indicator meaning. At the beginning of the first blowing period a minimum quantity of matte has to be added to the converter before it can start the process of converting which is termed "blowing", when air is blown through the metal bath. Once the critical limit of matte is reached the analog model representing the process dynamics of the converter will start the converting process,
and light (3) indicating the end of a blowing period is turned off. At the end of a matte blow the slag that was produced has to be removed and returned to the reverberatory furnaces by a crane. This state is indicated by light (6) at which time the converting process stops, indicated by light (3). To remove the top layer of slag from the metal bath, to skim slag, an empty ladle is placed below the furnace which turns and pours material into the ladle. If such a ladle is required light (5) is on.

At the end of a matte stage the model operator will instruct a crane to pick up a ladle of slag, pour it into a reverberatory furnace and return the empty ladle to the converter. The same command is repeated until all the slag is removed which will be indicated by light (6) being turned off while light (4) comes on, i.e. more matte can be added to proceed with the next matte blow.

When discussing the crane models it was pointed out that at the end of the waiting time at the target the respective command is executed to update the model affected by the operation.

In this particular case the task of skimming a ladle of matte will trigger the event "matte has been skimmed from converter No. 1".

In executing this converter event the quantity of slag removed will be deducted from the variable representing slag and any other operation regarding the converter model will be performed then. Similar events exist for every other addition or removal of material.

After adding one ladle of matte to the converter in any blow other than the first, light (3) is turned off and the converter blows until a given fraction of the available iron in the bath is slagged off. Matte can, but does not have to be added until light (4) is turned off. As are all other submodels the converter
models are protected from incorrect commands such as material additions beyond the allowable limit. Commands are illegal if the type of command is not compatible with one of 15 possible converter states and is simply ignored.

Whenever the copper content in a converter reaches a given minimum quantity after a material addition, a message on the typewriter gives the operator the option to end the particular last matte blow (the going-high blow during which a higher bath temperature is obtained) and proceed with a copper blow. Material removal from a converter is allowable at the end of any matte blow after skimming slag. The converter receiving the transfer-matte can be in the same state or blowing in a matte stage.

Slag can be skimmed during the going high blow which is indicated by light (6) without reaching the end of a matte stage which is indicated by light (3). Skimming of slag, once the end point is reached, is performed as in other matte blows except that the last ladle of slag is poured in another converter in the matte blow. This is because of the high percentage of copper in the slag.

Following this, lights (6) and (1) will be turned off and lights (2) and (4) turned on. At this point the model requires the addition of flue dust before the end point light (3) is turned off when blowing in the copper stage commences.

During the copper blow the converter model requests addition of cold copper lumps (pigs), indicated by light (4). At the end of a copper blow (light (3) on), a skull has to be added (light (4) on) before the blister copper produced can be removed (light (6) on). After the copper is removed and transferred to an anode furnace or poured in the pig bay a new converter cycle starts.

Other than the materials already mentioned copper scrap and reactor slag can be added when available to converters in the matte stage.
The response and demands of the converter models thus correspond very closely to the actual converter operation while the duration of blowing periods are provided by the process dynamics in the analog converter model. After addition or removal of material the analog model is reset to process the given quantity in the furnace at a rate determined by the rate of air flow through the converter bath.

Strip chart recorders are used to show the quantity of FeS in the furnace during the matte stage and Cu₂S during the copper stage. Examples of these will be shown in Chapter 6.

End points of blowing periods are relayed to the digital computer model in the same way as the end points of crane tasks namely by the change of the logic signal of a comparator.

Digital models provide converter breakdown events and indicate the need for collar pulling after a given quantity of material has been blown.
FIGURE 4.5.1 Converter Model showing Batch Processing of Matte

MATTE STAGE

2FeS + 3O₂ + SiO₂ → 2FeO·SiO₂ + 2SO₂

Cu₂S - content

BREAKDOWN OR MAINTENANCE

COPPER STAGE

Cu₂S + O₂ → 2Cu + SO₂

Flue dust addition

Cu₂S - content

Cu - content

Copper Skimming

Flue Dust, Pigs, Skulls

Cu-pig addition

Skull Addition

Time
4.6 ANODE FURNACE SUBMODELS

The final process of refining blister copper in the anode furnace is one of oxidation or reduction. This is accomplished by blowing either air or natural gas through the anode furnace, which is a large cylindrical vessel, holding 15 or more ladles of blister copper. There are three anode furnaces, one of which (#4) employs the method of poling when refining. Batches of blister copper from converters are poured into the furnace until it is full or near full. A refining period is preceded by skimming of slag from the furnace. The decision to start refining before the furnace is filled can be made, which is usually only done if a long delay is expected before a further batch of copper becomes available.

Refined copper flows out of the back of the furnace and is cast into anode plates on one of two anode wheels. Some time after the end of casting the furnace is again ready to accept copper. A short description of the implementation follows.

There are three light indicators on the display unit for each anode furnace; the state each one represents is given by Figure 4.1.3.

The anode furnace models request addition of copper until they are filled. After every addition respective events update the quantity of copper. No action will take place once a furnace is filled until a command for removing anode slag and one to start refining is given. These commands can also be issued before the furnace is filled to capacity. The end of the refining process is indicated by the "ready" indicator which will be turned off when a command to start casting is issued. Some time after casting is completed the indicator "copper can be added"
lights up again, and a new refining and casting cycle begins. The duration of the refining and casting processes is given by the furnace content and uniformly distributed rates for each process. Again illegal commands occur if commands concerning the anode furnace are not compatible with its state and are ignored.

Further details on this submodel can be found in Appendix E.
4.7. MISCELLANEOUS SUBMODELS

Apart from the three main furnaces and the overhead cranes for material handling there are several other units, processes and locations that have to be accounted for in the model. These models are purely digital models and are briefly described below. Further details can be found in Appendix E.

Pilot Plant Submodel

Located at the north end of the converter aisle is a pilot plant for a continuous copper smelting process. It produces so-called reactor slag and copper cakes which are lumps of copper cast in molds. Slag is picked up by a crane and poured into a converter in the matte blowing stage. Copper cakes are used for the same purpose as pigs are in the copper stage of converters, for cooling purposes.

The model utilises three light signals to indicate the availability of slag, copper cakes and the need for an empty ladle. Slag and cakes are produced in the model at varying intervals, and when available an appropriate command to a crane will allow the removal.

Prototype Plant

This process is shown as a future continuous smelting unit on the floor plan of the smelter, Figure 1.3.1. As the name suggests it is an advanced version of the continuous smelting process and began operating during the later stages of this study. A digital model makes provision for including its operation in the overall model. Its effect was, however, not included in the results described in this thesis.
Pig Bay

Blister copper is poured into moulds in the area marked pig bay to produce lumps of cold copper required for cooling in the converter copper blows. The model of the pig bay records the number of pigs at any time. If fresh pigs are cast a light indicator shows when these have cooled sufficiently to be removed.

Pig Pile

There is an area at the north end of the aisle where copper pigs are stored, if removed from the pig bay. Pigs are usually only taken from the pig pile, a safety stock, if there are no pigs left in the pig bay. The model of the pig bay records the number of pigs and allows additions and removals.

Bumping Block - Ladles - Skulls

After a ladle has been used several times a solidified layer deposits and is removed by using a crane to bump the ladle against a large block. Before this is done the ladle has to cool for some time. The loose crust is called the "skull". The submodel for the ladles provides a counter for the usage of ladles. A light at the bumping block signals that bumping is required if the usage of ladles reaches a given value.

By a suitable command the operator can cause "bumping". If there was no ladle at the bumping block a signal showing that there is a "ladle to be bumped" would be off. The command for bumping could not be executed in that case. The number of skulls are updated whenever one is added by bumping, or removed by a crane.
There are no models describing scrap or flue dust availability. It is assumed that there is flue dust available when required at the beginning of the copper blow, and scrap can be picked up at the north end when required. The amount of scrap processed in reality depends on its availability which varies and is not significant in the overall model.
5.0 MODEL VALIDATION AND CALIBRATION

5.1 VALIDATION OF SIMULATION MODELS

Although the stage of model validation is treated in a separate section, the processes of model building and verification are closely related. Every step during the model building phase was tested and verified against the postulates on which the model was based.

Once the overall model was operating, validation and modifications to improve the model formed an iterative process until a satisfactory performance was obtained. The isolated treatment here merely serves to identify the distinct process of validation.

Van Horn (V-1) stresses the importance of this stage and summarises the reasons for the difficulty of validating simulation models by saying that "simulation offers the most flexible and realistic representation for complex problems of any quantitative technique". He goes on to say: "Thus, many of the aspects that make validation difficult for simulation also give validation a great deal of importance."

Much attention is given to the problem of testing a simulation model in all texts on Modelling and Simulation of Systems or Processes. While no general procedure for validation of all types of models can be given most authors propose a similar approach. It should be stressed that a valid model is not one that duplicates reality but one, that meets specified design requirements.

Wigan (W-1) proposes the following hierarchy of stages involved in
developing a valid model:

1. Postulates
2. Fitting
3. Calibration
4. Identification
5. Validation

He points out that as the model is developed each stage can compensate for any errors of the previous stages and consequently the process of validation becomes more difficult.

Identification ensures that no deductions are made from the model that it is not able to reliably produce. In other words, the detail of the model should be consistent with the historic data available. This stage is of particular importance in developing regression models.

Naylor et al. (N-5) place the emphasis on the third of three phases in verification of computer simulation experiments, namely on forecasting. The first two stages, the formulation of postulates and testing of the postulates are essential in building a model but do not guarantee valid results. The ability of a model to predict the behaviour of the system under study is considered the ultimate test of a simulation model.

Mihram (M-4, M-5) makes a clear distinction between verification and validation stages and includes both in an iterative process of model development. This was pointed out earlier. Verification ensures the validity of the programmed logic whereas validation determines the correspondence of the real system and its model.
To ensure validity of the smelter model four stages of validation were considered.

1. Verification, or testing of the validity of the model.
2. Calibration of individual submodels.
3. Validation of the model using operating records.
4. Predictions of model compared to response of actual system.

The separate stages of validating the model will be discussed in the next sections.

Before leaving this section mention should be made of the generation of stochastic processes.

Random variables are used extensively in the model and have a significant influence on the validity of the model. Both uniform and normal distributions are used to represent process behaviour. During a simulation run random samples are drawn from the distributions to produce typical values of stochastic variables.

The generation of pseudo random numbers is treated in most texts on simulation of stochastic processes (M-4,N-3) and will not be discussed in detail. In this study a random number generator tailored to the EAI 640 is used. It is based on a multiplicative congruential relationship and produces a random number in the interval 0.0 to 1.0. The function routine used (1-1) is shown below where 32767 corresponds to the maximum word size of the digital computer. It is guaranteed to produce a sequence of $2^{n-2}$ numbers without repetition, where $n = 16$ is the number of binary bits in each word on the EAI 640 digital computer.

FUNCTION DRAND (IY)

IY = IY * 259
IF(1Y$5,6,6
5  IY = IY + 32767 + 1
6  YFL = IY
7  DRAND = YFL/32767
8  RETURN
9  END

The density function of a uniformly distributed variable \( y \) in the interval \((a, b)\) is given by

\[
    f(y) = \begin{cases} 
        \frac{1}{b-a} & \text{if } a < y < b \\ 
        0 & \text{otherwise}
    \end{cases}, \quad ....(1)
\]

Values of the variable \( y \) are generated by drawing a random number \( r \) in the range of 0. to 1. and converting it according to:

\[
    y = r(b-a) + a, \quad ....(2)
\]

This operation is performed in the simulation model by a function routine with two arguments, the upper limit of the distribution \( B \) and the lower limit \( A \).

FUNCTION UNFRM (A, B)

COMMON ISEED

UNFRM = A+(B-A) * DRAND (ISEED)

RETURN

END

In a similar way a normally distributed variable can be generated provided the mean and the standard deviation of the distribution are known.

The density function of a normally distributed variable \( y \) is given by:
\[ f(y) = \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{1}{2} \left(\frac{y-u}{\sigma}\right)^2} \ldots (3) \]

where \( u \) and \( \sigma \) are the mean and the variance of variable \( y \).

A method suggested by Box and Muller (B-2) is used to generate a normal deviate \( y \) directly by using two random numbers. This is suitably performed by the following function routine RNORM. Arguments being the mean of a normal distribution and the standard deviation.

FUNCTION RNORM (XMEAN, STDV)
COMMON ISEED
RA = DRAND (ISEED)
RB = DRAND (ISEED)
V = (-2.0 * ALOG(RA)) ** .5 * COS(6.283 * RB)
RNORM = V * STDV
IF(RNORM + 3. * STDV) 6, 6, 8
6 RNORM = 3. * STDV
7 RNORM = RNORM + XMEAN
RETURN
8 IF(RNORM - 3. * STDV) 7, 7, 9
9 RNORM = 3. * STDV
GO TO 7
END

To conserve memory space however, a shorter and faster routine is used which is based on the Central Limit Theorem. A detailed discussion of this method is given by Naylor (N-3). It consists of finding the mean of a sample of twelve
uniformly distributed numbers which is then transformed to give a normally distributed value. This is performed by the following function routine:

```
FUNCTION RNORM (XMEAN, STDV)
COMMON ISEED
DO 1 I = 1, 12
   S = DRAND (ISEED)
1   A = A + S
   RNORM = (A-6) * STDV + XMEAN
RETURN
END
```

At any time during the simulation a distributed variable is thus obtained simply by calling the respective function subroutine generating a uniform or normal variate and specifying the two characteristics for each distribution.
5.2 VERIFICATION OF THE MODEL

This stage is discussed by some authors under the topic of validation. It includes checking of the programmed logic, i.e. computations performed in different submodels are checked as it is done when debugging a regular computer programme. Debugging of analog models is done conveniently by observing process variables on display facilities during a simulation run rather than analysing results at the end.

Due to the interactive nature of this model, however, a completely verified model could still produce erroneous results which are not due to unsatisfactory calibration.

The operator controls aspects of the simulation by issuing commands to which the model responds. Incorrect commands due to a typing error or a decision error can thus also produce erroneous results if they are not detected and corrected before execution by the model.

To avoid these errors from occurring built in checks ensure that every command entered is legal, e.g. material can only be picked up at a location if it is available and can only be transferred to a furnace if the furnace is ready to receive it. In a similar way a broken down crane will not accept commands from an operator until it is operational again.

These checks at all submodels thus ensure that the logic of the operator, who is part of the simulation, is correct.

During the process of verification various assumptions had to be modified
since they prevented the model from operating in a realistic manner.

A good example of this is the logic of resolving conflicting crane operations. Initially certain postulates and assumptions were made as how to represent the operation of the cranes realistically. When running the simulation model it was observed that the crane submodels performed tasks in a different way than observed in reality. Changes in, the logic had to be made several times until a satisfactory operation was obtained.

Thus verification included not only checking whether the model performed what it was programmed to do but it also included a verification of the postulates put forward.

The extensive display facilities of the simulation model proved to be very useful at this stage. Most of the processes could be observed during simulation and compared to the behaviour of the system (e.g., crane movements as described above). In addition to the visual display a line printer provides information on any one or all submodels at any stage during a simulation run. This information is obtained by the command INFO followed by a component code. (The type of information provided is discussed in the detailed description of the submodels in the Appendices.) While this information is used frequently during simulation it was used extensively during verification and validation. In addition there are typeouts in most submodels that can be controlled by setting one of eight sense switches on the digital console. The information is thus provided every time a state change occurs in a particular submodel. Appendix F lists the functions of the sense switches.

Apart from consulting operating people constantly during the validation period experts from the smelter were invited twice to observe and criticise the logic and
accuracy of the model. This aspect is particularly stressed by Maisel and Gnugnoli (M-1). The involvement of operating experts was useful in particular here where the model of the operator was part of the overall simulation model.
5.3 CALIBRATION OF THE MODEL

At this stage the model was verified and assumed correct. This statement does not mean that from here on no changes to the programme logic were made. When calibrating the model, variables were assigned values based on the available data. If it was found that the model did not produce the required results further data was used to improve the model. If the additional data still did not provide a satisfactory model, the model had to be reexamined and improved in an iterative fashion as described before.

However, for the purpose of describing the process of calibration, this aspect will be treated separately, assuming that the model was completely verified.

Operating data from the plant was received in different forms which are listed below:

(1) Daily Reverberatory furnace shift reports
(2) Reverberatory furnace No. 3 control reports
(3) Reverberatory furnace log book
(4) Converter shift reports
(5) Individual converter reports
(6) Converter log book
(7) Converter air flow charts
(8) Anode furnace reports
(9) Daily smelter reports
(10) Weekly smelter control reports
In addition to these records three field studies were conducted during which operating data was recorded for all crane activities, for the operation and performance of several shifts and for particular processes such as matte tapping.

Other data such as break down statistics, weights, locations, durations of tasks etc., was supplied by Noranda upon request.

Details on the data required for particular submodels can be found in the Appendices where the individual models are described.

The extensive amount of data available was analysed and fitted to the models. Care had to be taken in analysing statistics of some processes to ensure compatibility with the mechanics of the respective submodel. An example will illustrate this point. Statistics on matte tapping times showed a normal distribution of tapping times except for some values that fell outside the range of all other variables.

These long delays in matte tapping are, however, taken into account already in the model and had to be filtered out of the distribution which was then used in the model to generate intervals for matte tapping. Calibration of the converter models was given most attention and involved variations of input variables such as air flow and oxygen efficiency for each unit while the blowing rates, and slag formation could then be compared against converter operating statistics.

Other models were calibrated in a similar way by comparing output variables of the models against operating data. If no correspondence existed, control variables were changed within allowable limits to produce the required agreement.

In some cases the interaction between processes prevented isolated testing of processes which was then done as part of the overall validation of the model as described in the next section.
5.4. VALIDATION OF OVERALL MODEL PERFORMANCE

While the validity of individual submodels is essential, it does not ensure the validity of the overall model. Interactions occur between processes which cannot be predicted. In addition the decisions of the simulation operator produce results that will not resemble operating data unless the decisions are based on the same policy that is employed in the smelter.

During field studies detailed records of five 8-hour shifts were obtained which included data on all process states crane activities and converter foreman decisions. Being able to copy the latter as closely as possible is an important prerequisite to simulate the actual operating conditions. In fact it became evident that it takes some training to become an efficient model operator.

Before describing the validation runs a brief description of the operating policy of the converter foreman will be given. The smelter is run on an 8-hour shift schedule during which time the converter foreman essentially has control over the running of the smelter.

With four converters in operation at any time the converter foreman charges matte into the converters during distinct matte blowing periods. At the end of each period slag is skimmed which is returned to the reverberatory furnace.

To proceed to a copper blow matte from one converter is transferred to another to give a large enough charge to start on a copper blow. Important in this process is proper timing of transfers to reduce idle times of converters to a minimum while ensuring that matte requirements are uniform. This policy for
controlling smelter operations is termed the "mate transfer" policy in contrast to the "no transfer" policy that is discussed in the next chapter.

This scheduling task has to be performed by the model operator during a simulation run to match the actual performance.

To compare operating data with results of simulation runs it is essential to run the model under identical conditions. Initial conditions of processes were calibrated to correspond to that of the field study. However, the decisions of the model operator, if different to those of the converter foreman of a particular shift will not satisfy the requirement of identical conditions.

The detailed data of three shifts obtained in March 1972 was drawn up as an operating sheet and used for the validation runs as a shift plan. In this way the model could be run under identical conditions including the human decisions.

Figure 5.4.1 shows results of one 8-hour simulation run compared to actual operating data based on the mate transfer policy.

Further operating data for operations under the "no mate transfer" policy were obtained in March 1973. While this policy is discussed in Chapter 6, Figure 5.4.2 is presented here to show the close correspondence of simulation data and operating data as obtained from the different operating policy. On the two Figures material additions and removals and process end points are compared for each converter.

The upper row shows simulation results while the lower row corresponds to the operating data. Material additions mt = mate, tm = transfer mate, fd = flue dust, pg = copper pigs are shown on the lower line while material removals sl = converter slag and cu = blister copper are shown on the upper line. Other
material additions that occurred (reactor slag, scrap etc.) are not shown on the diagram.

The correspondence of simulated converter operations with data from the real operation was improved to a satisfactory level by adjusting control variables such as blowing rates and service tasks. At this stage further data for matte tapping times was also obtained because of the important influence these distributions have on the availability of matte.

Apart from ensuring correspondence in the time history and material requirements of converters as discussed, data generated by the model was compared with actual data not yet used in calibrating the model. For example it is expected that the slag and matte levels (these are generated by the model) in the reverberatory furnaces change only slightly during an 8-hour shift if an average throughput of matte is obtained. Also the ratio of slag to matte from a converter is generated by the model and was compared with operating records. The amount of matte processed and the quantity of copper produced are, however, the most important measurements of validity since they are the result of all interactions of the system submodels.

While results of the validation runs above were in close agreement with actual data, exact duplication was not expected. The stochastic elements of most models prevent individual events from happening in the same sequence as they were recorded in reality. The matte tapping times for example are obtained from normal distributions in the reverberatory furnace submodels. While in reality a crane could have been free to move the material when it became available, the simulated availability of matte could occur at an earlier or later time when the crane or converter status could have been different. In simulating actual operating
conditions, that is following operating decisions in detail this situation could cause an unnecessary delay.

The overall effect of such random variations will, however, not affect the results if long enough operating periods are simulated based on free decisions of the model operator. When simulating several eight-hour shifts the average performance in terms of matte processed and copper produced can be compared. The high variability of both quantities is discussed in detail in the next chapter. Both in reality and in operating the model, the end of an 8-hour shift could come a few minutes before or after a converter is ready to be filled with several ladles of matte which could affect the shift performance by over 10%. However, if the average of several shifts is used this effect is minimised. Two sets of results are compared one for the "transfer policy" in 1972 and one for the "no transfer policy" in 1973. Figure 5.4.3 shows actual matte production for consecutive shifts expressed as a percentage of the average for the month in which operating data was obtained. On the same graph simulated results for matte production during 9 shifts are shown which were based on operating conditions of the same period. It can be seen that while the variation is of the same order the mean of the throughput is slightly lower. Figure 5.4.4 is a similar graph showing the matte throughput when operating under the "no transfer policy". Details of these results are discussed in a different context in Chapter 6. The results are presented here and show that the model is also capable of reproducing operations under the changed policy. The average simulated values here correspond more closely to the actually observed values. The reason for this seems to be the much simpler task of operating the model efficiently under the "no transfer policy". The transfer
policy requires more attention and experience to be handled efficiently which is reflected in the lower average shown in Figure 4.3.4.

A vital role was played at this stage by operating personnel. The model was operated according to instructions from operating people who assessed the response of the model. Many comments and queries indicated points of improvement which were implemented where necessary. This process ensured that the model operated to the satisfaction of operating personnel, which ensured their confidence in the model. Furthermore the role of operating people as model operators could be studied to be imitated for further simulation runs.

The fourth test of validity, the ability of the model to correctly predict the response of the system to specified inputs will be discussed in the next section.
FIGURE 5.4.1  COMPARISON OF REAL AND SIMULATED SHIFT OPERATIONS (TRANSFER)
FIGURE 54.2 COMPARISON OF REAL AND SIMULATED SHIFT OPERATIONS (NO TRANSFERS)
Figure 5.4.3 Simulated Matte Throughput Compared to Actual Throughput (1972 data)
FIGURE 5.4.4 SIMULATED MATTE THROUGHPUT COMPARED TO ACTUAL THROUGHPUT (1973 data)
6.0 MODEL ANALYSIS AND RESULTS BY MANUAL CONTROL

6.1 APPLICATIONS OF THE MODEL

Having established the validity of the model it was analysed to provide information about, and insight into the system it represents.

It was pointed out in Chapter 3 that the mechanism for improving the performance of the system was to be constructed to changes in organising, scheduling and control procedures. In other words the basic processes were fixed. The model could thus be used to analyse the following aspects of the system:

1. Improve converter scheduling
2. Improve crane use
3. Improve plant layout
4. Investigate extension of the plant or capacity increase of some processes.
5. Investigate effect of the startup of a continuous smelting process.
6. Investigate effects of changing the matte grade to increase throughput.
7. Study the use of the model as a training simulator for operating personnel.
8. Study the possibility of developing an on-line smelter control system.

For the purpose of this study the first and the last points were the most interesting to investigate, as they relate directly to the control of the system. Using
the policy of "matte transfer" described in Section 5.4 it proved to be difficult to run the model efficiently and it was felt that a "no-matte transfer policy" could provide the same or possibly a better production. This is discussed in Section 6.3.

As it turned out later this policy was implemented for a trial period in the smelter and provided further information on the validity of the model as discussed in the previous chapter.

The investigation of point 8. is discussed in Chapter 7.0. The other points listed were not studied systematically, although some mention will be made of point 2. in this chapter. In Chapter 8 further reference will be made regarding point 7.
6.2 MEASURE OF PERFORMANCE OF THE SYSTEM

To evaluate a simulation run of the model a measure of performance of the system has to be established.

When validating the model it was possible to compare individual system and simulation results directly. However, when trying to improve the performance of the model it is necessary to establish a function that represents the overall profitability of a proposed scheme or policy change.

This expression is generally termed the objective or cost function in optimisation studies. It reflects the profitability of an operation and is expected to increase as the efficiency of a system is increased. The aim in improving the system operation is thus to maximise the objective function. For a stationary linear system it can be represented by:

\[ U = \sum_{i=1}^{n} l_i x_i \]  \( \text{.... (1)} \)

where \( x_i \) = quantity of product or material > 0 for \( i=1, n \)

\( l_i \) = coefficient of contribution towards the objective

To measure the performance of a dynamic system the objective function can be defined to reflect the performance of the system during a time interval. By recording the performance of the system at equal time intervals it is possible to obtain a measure of performance for the system.
\[ U(t_i) = \sum_{i=1}^{n} l_i(t_i) \times_i(t_i) \quad i = 0, 1 \ldots n \ldots \ldots (2) \]

where \( U(t_i) \) represents the performance of the system in the time interval \((t_{i-1}, t_i] \).

In practice such a measure of performance is usually the profit of a system and would include all labour costs, overheads, energy costs, material costs which will affect the profit realised in producing copper at a given price.

The system modelled here does of course not account for all these factors, and an exact cost function for the smelter system with the given boundaries would be difficult to establish.

In the absence of such information the objective function of the system can be represented by:

\[ U(t_i) = \sum_{i=1}^{n} l_i(t_i) \times_i(t_i) + C \times (t_i - t_{i-1}) \ldots \ldots (3) \]

where \( C \) is a constant and its effect on the objective function only dependent on the time interval for which \( U(t_i) \) is evaluated. If this interval is kept constant then \( C \Delta t_i \) includes all fixed costs of the system and other costs which are assumed to be constant over given operating periods.

The quantities contributing to the objective function are all materials entering and leaving the system. All uncontrollable variables are incorporated in \( C \Delta t_i \) and are thus assumed to make constant contribution to \( U(t) \). Materials that are considered in the model are:

**Material leaving:**
1. Anode copper produced
2. Reverbogatory slag

**Materials entering:**
3. Matte
4. Flux
5. Heat

6. Air

Materials such as copper scrap, reactor slag and others are assumed to enter the system at a uniform rate and are thus also included in \( C \). Although these materials arrive in discrete batches at irregular times this effect can be approximated by an average uniform input over longer operating periods. Furthermore the quantities involved are small so that no significant error is introduced by this assumption.

The amount of flux, heat and air put into the system and the amount of reverberatory slag removed from it is proportional to the amount of matte of a given grade processed in a given period of time, thus their effect can be accounted for in the coefficient for matte which will be \( l_2 \) while that for copper \( l_1 \).

This gives:

\[
U(t_i) = l_1(t_i) x_1(t_i) - l_2(t_i) x_2(t_i) - C \Delta t
\]  

\( \ldots (4) \)

When considering inputs and outputs to the system only the levels of in-process materials are assumed to remain constant with time. The implications of this assumption are discussed below. For constant time intervals such as an 8-hour shift the term \( C \Delta t \) can be eliminated to give a relative performance of the system only dependent on the matte and copper production.

\[
R(t_i) = l_1(t_i) x_1(t_i) - l_2(t_i) x_2(t_i)
\]  

\( \ldots (5) \)

This equation will, however, produce a highly variable performance index for mainly four reasons. Copper is produced in one of the three anode furnaces in batches of several hundred tons at intervals of around 12 hours. These discrete batches reflect a copper output that appears to be rather irregular even at
daily or longer intervals.

(i) If the copper output is based on the refined anode copper cast, which
is the final output from the system, then extremely large fluctuations
would appear in the performance.

(ii) Furthermore, the storage of copper in the system in form of copper lumps
(pigs) causes an apparently lower copper output that will only be
removed if long time intervals are considered such as a week or more.
(It was assumed above that in-process material levels remain constant).

(iii) In general process cycles require additions and removals of discrete
batches of material which lead to fluctuations in the above performance
criterion if evaluated at short time intervals.

(iv) There is a considerable time lag between the input and output of
material that is not reflected in equation (5)

A simple solution to this problem would be to simulate longer operating periods which
was done to obtain reliable figures for average production. Since the simulation of
one 8-hour shift takes about one hour on the model, it was still necessary to obtain a
reliable measure of performance for that time interval which is a more convenient
simulation run than a 24-hour period for a full day.

To obtain a measure of performance for a period of 8 hours the only
useful quantity to measure is the amount of matte processed since it does not suffer
from the shortcomings discussed above. To eliminate $x_1$ from equation (4) the relation

$$x_1 = kx_2 + K$$

.... (6)

can be used which merely states that the amount of copper produced is proportional
to the amount of matte processed, of a fixed grade, plus a quantity $K$ which consists
of Copper Scrap, Reactor copper, etc. This quantity was assumed fixed for a given time period and we thus have:

\[
R(t_i) = x_2(t_i) \left[ k_l_1(t_i) - l_2(t_i) \right] + K l_1(t_i) \quad \ldots (7)
\]

or

\[
R'(t_i) = x_2(t_i) L_2 \quad \ldots (8)
\]

Equation (8) thus states that the relative performance of the system is given by the amount of matte processed during a given time interval provided the matte grade (% content of copper) remains constant.

This result will seem obvious and does in fact reflect one of the measures of performance in the smelter, namely the amount of matte processed per 8-hour shift or per day. The amount of copper produced is also quoted, but it becomes only meaningful if considered as an average over a longer period of time such as a day or a week.

Over the longer operating periods variations in the matte grade can be more significant making the measure of matte throughput less useful unless it is combined with the corresponding copper production.

Although the matte grade can change considerably even during an 8-hour interval this is rather uncommon and can safely be assumed constant.

The analysis thus shows the assumptions that are made when using the matte production as a performance index. Even with the limitations it provided a good measure for predicting the effect of an improved operating policy in the smelter.

Considering all the assumptions the result of the analysis above might seem of little value. Because of the need for a measure of performance which can be used to compare simulated and real results the above measure was chosen rather than an index that provides a good measure of performance for the model but is not directly applicable to the system.
6.3. MODEL RESPONSE TO "NO-MATTE TRANSFER" POLICY

The effect of operating the smelter without transferring white metal between converters was of particular interest and was the first area to be investigated.

It was pointed out that the transfers were made mainly to be able to schedule converters properly, such that at any time two converters are on a matte stage, the third being on a copper stage while the fourth is getting ready to go on a copper stage. In the "no transfer policy" matte is added to converters during the consecutive blowing periods, while slag is skimmed at the end of each. When a converter contains enough material it proceeds with a copper blow.

After simulating several shifts a good policy for controlling operations was developed and it appeared that the matte throughput could be increased. Figure 6.3.1 shows the matte throughput as obtained by the two different methods. The simulation runs were based on the data of March 1972. An increase of the average matte throughput is indicated in the order of 6-7%. The variation of matte throughput is of the same order as found in reality and it was realised that the significance of the result as given by a student's t-Test would be low even if more simulation runs were performed. The reasons for this were discussed in the previous section.

The willingness by the smelter management to give this policy a trial run in March 1973 provided an opportunity not only to prove that a predicted increase in production could be achieved, but also served to further validate the model. The ability of the model to accurately respond to different operating conditions had yet to be proven, which represents the fourth test for validity of the model.
The results of the field test are given in the next section.

Based on data obtained in March 1973, several shifts were again simulated according to the two different policies. The results are presented in Figure 6.3.2 and show a similar difference in the throughput of matte which is of the order of 4-5%.

Figure 6.3.3 gives a comparison of process statistics for two 24-hour simulation runs with identical initial conditions based on the two policies. As expected the usage of converters for refining has increased while the non-blowing times decreased slightly. This results in the increased throughput if sufficient material is available and if crane service is not a restricting factor.

Figures 6.3.4 and 6.3.5 show the full simulation reports at the beginning and the end of a no-matte transfer simulation run. The meaning of the variables of each process that appear on these reports is given in Appendices B-E where individual submodels are discussed. Figure 6.3.6 gives the converter blowing states as recorded on stripchart recorders. Again two simulation runs for 24-hours are presented for each of the operating policies.

The stripchart records show the amount of FeS and Cu2S in the matte and copper blows respectively. In a matte stage several batches of matte are represented by the quantity of FeS which diminishes with time as a proportional amount of slag is produced. The slag skimming periods correspond to a zero FeS level which is followed by another batch of matte addition. (The increments correspond to ladle additions.) The start of refining in the copper stage coincides with the flue dust addition and ends when all of the Cu2S has been oxidised to copper. After a skull is added copper is skimmed to complete a converter cycle. The spacing of converters is shown. The no matte transfer policy did not produce any irregular effect in fact it was pointed out that a uniform operation was very easy to maintain.
FIGURE 6.3.1 Matte Throughput for two Policies (1972 data)
FIGURE 6.3.2 Matte Throughput for two Policies (1973 data)
<table>
<thead>
<tr>
<th>Average statistics on smelter operations</th>
<th>Transfer Policy</th>
<th>No Transfer Policy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Percentage of time blowing in matte stage (converters)</td>
<td>65%</td>
<td>69.5%</td>
</tr>
<tr>
<td>Percentage of time not blowing in matte stage (converters)</td>
<td>35%</td>
<td>30.5%</td>
</tr>
<tr>
<td>Matte throughput</td>
<td>-</td>
<td>2% higher</td>
</tr>
<tr>
<td>Copper throughput</td>
<td>-</td>
<td>2% higher</td>
</tr>
<tr>
<td>Percentage of time cranes are empty</td>
<td>61%</td>
<td>63%</td>
</tr>
<tr>
<td>Percentage of time cranes are delayed</td>
<td>8.25%</td>
<td>9.15%</td>
</tr>
<tr>
<td>Percentage of time a ladle of matte was waiting on a matte car</td>
<td>31.8%</td>
<td>29.3%</td>
</tr>
</tbody>
</table>

**Figure 6.3.3** Comparison of particular 24-hour simulation runs for matte transfer and no transfer policies.
FIGURE 6.3.5 SIMULATION REPORT AFTER SIMULATING THREE 8-HOUR SHIFTS
6.4 SYSTEM RESPONSE TO "NO-MATTE TRANSFER" POLICY

In March 1973 a trial run at the smelter was undertaken which was to establish the advantage of operating without matte transfers.

It should be pointed out here that the "no matte transfer" policy is merely a general policy, it does not imply that white metal should never be transferred between converter furnaces. If due to a breakdown or some other delay two furnaces were operating almost in phase it would be advantageous to perform a transfer. This would bring the one converter back to the beginning of a cycle while the other would be advanced to a later matte blowing period. In this way the cycles of the four converters could be spaced if necessary by a transfer. Under normal operating conditions this spacing is done by varying the charge sizes in converter cycles. Good spacing of converter cycles is essential to ensure both a uniform need for matte and to avoid any two converters requiring major service at the same time.

The trial run was successful and matte throughput based on the no matte transfer policy could be increased almost immediately.

Operating personnel found it relatively easy to adjust to the new policy. After 24 hours, when each operating crew had completed one shift based on the new policy, production increased beyond the previous average. Figure 6.4.1 shows the matte throughput 10 days before and after the change-over expressed in terms of the average of the first ten days. An average increase of 6% in matte throughput was achieved over that period. This change is significant
since production before the change was considered at a high level. Also, the weekly average of matte throughput reached the highest level ever attained until then during the first week. While the initial 10 days of operation do not reflect any long term effects they show the potential of the policy on a day to day basis. Figure 6.4.2 shows the copper production in the same period.

Copper is produced, this was pointed out before, in large batches and shows a considerable variation for that reason. For this reason no conclusive evidence of an increase in copper output is provided by this plot but some increase is indicated. A somewhat lower copper output would also be expected in the first week because of a lower matte grade that was supplied in that period.

The long term effect of the policy change is shown in Figure 6.4.3. The weekly matte throughput for 1973 is plotted as a percentage of the 1972 average. This basis was chosen to eliminate the effect of the low matte throughput in the first weeks of the year. Except for the sixth week after changing to the new policy when the smelter was shut down for 7 1/2 hours a very uniform production was achieved. Using a corrected value of matte throughput for that week the average of the 7 weeks of no transfer policy is 2% higher than the 7 weeks prior to the change. Compared to the 1972 average the increase is 8%.

The trial run was initially planned for 48 hours and was later extended to a week, when it was decided that the new policy should be adopted.
Figure 6.4.1 Matte Throughput before and after policy change based on average of transfer policy
Figure 64.2 Copper Production before and after policy change based on average of transfer policy...
**Figure 6.4.3** Actual Matte Throughput Before and After Policy Change
6.5 DISCUSSION OF RESULTS

By changing operating policies or input variables of the model, the model is not changed and thus provides a basis for comparing results of different experimental runs. If, however, a particular change is to be tested in the real system it is important that the basic system is not affected by the proposed change. If the system changes, predictions would have been based on a different model and cannot be expected to correspond with results of plant tests.

It was suggested that the no-matte transfer policy could result in a lower air flow rate in the converters due to larger heads of liquid in the converter. This would adversely affect the reaction rates and reduce the refining capacity of converters. In the converter models a fixed air flow rate is assumed. This aspect of the converter model would have to be a function of the liquid head in the furnaces if such a relationship was observed.

Air flow charts of the trial period were analysed for this effect but no significant lowering was observed. A slight drop in flow rates occurred towards the end of a matte blowing stage but the overall effect was insignificant.

There were two further changes in the system when operating according to the no transfer policy that were not accounted for in the model when simulation experiments were run.

1. During the second week of operation the brick lining in converter 67 showed critical wear and the converter had to operate at reduced capacity.

During the third week it was then shut down and replaced by 65 converter.
Apart from a slightly different physical location converter #5 has a lower capacity, and a lower average air flow rate. This change in the system should decrease the overall matte throughput capability by as much as 4%. At the time it was not known that the converter would be shut down and simulation runs were based on the original configuration including converter #7.

2. The start-up of the continuous smelting plant almost coincided with the trial period for the new policy. In the first few weeks the effect was negligible but as the new plant produced more blister copper, which has to be processed in the converters during a copper blow, some of the converter capacity was used for this purpose. To estimate this effect is difficult but it is expected to be in the order of 1/2 - 1%.

Considering both negative effects on the capacity of the system the long term increase of matte throughput of 2% shown in Figure 6.4.3 is in fact in the order of 6% or more. This possible increase was indicated during the first 10 days of operation when both converter #7 was operating and the continuous smelting plant did not operate at full capacity.

The improved operating policy as predicted by the model was thus well substantiated by the data obtained from the plant test. Although there was a considerable variation in the measure of performance, the increase of matte throughput corresponded closely. This correspondence did in fact show that if the effect of any change in the system is represented in the dynamics of the model, the model can successfully predict such effects. It is thus important to ensure that the model still corresponds to the system.
before any inferences based on model predictions are made.

Of the other areas worth investigating all but one would require similar trial runs to study the effect of different factors until an improved performance was achieved, if indeed this was possible. If several dependent factors were to be investigated a number of experimental simulation runs have to be performed to account for all possible combinations of these factors. Usually the number of experiments is so large that only a selected number of factors are analysed.

Of particular interest was the possibility of producing a submodel that is able to control the smelter operations automatically without the need of any operator attention. This program could if successful control the smelter in a continuous and possibly optimal fashion. This problem and a preliminary solution is presented in the next chapter.
7.0 DEVELOPMENT OF AUTOMATIC MODEL CONTROL

7.1 DEFINITION OF PROBLEM AND OBJECTIVES

In Chapter 6 the successful change to a different operation policy, the "no matte transfer" policy was discussed for which simulation results had predicted a better performance. The policy provided the smelter operator with a different set of operating rules which when adhered to gave an improvement in smelter performance. The decisions that have to be made by the foreman were developed and refined by several simulation runs during which the strategy of no matte transfer was tested. The decision process involved in controlling the overall model can be systematically treated by defining the following four stages that occur continually during a simulation run, or for that matter in the real system:

1. Recognize the need for an operator decision.
2. Establish status of system (resources)
3. Establish projected aims (demand)
4. Establish optimum sequence of tasks, in response to required need.

Every input to the simulation model from the operator has thus been preceded by a sometimes trivial or more frequently by an intricate sequence of thoughts and decisions. An experienced operator will perform some tasks automatically while others will require conscious thought.

In the actual system, this process of routine execution of some tasks is very well displayed. If for example the operator has made a decision to skim slag from
a particular converter (this might already be done without his consent if there is no
immediate other service required), then the crane operations for skimming follow a
sequence known to the crane-operator and swumper. In other cases the foreman
relates his decisions to a swumper (a man directing cranes to perform certain services)
who will see to their execution.

The number of commands issued by a foreman during an 8-hour shift are in the
order of 100 or more. The number and complexity of the decision making process
increases with the number of variables influencing the overall performance of the
system, and only an "Economic Man" as defined by Taylor (T-3) could make
optimum decisions in all cases. An Economic Man is presumed to have three
properties. He is completely informed, infinitely sensitive and rational. Without
going into the literature of decision making behaviour by people such as Cyert (C-2)
and Simon (S-5), it can be said that the actual behaviour of human decision maker
in a complex system will fall short of the ideal. Even if decisions were perfect he
will be subject to fatigue and his performance would decrease in the course of time.

In this Chapter the development of a control procedure is described that allows
an automatic operation of the simulation model. Its limitations and possible
applications to optimise the real system performance are discussed.
7.2 HEURISTIC APPROACH TO DECISION MAKING PROCESS

The performance of the overall system is to be optimised by generating a decision sequence that results in the best possible throughput of material. In a static, deterministic system (depending on the nature and size of the problem) techniques such as dynamic programming can provide optimal decision strategies.

The smelter system, however, as represented by the simulation model represents a dynamic system changing in an indeterministic fashion with time.

For a problem of this size and structure a heuristic decision model is required which allows for one level of decision, namely that of the converter foreman. Heuristic models or procedures are described by Bowman and Fetter (B-3) as methods for searching a solution space for a solution. When used to represent a decision making process, heuristic procedures generally involve modelling of the human decision making processes. This process provides a great deal of insight into the decision making problem based on which the heuristics can then be refined and extended to produce improved control procedures.

These two goals, the identification of human decision making processes and development of automatic control procedures that could be superior to human operator control, were to be achieved.

The model with the available display unit provided an excellent tool to develop heuristic decision rules. These rules could be tested and improved in an iterative fashion by observing the behaviour of the model until a satisfactory level of control was attained.

- 127 -
While the basic idea is simple the implementation of the heuristic decision involves a fair deal of effort, if the level of sophistication of the human decision making capability is to be achieved. Work in this area has been mainly reported for conventional business models, and McMillan and Gonzales (M-7) report of one model that included aspects of human motivation expressed in terms of an index of stress.

In the smelter environment decisions are governed largely by process states and material availability and the decision model was thus based only on such variables. A control decision rule thus uses a set of state variables and causes the system to change to a new set of values by issuing commands that will initiate this change. Associated with every change is some cost and any decision should be of such nature as to maximise the profit. While the cost function is not used as such decision rules were developed to produce maximum matte throughput which corresponds with the aim of maximising profits. The decision making program developed performs three basic functions:

1. Sensing of information

2. Forecasting of expected activities

3. Making decisions and causing action based on 1 and 2.

By their nature algorithms are not easily generalised, for this reason a description of the rules is given in the next two sections, while the success of the program is discussed in section 7.5.
7.3 AUTOMATIC CONTROL OF SMELTER ACTIVITIES

The automatic scheduling and control of the converter aisle operations are performed in submodules 9-12 of the model. The different submodules are described in Appendix A but for the purpose of this description the four submodules can be considered subprograms of the overall program used for the automatic control. By pressing a sense switch (No. 7) the model is placed in "automatic operation", which thus describes the simulation state in which no operator action is required to run the model.

Apart from no commands being issued through the teletypewriter the operation of the model is identical to the manual mode. By resetting the sense switch, the model is returned to the manual mode at any time during operation.

Referring to Figure 4.1.5 on page 45 which shows the interaction of the operator with the model, the basic functions of the operator can be replaced by an automatic control program.

While the display facilities are essential in the manual mode of simulation information transfer from the model to the automatic control programme occurs directly. However, for developing control algorithms and in evaluating them the display facilities were essential. They provide continuous information and show whether or not a decision making algorithm is successful in dealing with all possible conditions that can arise during a simulation run.

Figure 4.2.3 on page 57 shows how the command input function of the operator is taken over by the "Automatic Control Programme" as soon as sense switch 7 is pressed.
The ACP has access to all state variables of each submodel and when called upon to generate a command, the decisions involved will be influenced by the state of related processes.

The task of a model operator of sensing information, making a decision and issuing a command is reproduced in a similar form in the ACP. Corresponding to Table 4.1.1 which lists triggers and actions of operator commands, similar triggers will activate the ACP. Responding to any such input a decision algorithm will generate a suitable command that ensures a continuous model operation.

What follows will be a description of the implementation which will give the best insight into the techniques used in developing the decision rules. Examples will show how particular commands are generated.

Having established the advantage of operating the smelter according to the "no matte transfer policy," the automatic operation was based on this policy. In submodule 9, coded values of the variable ISET are interpreted which are given a value whenever an event occurs which would have required an operator action during simulation. In other words the variable ISET = 1.25 relates the same information to submodule 9 that is also shown on the model display or the typewriter. Instead of the operator reacting to the information, the information is processed and the required response is generated automatically in a way that a satisfactory operation is obtained.

Table 7.3.1 lists all values of ISET with the corresponding event that sets the variable ISET. A brief description of the action that follows is given in the third column on the Table. Essentially every value of ISET is associated with a command
that is issued and a corresponding event will be placed in one of the four files, or directly executed. Events are then executed in the usual fashion, that is based on the time, if it is a time event or sequentially, if it was a crane event. In general, events generated in submodule 9 form a series of events for a given value of ISET. To give an example the value

\[ ISET = 3 \]

is set whenever an end point of a matte blow is reached.

The converter number is given by the variable NCV which allows a determination of the quantity of slag actually contained in the converter. This quantity is given by the array CV (NCV, 4) and by comparing it with the critical quantity of slag that may be left in the converter at the end of skimming, CMIN (1), it is possible to determine the number of ladles of slag that are to be skimmed. Knowing this quantity, a skimming procedure by either one or two cranes is ordered depending on the availability of the cranes. By pressing a sense switch (SSW(6)), a one-crane operation can be chosen which will override the decision of the program. A sequence of events is then generated which are interpreted and filed into the files in submodule 10 in a similar way as manual operator events are interpreted by submodule 4.

In generating the commands to skim slag, decisions have to be made as to which reverberatory furnace is to receive the slag and which crane is to transport the slag. Having skimmed all the slag, more matte is to be added which requires a decision about the place of supply for the matte. These decisions occur often and every time complete knowledge about the status of the process concerned is required. For this purpose subroutines were developed for each process or submodel which are called to provide the required information. Typically SUBROUTINE MATTE(MTUN)
is called if matte is required. The status of both matte tunnels is checked for available matte and either matte tunnel 1 or matte tunnel 2 will supply the required matte. If no matte was available at the instant, a message will be typed out to that effect.

Similar routines for other processes exist.

CONV (JZZZ, IT, NBEST, MUL)
REV (NBEST, IRS)
MATTE (MTUN)
ANODE (IT, NBEST)
CRANE (NO, LOCA, LOCB, NBEST)
TWOOCR (JCI, JC2, KAPUT, NQ, NCV)

Subroutine SORT (JU, K, MA, MI) is used to establish priorities in case of conflicting requirements of two of the same processes and subroutine TISTAT (IFI, JFI, IT, ISTAT, ICO) is used to establish the status of converters at any time. Subroutine FNDCEV (IVAL, JQ, JATT, KCOL, NSET) establishes whether at any time a particular event has been issued but has not been executed. This is necessary to prevent the same command from being issued twice.

Each of these subroutines represent a certain policy decision and can be replaced at any time by another. Typically, the reverberatory furnaces could accept converter slag in equal proportions. If each furnace has a specified fraction of converter slag assigned to it, another decision algorithm is used to establish these proportions. Another example is the use of cranes for specific tasks. Subroutine CRANE chooses the best crane to do a particular task based primarily on its availability and the location, where service is required. If skimming of slag from a converter is to be performed with two cranes, then subroutine TWOOCR performs the selection.

Figure 7.3.1 shows a simplified information flow of submodule 9. The program listing
is given in Appendix G. For every decision corresponding to a particular value of ISET a number of commands are generated and stored in the arrays:

(1) JSEQ(I)
(2) JCMND(I)
(3) NCRAN(I)
(4) JFROM(I)
(5) JTO(I)
(6) JPR(I)
(7) JQ(I)

Each array contains the coded information that corresponds to the basic command of the type:

XXYYNCR, X1, X2, IPR, Q

1 2 3 4 5 6 7

The command input in the manual mode is described in Appendix A. If less than the 7 possible variables are sufficient to make up one command, only the significant left hand variables are used. A maximum of twelve commands can be issued at one time which is enough to account for all possible sequences of commands. The arrays are transmitted to submodule 10 via COMMON statements where the commands are interpreted and filed in the crane or, time files in the same way as operator commands from the teletypewriter are.

Generation of internal commands is controlled in subroutine EVENTS in the main module. After execution of any event by subroutine EVENTS the program branches to submodule 9 if ISET > 0 which indicates that an internal command should be generated. Upon return to EVENTS the program branches to submodule 10 to file
the events if any commands were generated in submodule 9 automatically, i.e.,
when MODUL = 4 (see Figure 7.3.1). The need for an internal command does not
necessarily result in one, i.e. if material is required but not available no command
is issued at that time. The number of internal commands is given by NCRAN.

Having described the mechanism of generating commands internally during
simulation the logic of the decisions will be described briefly. It should be stressed
again that the structure of the program is such that decision algorithms can be modified
or replaced at any time. The decisions as presented are based on present operating
practice and were modified where necessary after trial runs on the model until a
satisfactory automatic operation was achieved. The discussion below will follow
the values of ISET as outlined in Table 7.3.1.

ISET = 1: Matte becomes available at Matte Tunnel 1.

This event will cause a search of all converter states which is performed by
subroutine TISTAT. If any converter is not blowing and waiting for matte it will
receive the highest priority for obtaining matte. Thereafter converters which are
blowing but require more matte are assigned priorities. Subroutine CONV and
subroutine SORT are used to select the converter with the highest priority which
corresponds to the converter with the least amount of matte added in a particular
blowing period.

Once the best choice for a converter is made subroutine FNDCEV searches
all crane files for the same event which might have been issued before. If such an
event is found no further command will be issued. Subroutine CRANE chooses the
best crane for the task if a command is issued. If no converter is ready to receive
matte the ladle is placed on the floor. No more than three ladles can accumulate
on the floor at each matte tunnel. In this case matte tapping is suspended.

ISET = 2: Matte becomes available at matte tunnel 2.

This event is identical to the one above except that the origin of matte is matte tunnel 2.

ISET = 3: End of matte blow in a converter.

If a converter was filled to capacity for a matte blow it requires skimming of slag at the end before more matte can be added. The amount of slag in the converter is given by CV(NCV, 4). In this event slag skimming is scheduled which will leave less than CMIN(1)*TPU Tons in the converter. This operation can be done by a single crane or with two cranes. For a single crane operation sense switch (6) has to be set.

The converter slag is returned to one of the reverberatory furnaces. The furnace number is supplied by subroutine REV. One algorithm for distributing slag evenly between the three furnaces and one for a 25%, 25%, 50% distribution, are available. Subroutine REV ensures that no two consecutive ladles of slag are returned to the same furnace to ensure that there is enough time to clean the slag launder between charges.

Again the selection of a crane is done by subroutine CRANE if a single crane operation is required. The selection is based on the target and origin of a task, the duration of the present task of a crane and the number of tasks to precede the task. If any crane is not operational it will not accept any commands. For a two-crane operation, subroutine TWOOCR chooses a combination of two cranes. If the centre crane is not operational a single crane operation is substituted.

Once all the slag is skimmed, one ladle of matte is charged to the converter.

The availability of matte is given by subroutine MATTE. If no matte was available, a message to this effect will appear on the typewriter: "Slag is skimmed only".
ISET = 4:  End of a matte stage.

The end of a matte stage differs from the end of a matte blow in two respects. The very last ladle of slag skimmed is not returned to a reverberatory furnace but to a converter in the matte stage. After skimming, flue dust is added to the converter which will proceed then with a copper blow. The maximum quantity of converter slag that may remain is given by CMIN(2).

These operations can also be performed with either one or two cranes. Converters able to receive high copper slag (CS) are determined in subroutine TISTAT and if there are more than one a selection is made in subroutine CONV. If no converter is available, the converter slag is placed on the floor near the particular converter. This situation should never occur in practice but if it did, provision is made for it.

ISET = 5:  End of a copper stage in a converter.

At the end of a copper stage, a command is issued to deposit a skull into the converter (Refer to ISET = 6 if no skull is available). Thereafter a number of ladles of copper are removed based on the quantity of copper as given by CV(NCV,5). These tasks can be performed with one or two cranes. Copper is always placed in one of the anode furnaces unless none of them are ready to receive any copper. In that case copper is poured in the pig bay. If there are no moulds available, the bay will be cleared first. (Refer to ISET = 15 where this is prevented from happening).

ISET = 6:  (a) A ladle has cooled sufficiently and has been bumped.

(b) Anode furnace becomes ready to accept copper.

If there were no skulls available, the first ladle bumped thereafter will cause
the status of all converters to be checked to establish whether a skull is required before skimming copper. If this is the case, the events corresponding to ISET = 5 will be issued, otherwise no action is required. The same applies if an anode furnace became ready again to accept copper, if there were none before.

ISET = 7: Addition of matte to a converter unless it is filled for a particular matte blow.

The availability of matte is established in subroutine MATTE and a command to add a ladle to the converter is issued. If no matte is available or if the same event exists in a crane file (as given by FNDCEV) then no command is issued.

ISET = 8: Converter is empty after removal of all copper.

A time event is issued here for event No. 54 at 15 minutes after the present time when charging of matte to the converter should start. When event 54 is executed as a regular time event, ISET is set to 7 and the action described above follows.

ISET = 11: A copper pig is required by a converter.

In the copper stage, pig requirements are satisfied by a command which causes a crane to pick up a pig at the pig bay and drop it in a converter. If the pig bay was empty or no cool pigs were available, then they will be taken from the pig pile. If there were no pigs at all, a message is typed out to that effect. (Refer to ISET = 15 where this is prevented from happening).

ISET = 12: Copper cakes are available at the pilot plant.

This event causes a crane to move the cakes to the pig pile.

ISET = 13: Reactor slag is available at the pilot plant.

The converter states are checked in TISTAT to find a furnace ready to
receive reactor slag. If more than one were available, selection of the best
converter is done on the same basis as for ISET = 3 or 4.

ISET = 14: Ladle requires bumping.

Subroutine LADLE records the usage of ladles and requires bumping whenever
a ladle has been used a given number of times for transporting material. Since this
condition when a ladle requires bumping can arise at the same time that another
event is to be scheduled automatically, the variable ISET would lose one of the two
coded commands. For this reason the number 5000 is added to ISET in subroutine
LADLE whenever bumping is required while the original value of ISET is retained.

At the beginning of submodule 9 a numerical value of

ISET > 5000

will cause both the bumping of a ladle and any other command code, i.e., ISET -
5000 to be executed. In addition to causing a crane to bump a ladle at the
bumping block, it is instructed to deposit another ladle at the bumping block for
cooling.

ISET = 15  (a) Less than 10 pigs available

(b) No anode furnace ready to accept copper.

If any one of the above conditions are detected when picking up pigs at the
pig bay or pig pile, then the pig bay will be emptied immediately. The next
charge of copper will then go to replenish the stock of pigs in the pig bay.

ISET = 16: Anode furnace is full.

This event triggers the command to skim anode slag from the anode furnace
which is charged into a converter in a matte blow. (At the time of programming
this event, it was not known that anode slag should be charged into a converter
in a copper blow. This change can be readily made.)
\textbf{ISET = 17:} Anode furnace is ready to cast.

Casting of the anode furnace is triggered if the respective anode wheel is ready. If it is not ready, the command will be issued again when ISET = 19.

\textbf{ISET = 18:} Slag available in the going high blow in a converter.

When enough slag is available in the final matte blow, an event is filed which causes a crane to skim slag.

\textbf{ISET = 19:} Casting is completed at one of the anode wheels.

If one of the two anode furnaces both casting on anode wheel 4 were ready to start casting before the other had finished, then this event would be triggered when the anode wheel is ready again.

\textbf{ISET = 20:} Anode furnace ready to refine.

This event causes an anode furnace to start the refining process.

\textbf{ISET = 25:} Converter requires collar pulled.

This event causes a crane to pick up the collar puller and pull the collar on a converter. It might be noted here that collar pulling is one task that can be interrupted while in progress by another crane if a higher priority task is to be executed.

A special state variable \texttt{KWT(NCR)} had to be introduced to allow a two-crane operation, which will be explained briefly by an example. If 3 ladles of slag are to be skimmed from a converter at the end of which matte is to be added, then the commands issued would be such that each crane skims one ladle of slag first, which each pours into a reverberatory furnace. Then one crane skims the third ladle and waits at the converter until the other crane has picked up and
dumped a ladle of matte into the converter. Thereafter the one crane returns the empty matte ladle while the other pours slag into a reverberatory furnace.

Under normal operations, crane tasks are executed independently by the crane and synchronization of tasks from two different cranes is not possible. In a two-crane operation, every crane event that is to be delayed contains a pointer, based on the variable KODE, to the other crane of the paired operation. This value is used to set KWT(NCR). Subroutine TKODE interprets this value in the controlling program and interrupts the sequential execution of crane tasks. This means that in the example given the crane that skims the last ladle of slag from the converter will be delayed thereafter until matte has been dumped into the converter. The event following the dumping of matte in turn contains another pointer (given by KODE in the event) which will cause the waiting crane to complete its task. If the crane with matte was to arrive before the last ladle of slag was skimmed, the same synchronizing would take place in the reverse order. This ensures that matte is not added before skimming is completed.
SUBROUTINE DISK IN SUBMODULE NINE

If a ladle requires bumping set flag, IFLAG = 1

Zero arrays for storing commands

Branch to position corresponding to ISET = 1, 2, 5 and generate command sequences

(1) Matte from matte tunnel #1 to a converter

(25) Pull collar on a converter

Requires calls to decision and support Subr. 8

Store commands in arrays, MODUL = 4: to file commands

Are commands generated? NO

MODULE = 0
no commands filed

YES

IFLAG = 1

YES

ISET = 14

NO

More than 12 commands? YES

RETURN

NO
<table>
<thead>
<tr>
<th>ISET</th>
<th>Event Requiring Response</th>
<th>Command(s) Scheduled Automatically</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Matte arrives at Matte Tunnel 1</td>
<td>Matte to a converter if required</td>
</tr>
<tr>
<td>2</td>
<td>Matte arrives at Matte Tunnel 2</td>
<td>Matte to a converter if required</td>
</tr>
<tr>
<td>3</td>
<td>End of matte blow in converter</td>
<td>Skim slag, add more matte at end</td>
</tr>
<tr>
<td>4</td>
<td>End of matte stage in converter</td>
<td>Skim high grade slag, add flue dust</td>
</tr>
<tr>
<td>5</td>
<td>End of copper stage in converter</td>
<td>Add skull and skim copper from converter</td>
</tr>
<tr>
<td>6</td>
<td>Ladle was bumped for skull</td>
<td>Add skull to converter if required before removing copper</td>
</tr>
<tr>
<td>7</td>
<td>Addition of matte to a converter if not filled for particular blow</td>
<td>Add more matte to converter if available at matte tunnels</td>
</tr>
<tr>
<td>8</td>
<td>Removal of all copper from converter</td>
<td>Schedule addition of matte for new converter cycle</td>
</tr>
<tr>
<td>9</td>
<td>Reserved for Prototype Plant</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>Reserved for Prototype Plant</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Requirement of pigs in copper stage</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>Pigs available at pilot plant</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>Reactor slag available at pilot plant</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>Ladle at bumping block has cooled sufficiently</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>More pigs required</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>Anode furnace ready for skimming before refining</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>Anode furnace finished refining</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>Slag available in going high blow</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>End of casting of A4 or A5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Anode furnace ready to refine</td>
<td>Trigger refining process in anode furnace</td>
</tr>
<tr>
<td>---</td>
<td>--------------------------------</td>
<td>------------------------------------------</td>
</tr>
<tr>
<td>20</td>
<td>Converter requires collar pulling</td>
<td>Schedule collar pulling at resp. converter</td>
</tr>
</tbody>
</table>
7.4 FORECASTING OF EXPECTED PROCESS STATE

Not all control decisions can be based on present and historic data, but require an estimate (schedule) of what might happen in the future. This is also necessary in the smelter model and will be described below by showing the estimates required and how they were obtained. If the model is placed in the automatic mode as described so far, it would not require any operator attention except for two decisions:

1. The decision to proceed with a copper blow before the end of the eighth matte blowing period, to ensure correct spacing of converter cycles.

2. The decision not to fill converters in a matte blow to capacity if this is desirable at any time.

To account for these decisions another subprogram was developed that allows forecasting of the availability of matte at the matte tunnel and the material requirements of the converters. It is on this estimate together with the smelter status that a converter foreman will base his shift schedule.

Figure 7.4.1 is a simplified flowchart of this program. Because of its size it is split up into two submodules, submodel 11 and submodule 12, but for the discussion here it can be assumed to be one subprogram. The event #40 triggers this forecasting program which is repeated after a given time interval (60 minutes and 30 minutes were used) by calling the same program to update the previous estimates. Based on data supplied by the converter and matte tunnel models the status of converters and matte tunnels is established at the beginning of the programme. A forecast of quantity and time of matte requirements for each converter cycle is made based on expected blowing times and crane service requirements. This forecast is performed by subroutine CVFORC. (Appendix G). It also provides the time when
each converter is expected to proceed with a copper blow. These expected times are used in an iterative algorithm for spacing of converter cycles until a satisfactory separation between copper blows is obtained. Subroutine MTFORC is used in submodule 12 to forecast the availability of matte within the next 70 minutes. This forecast is based on the present matte tunnel status and the expected times between arrivals of ladles of matte. The available matte is then assigned to different converters according to priorities, and additions per blowing period are stored in array MUL (NCV, IBLOW).

If a further refinement of matte charging was desired, this could be very easily implemented. The two-dimensional array MOP (NCV, IBLOW) contains the predicted optimum allocation per blow whereas presently the array MUL (NCV, IBLOW) specifies the maximum allowable charge per blow. The arrays are of the same size and MOP can replace MUL in subroutine CONV in submodule 9 where the actual charge is compared to the allowable limit before any further additions are made.

It was mentioned that a forecast to the end of each converter cycle is made. Of this information, only the expected time to start on the copper blow and the matte requirement within the next 70 minutes are used. Since the forecast is repeated every hour the projections within the next 60 minutes will be the most accurate if used for control decisions.

By pressing Sense Switch (8) the operator can request a printout of the forecast in the automatic mode of operation. The forecast will provide him with fairly reliable information regarding the next 60 minutes and less accurate estimates for later periods.
SUBROUTINE DISK IN SUBMODULES 11 & 12

Based on present status establish expected quantity and time of next matte requirement for each converter
ISSTAT = 1, 15

1

Write first matte requirements of converters and present matte supply

Produce forecast of matte requirements to the end of each converter cycle, record expected start of copper stages for each

Write forecast if sense switch 8 set

Select and order requirements within next 70 minutes

Write expected start of Cu-stages

Delay or advance start of Cu-stages to get proper spacing of converter cycles

Allocate matte to converters for next hour based on the expected availability of matte. Store allocation in MUL

RETURN
7.5 RESULTS OF THE AUTOMATIC MODEL CONTROL

The smelter model was run in the automatic mode and the response of the controlling algorithms was observed. After removing initial errors the controlling program reached a stage where actions taken were logically correct, but not satisfactory under the given circumstances.

Decision rules were improved to account for this. A good example of this is the selection of a crane for a particular task. In practice this decision is based on the present and future expected availability of a crane and its possible present and future interference with other cranes. The algorithm for selecting the best crane for a particular task was based initially on a simple decision rule and then gradually refined. Shortcomings of the algorithm were visible when cranes selection during simulation was poor. Knowing the exact status of the system at the time, the algorithm could be modified to account for additional variables.

The assignment of cranes to particular tasks is of great importance to achieve efficient smelter operation, and was given much attention. Below the stages in developing crane selection rules are given as the level of sophistication increased.

1. Based on location of crane and task only.
2. Based also on breakdown state of cranes.
3. Based also on the availability of a crane, i.e. if (1) and (2) satisfied but the crane is occupied for some time another choice is made.
4. Based also on the duration of present task. (The duration of some tasks such as collar pulling is so long that any urgent task will not be assigned
to a crane engaged in that task).

If a "two-crane" operation is desired the combination of their states has to be analysed before a selection is made based again on the above rules.

In a similar way other process selection routines were refined to give the most realistic control possible. This process should be continued until a set of decision rules is derived that shows the degree of accuracy required.

It was possible to arrive at a satisfactory level of control by just utilising all the memory space in submodule 9. Because of the need for cross-references the creation of another overlay sub-module was not considered.

While model control in the automatic mode of operation is not yet as sophisticated as in the manual operation the automatic operation does achieve the same level of production. This is possible by utilising slack capacity of the cranes for example.

Table 7.5.1 compares converter and crane statistics for manual and automatic control of the simulation model. The numerical values presented are averages of separate simulation runs. A comparison of single and two-crane operation is also given which shows the superiority of the single crane operation. Although two-crane operations can save valuable non-blowing times of converters a very sophisticated crane selection algorithm would be required to achieve this.

The merit of a particular set of decision algorithms is judged by their ability to reproduce and improve upon a decision procedure of a human model operator. Although it was possible to develop algorithms that approach the level of performance of human control, it is expected that the decision rules for the crane selection, converter selection and forecasting algorithms could yet be improved to give an
even better performance. While the logic developed for the decision model is not in
a form that would allow direct control of the smelter it was possible to show that the
decision processes involved in smelter operations can be rationalised. This is a
necessary prerequisite before any further work in this area should be pursued.
<table>
<thead>
<tr>
<th>Converter and crane statistics</th>
<th>Automatic &quot;single-crane&quot; operation</th>
<th>Automatic &quot;two-crane&quot; operation</th>
<th>Manual Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Converter blowing in matte stage (%)</td>
<td>69</td>
<td>70</td>
<td>69.5</td>
</tr>
<tr>
<td>Converter not blowing in matte stage (%)</td>
<td>31</td>
<td>30</td>
<td>30.5</td>
</tr>
<tr>
<td>Crane #1 busy (%)</td>
<td>66.5</td>
<td>69.5</td>
<td>52.0</td>
</tr>
<tr>
<td>Crane #3 busy (%)</td>
<td>72.5</td>
<td>78.5</td>
<td>58.5</td>
</tr>
<tr>
<td>Crane #1 delayed by #3 (%)</td>
<td>15.2</td>
<td>19.9</td>
<td>6.1</td>
</tr>
<tr>
<td>Crane #3 delayed by #1 (%)</td>
<td>12.3</td>
<td>13.7</td>
<td>4.7</td>
</tr>
</tbody>
</table>

**Table 7.5.1** Results of Simulated Smelter Operations by Automatic Control
8.0 CONCLUSIONS

A new technique to provide solutions to an important class of industrial problems is presented. The ability of interactive simulation for modelling systems of discrete and continuous processes using a hybrid computer was demonstrated by producing a model of a copper smelter system. The copper smelter studied represents a typical example for complex control systems encountered in practice.

The development of the model was based on existing knowledge of processes and readily measurable process variables were used to produce semi-empirical mechanistic models of smelter processes. It was necessary to include the model of a human operator which is achieved by allowing an operator to have control over those decisions in the simulation model that are also exercised by a foreman in the smelter. A display unit of smelter process states provides the major feedback on the progress of simulation. This device proved to be of considerable value in developing realistic submodels of all smelter processes. It provides a meaningful picture and the essential information of the simulation model to the model operator. This aspect was important to operating personnel who were consulted on the accuracy and realistic behaviour of the model. Mainly due to the display unit it is possible for smelter operators to gain full understanding of the model operation within a short period of time. The usefulness of this model as a training tool for operators is recognised and it is expected to be used as such in future.
Since the overall model consists of both discrete submodels which are based on the event structure of the GASP simulation language and analog models which operate in parallel it was possible to develop a model of minute by minute smelter operations which correspond closely to reality. The stochastic nature of several processes was included in the model which improves the accuracy of the system representation, but adds to the difficulty of testing and validating the model.

After careful validation the model was used to improve the model performance, which could thereafter be implemented in the real system. Several areas for improvement are indicated of which one was studied in detail. An increased throughput of the system was predicted based on a different control policy. During plant tests this prediction was substantiated which provided additional confidence in the accuracy of the model. The change in policy has since been adopted in the smelter.

The model as presented is an off-line model w.r.t. the real system that operates at a speed up factor of 20:1 and thus does not allow any continuous process monitoring.

One aspect of this work was to study the possible use of an on-line computer control system for the entire smelter which would continuously control smelter processes in an optimal way.

As a first step an automatic control program was developed which performs the decision task of the model operator. The level of sophistication obtained was satisfactory to suggest that on-line control seems feasible given that suitable process measuring devices are available. It is expected that the implementation of such a system could provide the smelter operator with such information that leaves him more time for higher level decisions.
Necessary routine decisions for optimal smelter performance could be suggested by the computerised control system. If the wealth of short term and long term information of the smelter can be made available to a control program in a reliable form then such a program could become a valuable aid in copper smelter or similar operations.
REFERENCES


G-1. GPSS, General Purpose Simulation System, IBM, Technical Publications Department.


Leaf 159 omitted in page numbering.
5-4. Syrcek, W.Y., and Sandholm, D.P., An Experience in the Selection of a

   N.Y., 1960.


T-1. Templeton, F.E. and Hankley, W.J., Dynamic Optimal Control of a Process
   with Discrete and Continuous Decision Variables, 1970, International

T-2. Tocher, K.D., Review of Simulation Languages, Operations Research Quarterly,

T-3. Taylor, D.W., Decision Making and Problem Solving, in Handbook of Organizations,

V-1. Van Horn, R., Validation, Symposium on the Design of Computer Simulation

W-1. Wigan, M.R., The Fitting, Calibration and Validation of Simulation Models,
   Simulation, May 1972, pp. 188-192.
APPENDIX A

CONTROL AND ARRANGEMENT OF OVERALL SIMULATION MODEL

A.1 MODES OF OPERATION

The mechanism of hybrid simulation as implemented in this study is described here. Together with Appendices F and G it provides a detailed overall picture of the simulation model.

After initialising the model at the start of a simulation run control of the simulation is maintained in subroutine GASP. A general flowchart of subroutine GASP is given in Figure A-1.

The two possible modes of operation of the simulation model (OPERATE and HOLD) are clearly seen on this diagram. (The INITIAL CONDITION mode is only used to set up the analog program before the simulation is commenced).

The area enclosed by dotted lines corresponds to the OPERATE mode while all other operations are performed in the HOLD mode. The function of the controlling digital program in the OPERATE mode is that of checking the logical links between the analog and the digital computer by continuously reading the logical states of sense lines and comparators. This is shown on Figure A-2 which shows the detailed flowchart of subroutine GASP. In the OPERATE-mode the analog computer executes all the functions of the analog submodels including those of the timing circuit and the time display.
When the analog computer is placed in the HOLD mode the remaining sections of the program are executed including all other subprograms to which the program might branch. Simulation time will thus only advance in the OPERATE mode, during which time only continuous variables can change. Although the HOLD mode can be described as being one in which all analog states are frozen it does not mean that no state change of the models occur.

In the HOLD mode all discrete changes to analog as well as digital variables occur which gives it an equally important part in the execution of the simulation programme.

In the following sections the two modes of operation will be discussed.
A.2 THE OPERATE MODE

The analog converter and crane submodels are treated separately in the following Appendices and will not be discussed further here. During the OPERATE mode these submodels are executed in parallel and represent the dynamic aspects of the model. There are two further analog circuits, the timing circuit and the time display circuit. Figure A-3 shows the circuit diagram of the timing circuit. Its purpose is that of linking the discrete event simulation to the analog simulation. At the start of an OPERATE period comparator C39 is set to a positive value TREL which corresponds to the time interval to the next discrete event in the time file. (The time file contains all discrete events in chronological order whose execution produces the discrete event simulation). Integrator A35 integrates during the OPERATE period until the condition TREL = TAN is reached when comparator C39 changes its logic state which resets A35 to the INITIAL CONDITION. This condition is sensed by the control programme which places the computer in the HOLD mode while executing the discrete event before removing the next event from the time file and resetting the timing circuit. The potentiometer P35 is set when initialising the programme to give the correct rate of integration TTC. With a speed-up factor BETA = 20 and a maximum time interval TIMEX = 60 min the scaling of the circuit is complete, and the integration is performed according to:

\[
\frac{TAN}{TIMEX} = \int_{0}^{t} TTC \, dt + 0 \tag{1}
\]

where

\[
TTC = \frac{BETA}{TIMEX \times 60}.
\]

- 163 -
Figure A-4 shows the time display circuit. It consists of a single integrator whose scaling is similar to that of the timing circuit described above. The potentiometer setting is multiplied

\[ TTC = 0.6 \times TTC \]

to give a 1/10 volt/min reading for the elapsed simulation time. The integrator output is displayed on a digital voltmeter as minutes. Every hour the integrator is reset to its INITIAL CONDITION in a time event (event 59) by setting control line 6 high. The hour is also updated in that event. The hour reading is scaled and its value is set on D/A - 10 which is displayed on another digital voltmeter. By selecting suitable scales on the voltmeters a simulation time reading correct to the nearest minute is obtained.

An OPERATE period can be terminated by one of four possible conditions.

1. The timing circuit reaches TREL
2. Converter submodel reaches an end point
3. Crane submodel reaches the end of a task
4. Model Operator interrupts simulation. (Usually to issue a command to a particular submodel)

All four conditions will be sensed by the controlling program and the operations that follow in each case are discussed in the next section.

If an OPERATE period is terminated by a condition other than a time event the simulation at that time is not given by TNOW and is computed by FUNCTION routine TIME(TLAST). The intermediate time TAN is read from the timing circuit and converted to give the present simulation time.

\[ \text{i.e. } TIME = TLAST + TAN \times TIMEX \]

where TLAST = TNOW only when a time event has just occurred
A.3 **HOLD-MODE**

Referring to the lower section of Figure A-1 each of the four possible conditions resulting in a HOLD call will cause the programme to branch to a different section for further execution.

The first condition corresponding to a time event will now be discussed. The program branches to statement 99 in Figure A-2 where the event given by JEVNT is executed at time TNOW through a call to subroutine EVENTS. Based on the event number a section of the program is called from disc to execute the particular event. (Execution of an event involves executing a section of a submodel which corresponds to the event number). The arrangement of submodules for the memory overlay technique is given later. After execution of the time event which is described by 4 ATTRIBUTES

\[
\begin{align*}
\text{ATTRB}(1) &= \text{Event Code 1 to 999} \\
\text{ATTRB}(2) &= \text{Time at which event occurs} \\
\text{ATTRB}(3) &= \text{Other characteristics of the event} \\
\text{ATTRB}(4) &= \text{Other characteristics of the event}
\end{align*}
\]

the next time event in line is removed from the time file by subroutine RMOVE.

Subroutine SETIME is shown in Figure A-5 which is then called to reset the timing circuit shown on Figure A-3. The four attributes of the next event are stored in different variables than ATTRB(1 to 4) since these variables are reset when other events occur before TREL has elapsed.
The GASP-Simulation language provides a choice of scale for variables used in the fixed point filing arrays. That is, values of ATRIB are stored with one decimal point if a SCALE = 10 is used, which was the choice in this model. With the minutes as units this gives a timing accuracy in the order of six seconds. A larger time scale such as 100 was not required, which would also have limited the length of simulation to less than six hours. With the present choice it is possible to simulate at least six consecutive 8-hour shifts without exceeding 32767/10 minutes. (32767 being the largest integer number possible on the computer)

This period of 48 hours of simulation might seem short for a simulation run, but since the model reproduces all detailed operations and is run by an operator it would take a minimum of six real-time hours to perform this simulation run. If in future a longer simulation period was required possibly by automatic operation it would be quite simple to extend the simulation period. Another temporary time file could be created in which all events to occur later than 48 hours could be filed - starting again from zero time. When the last event before 48 hours was removed from the time file all events could be transferred back to the time file and the simulation could go on for several 48 hour intervals.

When calculating statistics based on the total simulation time previous 48 hour intervals have to be accounted for.

If no other event had occurred (such as a crane reaching its target) at the same time then the program would branch to statement 72 and place the analog computer back into the OPERATE mode. Excluding other interruptions this process could continue and would represent a real-time discrete event simulation with a 20:1 speed up factor.
If the second condition end point of a converter, became true the corresponding discrete event IARG is executed. This is done by calling subroutine EVNTS in the same way as described above whereafter the OPERATE mode continues. Reference can be made to Appendix D, where details of the converter submodels are given. If the third condition arose, when a crane has completed a task, the program branches to statements 199, 299, 100 depending on the crane number in Figure A-2 on the second page of the flowchart.

The state variable JWT (NCR) ≠ 0 is an indication that a crane cannot proceed with its task, i.e. it is delayed by another crane. In the case that the crane is not delayed the event to be executed at the end of a crane task is again executed by a call to subroutine EVNTS. Unless there is no other event in the crane file the next crane event is removed from the particular crane file, crane interferences are resolved and any delayed cranes are reset to continue before returning to the OPERATE MODE. The details of these operations are discussed in Appendix B.

In subroutine GASP statistics on crane activities are also recorded by summing the "busy" and "delayed" periods of time in TCRAN(NCR) and SP(NCR).

The fourth condition to place the analog computer in the HOLD mode is given when a push button on the digital computer console is pressed which sets sense switch 1. In this case the control of the program goes to statement 50. A call to subroutine RD is made where any number of operator commands are accepted from the teletypewriter, decoded and translated into events and filed in the crane and/or time files. This section is described below. After the last input command the first crane events are removed if any cranes were idle and the program returns to the operate mode.

If there were no other source of time events other than time events
themselves, then whenever the timing circuit was set it would operate until the required time interval between events had elapsed. Since both operator commands and crane activities can cause filing of time events it is necessary to search the time file whenever one of the two conditions occur. This is done by subroutine TSTTFL (Figure A-6) which causes the timing circuit to be reset if a time event was to occur before the one previously scheduled. This ensures that all events are executed in the correct time sequence. During a HOLD stage any combination of the four conditions leading to the HOLD call can occur and will be executed consecutively.

The variable MODUL is used when operating the model in the automatic-mode only. If during simulation a command is to be generated internally or if forecasting of material requirements was necessary then this is indicated by MODUL = 4. That is before the controlling program returns to the OPERATE mode the program branches to the section where commands are internally generated or the forecasting is performed. Subroutine TKODE is called whenever a crane event is removed from a file. In the automatic mode TKODE is used to synchronise tasks of two cranes as discussed in Chapter 8.
A-4 PROGRAM DIVISION FOR MEMORY OVERLAY

Table A-1 lists the twelve submodules of the program as used for the memory overlay. A submodule refers to a section of the program that is characterised by a file name and if not in core it is stored on one of two discs available with the EAI 640 digital computer.

The program sections in each file do not necessarily refer to a particular subprogram, but are chosen in such a way that no reference from one submodule to another is necessary.

At any time only the main module is in core with any one of the submodules. In this way it is possible to utilise the 16K memory for a program of much larger size.

The main module consists of a short main program (see Appendix G) which first loads the submodule #1 to initialise the model and then branches to subroutine GASP from where the simulation run is then controlled. If any call to a submodule is made from the main module this is done by calling subroutine EXEC(NAME) with the corresponding submodule number as an argument. Every submodule has the same name subroutine DISC which is called by this routine.

A detailed description of the overlay system is given in Ref. (G-2)

Referring to Table A-1 submodules 2, 3, 6, 7, 8 thus contain digital submodels and digital parts of the digital-analog models.

Submodule 4 is used for interpreting and generating operator commands which are described in the next section.
Submodule 5 provides information on model states by writing variables and statistics when requested. No further description of this submodel is given, the listing in Appendix G is self-explanatory.

Submodules 9, 10, 11, 12 are used only if the model is run in the automatic mode.

Submodule 1 is called into core only once at the beginning of a simulation run to initialise a run. Model variables requiring initialisation are contained in the labeled COMMON blocks BLOK1 and BLOK2 (see Appendix F). Their values are read in from data cards in submodule one together with data cards as required for the GASP discrete event simulation. These include initial events to start certain smelter processes. At the end of Appendix G a typical set of data cards is shown.

Subroutine DATAN initialises all GASP variables, and analog models are initialised by setting corresponding analog components. Based on initial conditions the display unit is initialised by setting all light indicators.
A.5 COMMAND INTERPRETATION AND TRANSLATION

The program for command interpretation and translation was developed by Dr. U.P. Graefe and the description is taken from Reference (G-4).

Operator commands can be issued through the teletypewriter by pressing sense switch 1 until ENTER is typed on the teletype. Thereafter a command can be entered. After the carriage return key is pressed, the teletype will prompt for the next command by typing ENTER etc. If a typing error is made in a command, the command may be cancelled by issuing a CC command immediately after the faulty command. The command "will terminate command inputs. Except for the CC and " commands, the commands consist of two two character command syllables followed by variable lists of various length, depending upon the type of command.

A list of acceptable commands is shown in Table A-2.

Translation of operator commands is performed by subroutine RD and the routines in submodule 4 (See Appendix G). As soon as the operator pushes sense switch 1 on the digital computer console the analog computer is put in the HOLD mode and subroutine RD is called.

Subroutine RD will prompt the operator for input by typing ENTER. Upon entering commands submodule 4 is loaded in order to gain access to the interpreting routines. When entering several commands in sequence a one command delay will be noticed, since a command will only be interpreted after the succeeding command has been entered. This is to allow for correcting a faulty command just after it has been entered by following up the faulty command by the CC command.
A flow chart for subroutine RD is shown in Figure A-7. The command syllables of the command to be translated are sent to submodule 4 via the variables IDIX and ID2X in command block ARG, while the other command variables are transmitted in array IBUFX which is part of common block BLOK2.

With the exception of the FIX, STOP, OIL, INFO, LITE, GW, GOWT, REFN and CAST command, subroutine DISK assigns two codes to each command; a ISEQ code and a ICMD code. The ISEQ code is equal to 1 for pick commands, equal to 2 for dump commands and equal to 3 for pick-and-dump commands. The ICMD code is a number presently between 1 and 13, which determines the type of material to be picked up or dumped. Table A-3 shows the abbreviations used for components and materials which are marked by an asterisk in Table A-1.

A command is executable if it is a valid command involving the pick-up of material which is available at the specified location, or the dumping of a material into a device that is ready to receive the specified material.

The FIX, STOP, OIL, LITE, GW, GOWT, REFN and CAST commands are executed directly by subroutine DISK. For the INFO command the component code for the device in question is stored in variable IX of the common block ARG, which causes submodule 5 to be loaded and executed as soon as control returns to subroutine RD.

In subroutine CMNDG, the quantity Q of material being picked up is checked. If it was specified as being larger than 2.0, it is set to 2.0. A zero value for Q, TW and IPR - that is quantity of material, waiting time and priority respectively - is meant to signify that default values are to be substituted for them. The default value for Q is 1.0. Default values for TW and IPR depend on the
command, and are picked out of the two dimensional arrays IWTNGT and IPRA respectively, using codes ISEQ and ICMND as array indices. The stored default waiting times are given in seconds, and hence require conversion to minutes.

Following this an event code is picked and the event filed in the appropriate crane file using subroutine GWAIT. If a command is issued to a crane which is not operational at the time, the event is not filed, but a message - CRN "CRANE "DOWN TILL "TIME" - is typed out instead.
FIGURE A-1 General Flowchart of GASP

Subroutine GASP

SEN$SW(6)

Remove next event from time file

Execute present discrete time event

Accept operator command

Select tasks from crane files according to priorities and execute crane tasks if cranes are idle

Was automatic event filed

Set analog comp. in operate mode

Test sense lines and comparators for end points of continuous time processes Test S$SW(1)$ for operator input

Are one or more true

Set analog comp. in hold mode

Crane event

Executive converter Ex

Execute crane event, pick next crane task from file and execute

Continue time event

Time Circuit

SS$W(1)$ Set

50

72
FIGURE A-2  DETAILED FLOWCHART OF SUBR. GASP
FIGURE A-3  TIME CIRCUIT LINKING DISCRETE DIGITAL TO CONTINUOUS ANALOG SIMULATION

- True when a time event is reached
- False when resetting TREL
FIGURE A-5  FLOWCHART FOR SUBR. SETIME

SUBROUTINE SETIME (NSET)

TNOW = ATRIB(1)
JEVNT = ATRIB(2)
ISTRIB(1) = ATRIB(3)
ISTRIB(5) = ATRIB(4)

TREL = TNOW-TLAST

TREL < 0.0

TREL = 0.05

TREL > TIMEX

PAUSE II

TREL = TREL/TIMEX

CALL CONTRL (5, 1)
CALL QWJDAR(TREL, 9, IER)
CALL CONTRL (5,0)

RETURN END
SUBROUTINE TESTFL(NSET)

MFE(1) < 1

KXXX = MFE(1)
TTEST = FLOAT(NSET(1,KXXX)) /SCALE

TTEST < TNOW

RETURN

ATRIB(1) = TNOW
ATRIB(2) = JEVNT
ATRIB(3) = ISTRIB(1)
ATRIB(4) = ISTRIB(5)

CALL FILEM (1,NSET)
CALL RMOVE (KXXX,1,NSET)
CALL SETIME (NSET)

RETURN
SUBROUTINE RD

PROMPT FOR INPUT

ACCEPT INPUT OF COMMAND SYLLABLES IN IDIX, ID2X and REMAINDER OF COMMAND IN BUFFER IBUX

PROMPT FOR INPUT NO YES RETURN

IDIX = END SYMB.

ACCEPT INPUT COMMAND SYLLABLES IN ID1Y, ID2Y and REMAINDER OF COMMAND IN BUFFER IBUFY

IDIX = END SYMB. ??

ID1Y = CORR. SYMB. CC?

LOAD, IF NECESSARY, AND LINK TO SUBMODULE 4

IF IX ≠ 0 LOAD SUBMODULE 5

REPLACE :IDIX, ID2X BY ID1Y, ID2Y and IBUX by IBUFY

LOAD, IF NECESSARY, AND LINK TO SUBMODULE 4

IF IX ≠ 0 LOAD SUBMODULE 5

RETURN

FIGURE A-7 FLOWCHART FOR SUBR. RD
**Figure A-8 Hybrid Computing Components**

- **Analog Signal**
  
- **Logic or Digital Signal**
  
- **Potentiometer**
  ![Potentiometer Diagram]
  
  \[ x \rightarrow \frac{a}{1} \rightarrow ax \]
  \[ a < 1 \]

- **Inverter**
  ![Inverter Diagram]

- **Summer**
  ![Summer Diagram]
  \[ y = (w + z + 10x) \]

- **Integrator**
  ![Integrator Diagram]
  \[ y = -z \text{ (INITIAL CONDITION)} \]
  \[ y = -z-k \int_0^1 (w+10x)dt \text{ (OPERATE)} \]
  \[ y = -z-k \int_0^1 (w+10x)dt \text{ (HOLD)} \]
  \[ k = \text{Analog Time Scale} \]

- **Zero Limiter**
  ![Zero Limiter Diagram]
  \[ z = -(x+y) \text{ for } z > 0 \text{ (+LIM)} \]
  \[ \text{or for } z < 0 \text{ (-LIM)} \]
  \[ z = 0 \text{ for } z < 0 \text{ (+LIM)} \]
  \[ z = 0 \text{ for } z > 0 \text{ (-LIM)} \]

- **Comparator**
  ![Comparator Diagram]
  \[ U \text{ true for } (x+y) > 0 \]
Function relay

\[ s^+ \quad s^- \]

Relay under logic control with manual override

And Gate

\[ a \quad b \quad c \quad \bar{c} \]
c true if a and b true
\bar{c} inverted signal of c

Individual Flip/Flop

\[ S \quad T \quad R \]
S- input to set F/F
R- input to reset F/F

Monostable Timer

Output high for \( t \) seconds after arrival of input

Logic Differentiator

Output high for one clock period after arrival of input

Digital to Analog Converter

\[ x^1 \quad x \]
\( x^1 = \) quantized \( x \)

Analog to Digital Converter
<table>
<thead>
<tr>
<th>Submodule No.</th>
<th>File Name</th>
<th>Contents of files or function</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>UPGCRN</td>
<td>Crane and other subroutines</td>
</tr>
<tr>
<td></td>
<td>UPGGSP</td>
<td>Required GASP subroutines</td>
</tr>
<tr>
<td></td>
<td>UPMNCl</td>
<td>Controlling program GASP, files</td>
</tr>
<tr>
<td></td>
<td>UPGCRN, UPGGSP and other system routines</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>UPS1Cl</td>
<td>Initialising entire simulation model</td>
</tr>
<tr>
<td>2</td>
<td>UPS2Cl</td>
<td>Events 1-38</td>
</tr>
<tr>
<td>3</td>
<td>UPS3Cl</td>
<td>Events 60-169</td>
</tr>
<tr>
<td>4</td>
<td>UPS4Cl</td>
<td>Interpretation of manual commands</td>
</tr>
<tr>
<td>5</td>
<td>UPS5Cl</td>
<td>Information on model status</td>
</tr>
<tr>
<td>6</td>
<td>UPS6Cl</td>
<td>Events 170-249</td>
</tr>
<tr>
<td>7</td>
<td>UPS7Cl</td>
<td>Events 250-300</td>
</tr>
<tr>
<td>8</td>
<td>UPS8Cl</td>
<td>Events 39-59</td>
</tr>
<tr>
<td>9</td>
<td>UPS9Cl</td>
<td>Automatic event generation</td>
</tr>
<tr>
<td>10</td>
<td>UP10Cl</td>
<td>Automatic event interpretation</td>
</tr>
<tr>
<td>11</td>
<td>UP11Cl</td>
<td>Forecasting and scheduling of material availability and requirements</td>
</tr>
<tr>
<td>12</td>
<td>UP12Cl</td>
<td>Continuation of UP11Cl</td>
</tr>
<tr>
<td>1st Symb.</td>
<td>2nd Symb.</td>
<td>List of Variables</td>
</tr>
<tr>
<td>-----------</td>
<td>-----------</td>
<td>-------------------</td>
</tr>
<tr>
<td>CC</td>
<td>FN</td>
<td>Furnace</td>
</tr>
<tr>
<td></td>
<td>ST</td>
<td>Furnace</td>
</tr>
<tr>
<td>RE</td>
<td>XX*</td>
<td>Crane</td>
</tr>
<tr>
<td>CA</td>
<td>WT</td>
<td>Crane</td>
</tr>
<tr>
<td>GW</td>
<td>LL</td>
<td>Crane</td>
</tr>
<tr>
<td>GO</td>
<td>MP</td>
<td>Crane</td>
</tr>
<tr>
<td>PU</td>
<td>YY*</td>
<td>Crane</td>
</tr>
<tr>
<td>BU</td>
<td>YY*</td>
<td>Crane</td>
</tr>
<tr>
<td>PK</td>
<td>LZ</td>
<td>Device code, XX*</td>
</tr>
<tr>
<td>DP</td>
<td>FO</td>
<td>Blank or 9</td>
</tr>
<tr>
<td>OI</td>
<td>YY*</td>
<td>Source device</td>
</tr>
<tr>
<td>IN</td>
<td>TE</td>
<td>Code, 0, for off</td>
</tr>
<tr>
<td>PD</td>
<td>OP</td>
<td>Conv, code,</td>
</tr>
<tr>
<td>LI</td>
<td>X</td>
<td>Light, 0, for 1,</td>
</tr>
<tr>
<td>ST</td>
<td></td>
<td>Length of stop,</td>
</tr>
<tr>
<td>FI</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Move crane to pre-established position without going through crane files
# Table A-3 - Device and Material Codes for Command Input

<table>
<thead>
<tr>
<th>XX is the 2-character device code</th>
<th>YY is the 2-character material code</th>
</tr>
</thead>
<tbody>
<tr>
<td>R&lt;sub&gt;i&lt;/sub&gt;</td>
<td>SL</td>
</tr>
<tr>
<td>A&lt;sub&gt;i&lt;/sub&gt;</td>
<td>MT</td>
</tr>
<tr>
<td>M&lt;sub&gt;i&lt;/sub&gt;</td>
<td>TM</td>
</tr>
<tr>
<td>C&lt;sub&gt;i&lt;/sub&gt;</td>
<td>CU</td>
</tr>
<tr>
<td>F&lt;sub&gt;i&lt;/sub&gt;</td>
<td>PG</td>
</tr>
<tr>
<td>PP</td>
<td>SC</td>
</tr>
<tr>
<td>PB</td>
<td>RS</td>
</tr>
<tr>
<td>BB</td>
<td>FD</td>
</tr>
<tr>
<td>PI</td>
<td>SK</td>
</tr>
<tr>
<td>NE</td>
<td>LD</td>
</tr>
<tr>
<td>SE</td>
<td>CP</td>
</tr>
<tr>
<td>GP</td>
<td>CS</td>
</tr>
<tr>
<td>AL</td>
<td>AS</td>
</tr>
<tr>
<td>PT</td>
<td></td>
</tr>
</tbody>
</table>
APPENDIX B
CRANE SUBMODELS

B.1 DESCRIPTION OF CRANE TASKS

In modelling the activities of the three overhead gantry cranes in the converter aisle the following terms are used frequently:

- command - instruction to a crane to perform a task
- task - activity of moving a crane from one point to another in the converter aisle with or without any load where it will remain for a specified period of time, at the end of which an event can be triggered by the crane.
- origin - point of departure of crane
- target - destination of crane when responding to a command
- waiting time - period of time that a crane remains at the target which corresponds to the duration of the activity at that point
- crane file - memory locations containing a number of crane tasks that have not been executed.

It is possible to break down all crane activities into four basic phases:
1. Establish feasibility of an operation.

2. Movement from origin to target.

3. Waiting at target for the time it takes to perform an operation.

4. Recording the change of state of a process caused by the crane activity.

The following assumptions are made:

1. A crane moves at its maximum speed, unless it is slowed down by a slower crane.

2. Crane movements along the aisle are always longer than across, i.e., the trolley movement is not considered to contribute to the duration of a crane movement.

3. The duration of any lifting lowering or other operation can be represented by average times for such tasks.

4. The change of state of a process receiving service from a crane occurs instantly and at the end of the servicing period.

5. Every task has a unique priority assigned to it which increases every time the task is interrupted or delayed.

The four phases of a task are executed partly by a digital and partly by an analog crane model. It is important to note that these tasks are performed in parallel and independently by each crane. Once a command is issued to a crane the task is recorded and the model branches to the respective subroutine. Figures B-1, B-2 and B-3 are flowcharts of each of these CRANE subroutines.
PHASE ONE

In the first phase the feasibility of executing the task is established.

Apart from the target and waiting time the priority and state of activity is used to test whether an interference with another crane exists. No conflict arises if the target of the crane ready to execute a task is separated more than 20 ft. from the next crane, and the task will be executed instantly. Some examples will show best how the conflicts are resolved if interferences of two cranes occur.

The logic involved is shown on the flowcharts of the crane routines.

The arrangement of the cranes as it exists in the smelter is used here where crane No. 1 is on the south end, crane No. 2 on the north end, and crane No. 3 in the centre.

1. If No. 1 crane is to move beyond the present position of both other cranes which are idle, then these move out of the way. Without delay the task is executed.

2. If in the same situation as in 1. crane No. 3 is busy with a task that is to take less than 2 minutes, then crane No. 1 is delayed until that time, when it will take up its task with a priority increased by one unit in a range from 1-9. (See also Table B-1 for a description of all variables of the crane models.)

3. If in the same situation as in 2. crane No. 3 was only about to start its task which had a priority lower than that of crane No. 1, then the task of crane No. 3 would be delayed until crane No. 1 had completed its task. (The situation would be reversed if the priority of crane No. 1 was the lower of the two.)
4. If an interference between the cranes 1 and 2 occurs with crane No. 3 being idle, then the situation is the same as the above with the idle centre crane responding freely to any movements.

5. If however, both cranes No. 3 and No. 2 obstruct the movement of No. 1 crane, then first the interference with crane No. 3 is resolved according to the same rules. If No. 1 crane is successful, then the interference with crane No. 2 is resolved. If crane No. 3 is executing a task that can delay No. 1 crane, then crane No. 2 will be unaffected. If on the other hand crane No. 2 is the cause of a delay then crane No. 3 will continue its task unaffected.

In general any task that is delayed will be commenced as soon as the crane causing the delay has completed its task.

The situation where a task of lower priority will be executed first arises when the task has been started already and cannot for practical reasons be interrupted. In effect the priority increases as the end of the task approaches. On the other hand tasks of long durations such as "collar pulling" can be interrupted if necessary to allow passage of another crane.

This concludes the phase of establishing the feasibility of a crane operation. It is the most difficult one and only by refining the decision algorithms and assigning correct priorities to different tasks it was possible to obtain realistic crane models. Both phases two and three of a crane task are performed by the analog model of the crane. Figure 8-4 is the circuit diagram for one analog crane model. There is one for each crane.
PHASE TWO

Once interferences are resolved the crane circuit (Fig. B-4) is set according to the task that is to be performed. The variables, $P_0(l)$, the target position and $WT(l)$, the waiting time, together with the crane number are specified when calling the subroutine GOWT, which performs the setting up of the circuit.

A flowchart of GOWT is shown in Figure B-5.

The crane circuit is composed of analog components that are controlled by logic signals, the functions of each will be described below. The equation describing the movement of the crane along the converter aisle is:

$$X = \int_{T_0}^{T_1} X \, dt + X_0$$

where

- $X_0 = \text{position of crane at time } T_0$
- $X = \text{position of crane after time } T_0$
- $\dot{X} = \text{speed of crane}$

This equation reads as follows after scaling if the corresponding variables are used:

$$\frac{X_{CR}}{X_{MAX}} = \int_{T_0}^{T_1} \frac{X_{DOT}}{X_{MAX}} \, dt + \frac{X_0}{X_{MAX}} \quad \ldots(2)$$

The assumption that cranes move at the fixed maximum speed was specified earlier which gives:

$$\frac{X_{CR}}{X_{MAX}} = \frac{X_{DOT}}{X_{MAX}} \int_{T_0}^{T_1} \, dt + \frac{X_0}{X_{MAX}} \quad \ldots(3)$$
This integration is performed by integrator I-1 while the speed is specified by potentiometer P1. The direction of travel forward or backward is determined from the previous position and is specified by function relay F1 which is switched to a positive or negative reference voltage. Given the initial position (P2) and a speed, the crane position will at all times be available to the digital model by reading the A/D converter that gives XCR. If the analog model is placed in the OPERATE mode the crane will move until it reaches its target position, where the crane movement stops.

This is achieved by subtracting the target position (XTAR = P0(I)) from the crane position and finding when its absolute value becomes zero:

\[
\frac{XCR}{XMAX} - \frac{XTAR}{XMAX} < 0.001 \quad \ldots(4)
\]

The subtraction and finding of the absolute value is performed by the two summers (S1, S3) and one limit summer (S2) as shown. If the value reaches .001, the value given by potentiometer P3, comparator C1 changes sign, which in turn affects the logic circuit connected to it. At this point the logic signal A will cause the individual integrator I1 to stop integrating, i.e., the crane will stop once the target is reached.

**PHASE THREE**

When a task is to be executed by the crane both the target position XTAR and the time to wait there TWAIT (TWAIT = WT(I)) is set. The lower half of the circuit simulates the waiting period according to:

\[
\frac{T}{TMAX} = \frac{C}{TMAX} \int_{T1}^{T2} dt + \frac{\text{CONST}}{TMAX} \quad \ldots(5)
\]
Where $T$ is the time the crane waits at its target, $\frac{\text{CONST}}{\text{TMAX}}$ will be zero at the beginning of every waiting period. The integration is performed by integrator I2 while the rate C is set on potentiometer P4. The rate is dependent only on the time scale of the simulation which could have any desirable value.

Since a speed-up of $\text{BETA} = 20$ was used:

$$C = \frac{\text{BETA}}{\text{TMAX} \times 60} = 0.0167 \text{ (in volts/sec.)}$$

Scaling of the analog circuits is completed by choosing the maximum travel distance and task durations for which the following values were found suitable:

$$\text{XMAX} = 500 \text{ ft.}$$
$$\text{TMAX} = 20 \text{ min.}$$

Returning to the point where a crane had just reached its target and integrator I1 was placed in the HOLD mode by the logic signal $A$, the same signal will also place the waiting time integrator I2 into the OPERATE mode to start the waiting period. Time $T$ is compared against the required time $\text{TWAIT}$ by comparator C2. When $T = \text{TWAIT}$ logic signal returns integrator I2 to the HOLD mode. This basically concludes the two phases of moving and waiting which will represent a continuous process between time $T_0$ and $T_2$.

At the end of the waiting time the logic signal C also sets a flip-flop FF1 by producing a pulse signal in the differentiator D1. A sense line is connected to the output LS of the flip-flop and is continuously sampled in the controlling program. If the logic value LS changes it indicates the end of a waiting period and the controlling program places the entire simulation in the HOLD mode.

This is done by connecting both integrators I1 and I2 to the computer mode signals.
as shown. Before describing the last phase of the crane task the function of the logic circuitry is summarised by a set of logic equations

where

\[ A, B, C, D \quad = \quad \text{logical true signal} \]
\[ \bar{A}, \bar{B}, \bar{C} \quad = \quad \text{logical false signal} \]
\[ \text{OP} \quad = \quad \text{computer operate mode signal} \]
\[ \text{IC} \quad = \quad \text{computer initial condition mode signal} \]

\[ \text{LM} \quad = \quad \bar{A} \quad \text{and} \quad B \]
\[ \text{LN} \quad = \quad A \quad \text{and} \quad \text{OP} \]
\[ \text{LO} \quad = \quad \bar{A} \quad \text{and} \quad \bar{C} \]
\[ \text{LP} \quad = \quad \bar{A} \quad \text{and} \quad \bar{C} \quad \text{and} \quad \text{OP} \]
\[ \text{LQ} \quad = \quad A \quad \text{or} \quad C \]
\[ \text{LR} \quad = \quad C \quad \text{and} \quad B \]
\[ \text{LS} \quad = \quad C \quad \text{or} \quad D \quad \text{(C = true when task is done until reset.)} \]
\[ D = \text{false at start of every OPERATE period} \]

**PHASE FOUR**

In the fourth phase the crane task is completed by executing an event caused by the crane activity. The following example will explain this: The action of moving to a converter and adding a ladle of matte to it has to be recorded by the converter model as a material addition. At the end of the time period that the crane waits at the target the material has thus been added but the converter model is unaware of this. An event code associated with the particular task is contained in the variable NEXT (I) for the particular crane while ISAVE(I)/100 contains the quantity of material if it is other than unity. The controlling program will use the value of NEXT(I) to branch to the respective submodel where
the state of the submodel will be updated accordingly. For example, the addition of a ladle of matte into converter No. 3 will correspond to an event number 80 if it originates from matte tunnel No. 1. This code will allow the controlling program to branch to that section of the converter submodel where the addition is recorded.

This completes the execution of one crane task and all other tasks will be executed in a similar way. Commands entered via the teletypewriter are encoded and a typical command

\[ \text{PDMT1, M1, C3} \]

would stand for "Pick up matte at matte tunnel No. 1, pour it into converter No. 3 and return the empty ladle to matte tunnel No. 1." This command in fact consists of three basic tasks (3 movements, with a time period for lifting or depositing at the end of each). The target positions, waiting times and priorities for each task are stored internally and need not be specified. A detailed description of command input and interpretation is given in Appendix A.

In the next section the continuous filing and retrieving of a series of crane tasks will be dealt with.
B.2 EXECUTION OF CRANE TASKS

Modelling of the crane tasks has been described in the previous section. In this section the control of a number of tasks waiting to be executed is to be described.

All crane tasks are issued via the teletype input if the model is operated manually. If automatic operation is desired, which is described in Chapter 7, the same type of tasks are used with one additional variable included. All tasks are then stored in the crane files numbered 2 for crane No. 1, 3 for crane No. 2, and 4 for crane No. 3. This is done in the same way as time events are stored in the time file which is labelled No. 1. Tasks are stored in the order in which commands are issued by means of the ranking variable NRANK. The term crane task is thus synonymous with crane event to use the terminology of the GASP simulation language. For this particular study the maximum number of attributes associated with each event was chosen to be four to save computer memory. It is necessary to use at least six attributes to describe a crane event uniquely which was achieved by encoding four integer variables before storing them in two locations with the remaining two floating point variables stored separately. This technique proved to be very useful, but is only possible if the range of values that a variable can have is limited. When an event is removed from the file for execution the values are then decoded again. Table B-2 lists the six variables with the constraints placed on their numerical values.

Filling a crane event is performed by subroutine CFILE, the flowchart of
which appears in Figure B-6, where the change from 6 to 4 attributes is also performed. The opposite is performed when retrieving an event which is done in subroutine GETEV shown in Figure B-7.

Assuming that all crane files are empty, that is there are no tasks to be performed by the cranes, then the busy light on the crane display will be turned off and the variable

\[ \text{IDLE}(\text{NCR}) = 0 \quad (\text{NCR} = \text{crane No.}) \]

signifies that crane No. NCR is idle. Following the issuing of a crane command, when the computer is still in the HOLD mode, the controlling program GASP performs a search of all crane files (see flowcharts of GASP in APPENDIX A).

If more than one crane file is not empty

i.e., \( MFE(\text{NCR} + 1) \neq 0 \)

the first crane events in each file will be executed in the order of their priorities.

For crane No. 1 having the highest priority this will take the following typical sequence:

1. Remove event from crane file (subr. GETEV)

2. Call resp. digital crane model to establish feasibility of executing task (subr. CRA1).

3. If feasible, set up analog model (subr. GOWT). If not feasible refile crane event with higher priority (subr. CFILE).

4. Repeat 1 - 3 for other cranes in decreasing order of priority.

5. Start simulation by placing computer in OPERATE mode.
Assuming that no other event occurs until the first crane task is completed, the simulation will run with all busy cranes performing their tasks in parallel. At this time a sense line in the analog crane model will change its logic level. During this period the digital program performs no other task than reading logic signals from the analog computer. The end of the crane task is thus recorded by the controlling program which places the analog computer in the HOLD mode to execute the fourth phase of the crane task as explained in Section B.1.

Thereafter the crane file of the particular crane is searched and the next event is executed in the same way as described above. In this way any crane will be busy as long as there are tasks to be performed. Once a particular crane file becomes empty,

\[ MFE (NCR + 1) = 0 \]

the busy indicator which was turned on is reset to indicate that the crane is now idle. The crane remains at the last target position until it has to move out of the way of another crane or when it receives another command. Apart from issuing one of several commands that occur regularly in actual practice such as moving a ladle of matte or copper, the crane models will respond to any arbitrary command where the choice of the target, the waiting time and priority can be chosen freely. These variables are checked and if they exceed the allowble limits they will be set to the maximum or minimum permissible value.
B.3 DIGITAL CRANE MODEL

If the combination of three cranes were to operate as described in Section B.2, they could be considered to be ideal models. In reality scheduled maintenance has to be performed on the cranes twice a week during which time they become inoperative. Furthermore breakdowns occur at irregular intervals when again a crane is operative for an undetermined duration. These aspects were included in the crane model by utilizing the event structure of the GASP discrete event simulation language. The problem of maintenance is thus modelled as follows. Since maintenance is performed at a predetermined time a crane maintenance event in the time file will schedule the maintenance for a given time. At this time a typeout will appear on the teletype "Crane is due for maintenance". The operator will respond to this by issuing the following command:

OIL (NCR = 1, 2, 3)

After interpretation an event is filed that will be executed five minutes from the time the command is given. This event is a crane maintenance event causing a crane to become inoperative for the next 90 minutes, the average time required for maintenance. At the same time information will appear on the teletype as to remind the operator when the crane will be operational again. Also the light indicator for the operational state will be turned off and the crane is now unable to accept any commands. If the command OIL is not issued within 30 minutes of the time the crane became due for maintenance, then a reminder to
operator will appear every 30 minutes on the teletypewriter until the maintenance is performed. After the maintenance period has elapsed the crane will be operational again until the next maintenance period comes up.

Irregular breakdowns are dealt with in a similar fashion, except that the time and duration of breakdowns are not deterministic quantities. Separate statistics on both quantities for each crane are used and random samples are drawn to determine breakdowns for each crane. This technique of random sampling from operating statistics is used frequently in the model and needs no further explanation here.

At the beginning of any simulation run crane failure times are generated. It was found that a uniform distribution is the best approximation to represent the breakdown behaviour of the cranes. If this time falls within the simulation period of the particular run the crane will break down and the operator is informed of this by a message on the teletypewriter. At the same time the repair time is sampled from a normally distributed function and the expected duration of the breakdown will also be supplied to the operator. Depending on the length he will then decide to use another crane or just wait until the crane is operational again. As in the case of scheduled maintenance the operational light is turned off and the no command can be issued to the crane until it is operational again.
8.4 CRANE STATISTICS

During any simulation run statistics on the cranes are accumulated and can be obtained at any time by giving the command

INFOK (NCR)

where NCR is the crane number. A call to submodule 5 will provide a tabular listing of all statistics together with all variables describing the crane state. A typical typeout is shown on Table B-3. Shown are the variables of the task presently executed as well as those of the previous task. They are rearranged slightly and the value of KODE is shown elsewhere (the previous event number is not shown as it is always reset to zero after executing the previous event). The three statistical quantities recorded are:

1. busy time (%)  
2. delay time (%)  
3. down time (min.)

Also shown are the attributes of the next events for each of the three cranes. Where the variables described before give information on the present and previous events, the attributes above contain the same information on the 1st future event.

A complete listing of all future events can be obtained for each crane by issuing the command:

INFOGP
This provides the information on all three crane files as well as the time file.

Table B-4 shows a typical typeout of the filing array together with relevant variables. The starting point of each file is given and by using column 5 and 6 of the array, it is very easy to obtain a full picture of the number and type of events in each file. Column 5 gives the line number of the succeeding event, while column 6 gives the preceding event. 9999 signifies the start of a file and 7777 indicates the end of a file.

Returning to Table B-3 the last two variables typed out KWT(I) and JWT(I) are state variables of the cranes, indicating whether one crane is waiting for another. Their functions can be described as follows:

\[
\text{JWT}(I) = 1
\]

\[
i = 1, 3 \quad l = 1, 3 \quad \text{indicates that crane No.} \ i \ \text{is delayed by crane No.} \ l
\]

and will continue when crane No. \ l \ has completed its present task.

\[
\text{KWT}(I) = 1
\]

\[
i = 1, 3 \quad l = 1, 3 \quad \text{indicates that crane No.} \ l \ \text{waits for crane No.} \ i \ \text{and will only continue when triggered by crane No.} \ l.
\]

(This variable allows synchronization of a two-crane operation which is used in the automatic operation.)

\[
\text{JWT}(I) = 0 \quad i = 1, 3 \quad \text{KWT}(I) = 0 \quad l = 1, 3
\]

signifies that no crane is delayed or waiting. Table B-5 lists all values input variables to the crane models. These values are fixed quantities during any simulation run.
B.5 CRANE COLLISIONS

While any collision is avoided by resolving crane interferences in the crane models a collision due to the slower crane speed of No. 2 crane could occur if it was followed by the faster No. 3 and No. 1 cranes. In practice the faster cranes would slow down in such a situation to the slower speed. Since it is only important in the simulation model, to ensure that the overall travelling time is the same, the faster crane could move at its maximum speed and then stop when it comes too close to the slower crane. After the slower crane has moved a far enough distance the other crane can proceed to its target. This concept is implemented in the analog models of the cranes. In Figure B-8 the circuit diagram is shown. The monostable timer is set to a suitable value and will delay the faster two cranes if they were to follow crane No. 2. In practice this does not occur very often but it might be important when the possible use of No. 2 crane is further investigated.
FIGURE B-1 FLOWCHART OF SUBR. CRA1

FIGURE B-2 FLOWCHART OF SUBR. CRA2
FIGURE B-3  FLOWCHART OF SUBROUTINE CRA3
FIGURE B-4 ANALOG CIRCUIT FOR CRANE MODELS
SUBROUTINE GOWT (NCR, POS, TW)

Define NCR, POS, TW

TW < .05
t

TW = .05

TW > TMAX
t

TW = TMAX

Read crane position, XCR

Scale Variables POS, TW

Unlatch Comparators
Set Target Position
Set Waiting Time

S = XCR - POS

S > .0012

Set direction of Movement

S < .0012

Set direction of Movement

Latch Comparators

RETURN
FIGURE B-6  FLOWCHART OF SUBR. CFILE

SUBROUTINE CFILE (NCR, POS, TW, IPR, NRANK, KODE, NXEVT, INPUT, NSET)

INPUT = 1

NRANK = NRANK - 1

FIRST ATTRIBUTE = POSITION

SECOND ATTB. = WAITING TIME

THIRD ATTR. = .1 * PRIORITY + RANK

FOURTH ATTR. = .1 * EVENT No. + 100 * KODE

FILE EVENT IN CRANE FILE

INPUT = 1

EVENT No. = 0

RETURN
FIGURE B-7  FLOWCHART OF SUBR. GETEV

SUBROUTINE GETEV(NSET)

REMOVE EVENT FROM FILE

TARGET POSITION = ATRIB(1)

WAITING TIME = ATRIB(2)

RANK = 4 LEFT DIG. OF ATRIB(3)

PRIORITY = RIGHT HAND
          DIGIT OF ATRIB(3)

KODE = 2 LEFT DIG OF ATRIB(4)

NEXT EVENT NO = 3 RIGHT
               DIGITS OF ATRIB(4)

STORE KODE IN ISTRIB(NCR + 1)

RETURN
FIGURE B-8 FEATURES OF ANALOG CRANE MODELS TO PREVENT COLLISIONS
<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PO(NCR)</td>
<td>present position</td>
</tr>
<tr>
<td>POO(NCR)</td>
<td>previous position</td>
</tr>
<tr>
<td>WT(NCR)</td>
<td>waiting time of present task</td>
</tr>
<tr>
<td>WTO(NCR)</td>
<td>waiting time of previous task</td>
</tr>
<tr>
<td>NRANK(NCR)</td>
<td>rank of last entry in crane file</td>
</tr>
<tr>
<td>LSTR(NCR)</td>
<td>rank of last event removed from crane file</td>
</tr>
<tr>
<td>TP(NCR)</td>
<td>priority of present task</td>
</tr>
<tr>
<td>IPO(NCR)</td>
<td>priority of previous task</td>
</tr>
<tr>
<td>NEXT(NCR)</td>
<td>event to follow execution of present crane task</td>
</tr>
<tr>
<td>IDLE(NCR)</td>
<td>state variable (busy or idle)</td>
</tr>
<tr>
<td>KAPUT(NCR)</td>
<td>state variable (operational or down)</td>
</tr>
<tr>
<td>NOIL(NCR)</td>
<td>state variable (maintenance required)</td>
</tr>
<tr>
<td>KODE</td>
<td>code used to retain different information on crane events</td>
</tr>
<tr>
<td>ISTRIB(NCR)</td>
<td>contains the value of code after event is removed from crane file</td>
</tr>
<tr>
<td>ISAVE(NCR)</td>
<td>retains 100% quantity of material picked up by a crane</td>
</tr>
<tr>
<td>JWT(NCR)</td>
<td>state variable (delayed or not)</td>
</tr>
<tr>
<td>KWT(NCR)</td>
<td>state variable (waiting or not)</td>
</tr>
<tr>
<td>NCR</td>
<td>Crane Number 1, 2, 3</td>
</tr>
<tr>
<td>TDOWN(NCR)</td>
<td>total time crane was broken down</td>
</tr>
<tr>
<td>TCRAN(NCR)</td>
<td>total time crane is busy</td>
</tr>
<tr>
<td>SP(NCR)</td>
<td>total time crane is delayed</td>
</tr>
<tr>
<td>TDL(NCR)</td>
<td>mean of crane repair duration</td>
</tr>
<tr>
<td>TDH(NCR)</td>
<td>standard deviation of crane repair duration</td>
</tr>
<tr>
<td>CRL(NCR)</td>
<td>lower limit of time between breakdowns</td>
</tr>
<tr>
<td>CRH(NCR)</td>
<td>upper limit of time between breakdowns</td>
</tr>
<tr>
<td>NAME</td>
<td>REPRESENTATION</td>
</tr>
<tr>
<td>--------</td>
<td>-------------------------------------</td>
</tr>
<tr>
<td>POS</td>
<td>Target position</td>
</tr>
<tr>
<td>TW</td>
<td>Waiting time</td>
</tr>
<tr>
<td>NRANK</td>
<td>Rank of task</td>
</tr>
<tr>
<td>IPR</td>
<td>Task priority</td>
</tr>
<tr>
<td>KODE</td>
<td>Code for different purposes</td>
</tr>
<tr>
<td>NXEVT</td>
<td>Event number to follow crane task</td>
</tr>
<tr>
<td>NCR</td>
<td>POS</td>
</tr>
<tr>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>1</td>
<td>62</td>
</tr>
<tr>
<td>2</td>
<td>469</td>
</tr>
<tr>
<td>3</td>
<td>376</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>NCR</th>
<th>ATRIB(1)</th>
<th>ATRIB(2)</th>
<th>ATRIB(3)</th>
<th>ATRIB(4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>10</td>
<td>-32768</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>13</td>
<td>-32768</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>10</td>
<td>-32768</td>
</tr>
</tbody>
</table>

**STATE VARIABLES**

- **WAITING(KWT)**: 0 0 0
- **INTERFERENCE(JWT)**: 0 0 0

**TABLE B-3  TYPICAL SIMULATION REPORT OF CRANE STATUS**
TABLE B-4  TYPICAL SIMULATION REPORT ON EVENT FIUNG
ARRAY NSET(6,1), WHERE I=1,60 AND RELATED VARIABLES

GASP VARIABLES

<table>
<thead>
<tr>
<th>ATTRIB(1-4)</th>
<th>1441.00</th>
<th>41.00</th>
<th>3.00</th>
<th>4.23</th>
</tr>
</thead>
<tbody>
<tr>
<td>ISTRIB(1-5)</td>
<td>3</td>
<td>2</td>
<td>12</td>
<td>3</td>
</tr>
<tr>
<td>NEXT EVENT</td>
<td>14461</td>
<td>2690</td>
<td>50</td>
<td>185</td>
</tr>
<tr>
<td>PREV. EVENT</td>
<td>1441.00</td>
<td>41</td>
<td>3</td>
<td>4</td>
</tr>
</tbody>
</table>

EVENTS IN NSET

<table>
<thead>
<tr>
<th>NO</th>
<th>POS</th>
<th>TW</th>
<th>RNK/IPR</th>
<th>KODE/NXTEV</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>19200</td>
<td>590</td>
<td>303</td>
<td>74</td>
</tr>
<tr>
<td>5</td>
<td>14710</td>
<td>410</td>
<td>10</td>
<td>42</td>
</tr>
<tr>
<td>6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>15220</td>
<td>590</td>
<td>10</td>
<td>6230</td>
</tr>
<tr>
<td>8</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>14550</td>
<td>410</td>
<td>22</td>
<td>190</td>
</tr>
<tr>
<td>10</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>11</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>12</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>13</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>14</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>15</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>17</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>18</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>19</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>20</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>21</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>22</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>23</td>
<td>14520</td>
<td>480</td>
<td>1205</td>
<td>178</td>
</tr>
<tr>
<td>24</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>25</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>26</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>27</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>28</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>29</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>30</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>31</td>
<td>14461</td>
<td>2690</td>
<td>50</td>
<td>185</td>
</tr>
</tbody>
</table>

FILE NAMES

<table>
<thead>
<tr>
<th>TIME</th>
<th>CR#1</th>
<th>CR#2</th>
<th>CR#3</th>
</tr>
</thead>
<tbody>
<tr>
<td>31</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

START OF FILES(MFE)

| 31   | 0    | 0    | 0    |
### TABLE B-5 - FIXED CRANE VARIABLES DURING A SIMULATION RUN WITH TYPICAL NUMERICAL VALUES

<table>
<thead>
<tr>
<th>Variable</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>BETA</td>
<td>20</td>
<td>speed up factor</td>
</tr>
<tr>
<td>XMAX</td>
<td>500 ft</td>
<td>maximum crane movement</td>
</tr>
<tr>
<td>TMAX</td>
<td>20 min</td>
<td>maximum length of crane waiting time</td>
</tr>
<tr>
<td>SP(1)</td>
<td>394 ft/min</td>
<td>ft/min crane speeds</td>
</tr>
<tr>
<td>SP(2)</td>
<td>330 ft/min</td>
<td>ft/min crane speeds</td>
</tr>
<tr>
<td>TOL</td>
<td>2 min</td>
<td>shortest started crane task that is not interrupted</td>
</tr>
<tr>
<td>W</td>
<td>20 ft</td>
<td>closest distance between cranes</td>
</tr>
</tbody>
</table>
APPENDIX C

REVERBERATORY AND MATTE TUNNEL SUBMODELS

C-1 EMPIRICAL REVERBERATORY AND MATTE TUNNEL MODELS

While no generally accepted mathematical models for reverberatory furnace operations are available, it is necessary to account for the reverberatory furnace response to material inputs and outputs from the converter aisle. Since the requirements of a reverberatory furnace submodel in the overall simulation of smelter operations are quite different to those of a model to be used for furnace control, the lack of exact theoretical representations of reverberatory furnace practice can be overcome.

Studies of reverberatory furnaces have been based on heat or material balances. The primary concern in developing a reverberatory furnace model, or for that matter any of the simulated submodels, was to use only existing measurable control variables to represent the behaviour of the process. Analysis of reverberatory furnace practice showed that an empirical model based on material balances would produce the most satisfactory representation of reverberatory furnace operation.

Three primary objectives were to be met by the reverberatory model all of which relate to its interaction with the converter aisle operation:

(1) Reproduce the rate of supply of matte of a given grade to the converter aisle.
(2) Represent the availability of both matte and slag in the reverberatory furnace.

(3) Account for the amount of copper lost in the reverberatory slag.

In the empirical model developed it was thus not necessary to account for variables which have no direct effect on above quantities. The usefulness of this might become evident when considering that a regression model of a reverberatory furnace included no less than 29 variables, many of which were difficult to measure.
C-2 ASSUMPTIONS IN THE REVERBERATORY FURNACE MODEL

The following assumptions were made in developing a model of the reverberatory furnace:

(1) The grade of matte from the reverberatory furnace is a fixed input quantity during simulation but can be varied if desired.

(2) The reverberatory furnace operation is a steady state operation, i.e., feeding, firing and consequently the smelting rate is constant unless it is decreased or increased externally.

(3) Matte is tapped from the matte tapping holes as long as there is enough matte in the furnace and as long as there is a demand for it in the converter aisle. (If there is no demand tapping can be temporarily suspended by not removing the last ladle from the matte car.)

(4) The duration of tapping a given quantity of matte or of reverberatory slag is not deterministic but can be represented statistically.

(5) There are definable limits to the levels of both slag and matte in the furnace within which the basic operating practice of slag and matte tapping are fixed.

(6) The assumptions made apply to both dry charged and wet charged furnaces.

Without attempting to give complete justification the following points will support the assumptions on the previous page:
In operating practice it is attempted to keep fluctuations of the matte grade to a minimum. If larger changes occur due to different ore grades, this change can be accounted for in the model by specifying a required higher or lower grade at any time during simulation.

The matte output of reverberatory furnaces is in direct response to the requirement of the converters. This demand is usually uniform. If a drastic decrease or increase occurs the rate of smelting in the reverberatory furnace models can be changed.

The assumption made under (3) corresponds very closely with operating practice.

The process of tapping matte involves several tasks that can hardly be described analytically. (Typically the time it takes to open a matte hole might depend among other variables on the hardness of the clay used to close the tapping hole). This assumption is supported by time studies that were taken of the matte tapping times. These are shown in Figures C-1 and C-2 where the tapping times show a variation very close to a normal distribution.

In practice these limits are not as exactly defined as they have to be in a mathematical representation but such limits exist. The main difference between reality and the model representation being the recognition that such limits have been reached. In practice properties of the material emerging from the tapping holes indicate whether or not such a limit is reached, whereas in the model direct
level measurements are recorded. In operating practice level measurements are taken every four or eight hours.

It should be pointed out that the lumped parameter approach in obtaining the smelting rate as a control variable does not need to account for the different material inputs but reflects a given different rate of production for each furnace. In the absence of a relationship between the smelting rate and variables such as rate of fuel consumption, rate of oxygen supply, furnace draught, furnace temperature, etc., this was the most accurate representation possible.
C-3 DESCRIPTION OF REVERBERATORY EVENTS

Although the matte tunnel operation is not part of the reverberatory furnace it is closely related to it and will be described as part of the model.

The model of the reverberatory furnace is composed of a sequence of logically related discrete events. If these events are executed as time advances at the respective times they will reproduce the furnace operation. The events are all stored in the time file with one common time attribute which indicates the time at which the event is to occur. For each of the three reverberatory furnaces the same events are used only with different subscripted variables.

The same does not apply for the matte tunnels, where there are separate events for each matte tunnel. This is necessary to account for the difference in the procedure of tapping matte from the furnaces (i.e. both reverberatory furnaces No.1 and No.2 supply matte to matte tunnel No.1 whereas only reverberatory furnace No.3 supplies matte to matte tunnel No.2).

In Table C-1 a list of events appears with a brief description of the operations performed by them. The abbreviations R1, R2, R3 and M1 and M2 are used to represent the reverberatory furnaces and matte tunnels. Logical and arithmetic operations performed in each event will now be described. Whenever variables are used they are subscripted and are calibrated differently for each furnace. Table C-2 lists and defines these variables.

1. (Event 41) A fixed smelting rate is represented by a given rate of matte level increase in each furnace, RMAT(NRE) in min/ins. This
event is initialised at the beginning of the simulation where the initial level of matte is specified as RMLEV(NRE). Every time this event is executed the matte level is increased by one inch. Based on the rate of matte production a future event is scheduled when the matte level will have increased again by one inch. The increments of one inch were chosen to produce an almost continuous variation of the matte level.

The matte level is compared to the allowable upper and lower limits. If any of these are reached the indicator lights "High Matte Level" or "No Matte Available" will be set to alarm the operator. At the same time slag or matte tapping is suspended, whichever occurs, high or low matte.

If the matte level has reached or exceeded its maximum allowable limit in all three furnaces a typeout will appear on the teletypeewriter giving the operator the option of:

(a) Increasing the matte grade, which will reduce the matte production,
(b) Decreasing the fire in a furnace to directly reduce the smelting rate.

Individual changes to each furnace are allowable and the resulting changed rates are shown for verification. In the absence of data relating an increase of matte grade to the decrease in smelting rate, the following estimate is used:

\[-4 \times (\% \text{ change in matte grade}) = \% \text{ change of smelting rate}\]
This equation expresses the response of the smelting rate as
estimated by operating personnel.

2. (Event 42) When a crane returns an empty ladle to matte tunnel No. 1,
the variable describing the state of the matte car

ITROL(ITUN) is checked. The states are given by:

ITROL(ITUN) = 1 - full ladle of matte on car at converter aisle
ITROL(ITUN) = -1 - car near matte tapping hole of rev.
ITROL(ITUN) = 0 - car empty at converter aisle

If ITROL(ITUN) = 1, or -1, no action is taken signifying that the
ladle is placed on the floor. If ITROL(ITUN) = 0 it is assumed
that the empty ladle is placed on the empty car which thereafter
moves the ladle to the matte tapping hole. Matte is tapped if
the "matte available" indicator is switched on in either reverberatory
furnaces R1 or R2.

i.e., if RMLEV(NRE) > YMMIN(NRE)

Matte is tapped from both furnaces if matte is available in both.
If no matte is available in one, matte is tapped in the other. If
the matte level has exceeded the upper level in only one, matte
is tapped there only. The time that passes before the filled ladle
of matte arrives at the converter aisle again is sampled from the
distribution shown in Figure C-1. The data shows times for
tapping matte from both furnaces and it is assumed that tapping
from one furnace takes twice as long, while the time of moving
the car to and from the converter aisle remains the same. This
time is a fixed variable given by TRAIL (ITUN) and could thus have been lumped into the distribution of tapping times. This was not done to account for breakdowns of the matte car which could delay the matte supply. In the absence of suitable statistics the effect was not considered significant but could be included in the model at any time.

If no matte is available in both furnaces tapping is suspended until the matte level has increased sufficiently. At that time tapping is resumed and the next time a ladle of matte becomes available is determined as described above.

3. (Event 43) The logic follows that of event 42 except that matte is only tapped from reverberatory furnace No. 3. The distribution of tapping time is shown in Figure 2.

4. (Event 44) Together with a light indicator "Matte has arrived" a bell ringing on the teletypewriter will inform the model operator of the arrival of a ladle of matte at matte tunnel No. 1. At this time the decrease of matte level due to tapping matte in the reverberatory furnaces is recorded, while light indicators for "low matte" are set if applicable. Furthermore the total number of ladles tapped are recorded as well as the time a full ladle stands on the matte car at the converter aisle.

5. (Event 45) This event for matte tunnel No. 2 is similar to event 44 except that it accounts for tapping of matte from reverberatory furnace No. 3 only. It might be noted that here the human interaction, i.e., a command from the model operator, is necessary to complete the cycle of tapping.
matte from the reverberatory furnace. In the actual operation a

6. (Event 46) crane will be sent to pick up matte from a matte tunnel if there

is one available and if a converter is ready to receive it. This

same operation is performed in the model if a crane is ordered
to pick up matte at a matte tunnel and to return the empty ladle
thereafter. Only then will the next ladle of matte be filled
and returned to the converter aisle to repeat the cycle.

In the automatic mode of operating the simulation model which is
described elsewhere, the crane command is issued automatically
after the need for matte is established and a converter is chosen
to give a smooth converter operation.

Once a crane has picked up a ladle of matte at matte tunnel No. 1,

event 46 is triggered when the state of the matte car is set to "empty".

This is also indicated by a green light that is turned on at the matte
tunnel. The end of the period of waiting of a full ladle of matte
on the car is recorded here.

Even 46 is also used if matte is picked up from the floor. That is, if

\[ \text{IFLOR}(1) > 0 \]

and \[ \text{ITROL}(1) \neq 1 \]

i.e., if there is matte on the floor but no matte on the car, a ladle
is picked up from the floor. If there were no ladles of matte left
on the floor a light indicating this would be turned off. It is
assumed that ladles of matte are always full, the quantity multiplied
by 100 being stored by the variable

\[ \text{ISAVE}(\text{NCR}) = 100 \]
7. (Event 47) Event 47 is the equivalent of event 46 for matte tunnel No. 2.

8. (Event 48) The increase of the slag level in the reverberatory furnaces is mainly due to the returned converter slag, however there is also a certain amount of slag produced in the reverberatory furnace. This uniform rate of slag production is represented by the variable 

\[ \text{RSLG(NRE)} \quad \text{in} (\text{ins./min.}) \]

which is used to update the slag level in a way similar to the matte level. If the slag level reaches its lower or upper limit, a light indicator will inform the operator. Event 48 is also used to perform the operation of slag tapping.

The quantity of slag tapped varies due to variations in slag viscosity and due to blocking of tapping holes. Available statistics showed that the variations of tapping rates can be best represented by uniform distributions for each of the three furnaces.

9. (Event 49) If a crane returns converter slag to reverberatory furnace No. 1, the quantity of slag is given by the variable \( \text{ISAVE(NCR)} \) which is used to record the increase of the slag level in the furnace, \( \text{SLEV(NRE)} \).

If the slag level exceeds the upper limit,

\[ \text{SLEV(NRE)} > \text{SMAX(NRE)} \]

a light indicator will be turned on to inform the operator. On any subsequent occasion no converter slag can be accepted by the reverberatory furnace until the slag level has fallen below the limit. If the command of pouring slag into the reverberatory furnace is given while the "high slag level" indicator is turned on, a message
will appear on the teletypewriter to inform the operator of this, (i.e., REVERB #1 CANNOT ACCEPT CV.SLAG.). Each ladle of converter slag that is poured into the reverberatory furnace is counted and stored in TSLAG(NRE).

10. (Event 50, Event 51) The procedure of returning converter slag to reverberatory furnaces No. 2 and No. 3 is the same as that of No. 1.

It should be noted here that the distribution of converter slag between the reverberatory furnaces is performed by the model operator who can request information on the amount of slag returned to each at any time. If he wishes to distribute slag evenly or in any other fashion the statistics on the reverberatory furnace will allow him to control this as desired.

In the automatic operation a given distribution pattern can be specified according to which the converter slag will then be scheduled to return to the reverberatory furnaces.

11. (Event 52, Event 53) If a ladle of matte is removed from a matte car and placed on the floor near one of the matte tunnels the variable IFLOR(ITUN)

will be incremented and an indicating light "matte on floor" will be turned on. Again this case is treated in a similar way for both matte tunnels.
Table C-3 shows a typical typeout as obtained upon requesting information on the reverberatory furnaces and the matte tunnels. The commands are \texttt{INFOR\{NRE\}} and \texttt{INFOM\{ITUN\}} respectively. All relevant quantities are listed as shown and give a full status report on the matte supply.

The matte level relative to its tolerance limits would indicate the amount of available matte while a high slag level could indicate that either or both reverb slag tapping was slow and converter slag was poured into the furnace frequently.

The matte level in a reverberatory furnace at any time \( t \) can be expressed as follows:

\[
M(t) = f(t, r_m, q_i, l_i) \quad \ldots \quad (1)
\]

where

- \( M(t) \) = matte level at time \( t \) (ins)
- \( r_m \) = rate of matte smelting (ins/min)
- \( q_i \) = quantity of matte tapped in a time interval \( t_i \) (this value is normally distributed) (ins)
- \( l_i \) = logical condition during \( \Delta t_i \) (=1 if matte is required and available, = 0 if matte is unavailable or not required)

Thus after the period \( t_n = \sum_{i=0}^{n} \Delta t_i \) has elapsed,

\[
M(t_n) = M(t_0) + \sum_{i=0}^{n} r_m \Delta t_i - \sum_{i=0}^{n} q_i l_i \quad \ldots \quad (2)
\]
Equation (2) reflects the function of both equations (3) and (4) and the logical states as they occur in the course of simulation:

\[ R_{MLEV(NRE)} = R_{MLEV(NRE)} + 1.0 \times R_{MAT(NRE)} \quad \ldots (3) \]

\[ R_{MLEV(NRE)} = R_{MLEV(NRE)} - R_{ATI(NRE)} \quad \ldots (4) \]

In the reverberatory furnace models these equations are executed whenever necessary and provide the matte level at any time.

In a similar way the slag level is represented by:

\[ S(t) = f(t, r_i, r_s, q_i, \bar{t}_i) \quad \ldots (5) \]

where:

- \( S(t) \) = slag level at time \( t \) (ins)
- \( r_i \) = quantity of converter slag returned in \( \Delta t_i \) (ins)
- \( r_s \) = rate of slag production in furnace (ins/min)
- \( q_i \) = quantity of reverberatory slag tapped in time interval \( \Delta t_i \) (this value is uniformly distributed) (ins)
- \( \bar{t}_i \) = logical condition during \( \Delta t_i \) (= 1 if matte level is below its upper limit, 0 if above upper limit)

and considering again an elapsed period of time to be made up of \( n \) discrete time intervals:

\[ t_n = \sum_{i=0}^{n} \Delta t_i \]

then

\[ S(t_n) = S(t_0) + \sum_{i=0}^{n} r_i + \sum_{i=0}^{n} r_s \Delta t_i - \sum_{i=0}^{n} q_i \bar{t}_i \quad \ldots (6) \]

Equation (6) is represented in the model by equations (7), (8) and (9) and the state of the logical variable \( \bar{t}_i \).
SLEV(NRE) = SLEV(NRE) + 1.14 \times SLRA(NRE) \times ISAVE(NC) \quad \ldots (7)
SLEV(NRC) = SLEV(NRE) + RSLG(NRE) \quad \ldots (8)
SLEV(NRE) = SLEV(NRE) - POT(NRE) \times UNFRM \times STAPLE(NRE),
\quad STAPH(NRE) \quad \ldots (9)

The numerical constant in question (7) accounts for the difference in specific weights of reverberatory slag and converter slag.

In the case of slag tapping the level changes are not directly influenced by decisions of the model operator, i.e., a continuous but non-uniform rate of tapping is assumed as long as the matte level is not too high. If the matte level exceeds its upper limit slag tapping is discontinued to avoid the loss of high grade matte in the reverberatory slag.

The reverberatory furnace model described is based on material balances only. The input of concentrates or calcine to the furnace together with the rate of oil burning determines the smelting rate of matte and accounts for a small part of reverberatory slag production. Both the matte production and the slag production rates were obtained from operating statistics and proved to be a sensitive control variable for the reverberatory furnace.

Matte tapping from the furnaces is a critical operation as it directly affects the overall converter aisle performance. The operating statistics can be represented accurately in the model whereby a realistic rate of matte supply is produced.

Converter slag is returned to the furnaces in response to the operator commands. The operator will attempt to follow regular operating practice in distributing the slag between the furnaces. Reverberatory slag is tapped at a varying rate. The total quantity of matte processed and the Cu loss in the
reverberatory furnace slag are the important quantities to consider when estimating
the overall efficiency of smelter operation. When considering the advantages of
operating at a higher grade matte the benefits would have to be weighed amongst
others against the total Cu-losses as supplied by the reverberatory model.

While the level measurements might not reflect the exact volume of molten
material in a bath of varying area, they were used to provide the operator with the
same measurement that is used in reality.

During the model development and validation phases it was necessary to
constantly revise aspects of the model. This was a very useful process since it
gave a better understanding of all interactions involved, which when accounted for
produced an accurate and meaningful reverberatory furnace operation.
FIGURE C-2 DISTRIBUTION OF MATTE TAPPING DURATION FOR MATTE TUNNEL 2

MEAN = 23.46
STANDARD DEV. = 5.30
<table>
<thead>
<tr>
<th>Event No.</th>
<th>Description of Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>41</td>
<td>Updating of matte level in R1, R2, R3</td>
</tr>
<tr>
<td>42</td>
<td>Tapping of matte is initialised at M1</td>
</tr>
<tr>
<td>43</td>
<td>Tapping of slag is initialised at M2</td>
</tr>
<tr>
<td>44</td>
<td>Full ladle of matte is available at M1</td>
</tr>
<tr>
<td>45</td>
<td>Full ladle of matte is available at M2</td>
</tr>
<tr>
<td>46</td>
<td>Ladle of matte is removed from car or floor at M1</td>
</tr>
<tr>
<td>47</td>
<td>Ladle of matte is removed from car or floor at M2</td>
</tr>
<tr>
<td>48</td>
<td>Slag level is updated in R1, R2, R3</td>
</tr>
<tr>
<td>49</td>
<td>Slag skimming is performed in R1, R2, R3</td>
</tr>
<tr>
<td>49</td>
<td>Converter slag is returned to R1</td>
</tr>
<tr>
<td>50</td>
<td>Converter slag is returned to R2</td>
</tr>
<tr>
<td>51</td>
<td>Converter slag is returned to R3</td>
</tr>
<tr>
<td>52</td>
<td>Ladle of matte is deposited on floor near M1</td>
</tr>
<tr>
<td>53</td>
<td>Ladle of matte is deposited on floor near M2</td>
</tr>
<tr>
<td>Variable</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>RMLEV(NRE)</td>
<td>relative matte level in furnace (ins)</td>
</tr>
<tr>
<td>YMMAX(NRE)</td>
<td>upper limit of matte level (ins)</td>
</tr>
<tr>
<td>YMMIN(NRE)</td>
<td>lower limit of matte level (ins)</td>
</tr>
<tr>
<td>RMAT(NRE)</td>
<td>rate of matte level increase (min/ins)</td>
</tr>
<tr>
<td>TR(NRE)</td>
<td>interval between matte level increases (min)</td>
</tr>
<tr>
<td>TLO(NRE)</td>
<td>mean duration of matte tapping (min)</td>
</tr>
<tr>
<td>THI(NRE)</td>
<td>standard deviation of time duration of tapping (min)</td>
</tr>
<tr>
<td>TOLA(NRE)</td>
<td>total number of matte ladles tapped</td>
</tr>
<tr>
<td>RATI(NRA)</td>
<td>conversion factor (ins matte/ladle)</td>
</tr>
<tr>
<td>SLEV(NRE)</td>
<td>relative slag level in furnace (ins)</td>
</tr>
<tr>
<td>SMAX(NRE)</td>
<td>upper limit of slag level (ins)</td>
</tr>
<tr>
<td>SMIN(NRE)</td>
<td>lower limit of slag level (ins)</td>
</tr>
<tr>
<td>STAPL(NRE)</td>
<td>lower limit of slag tapping rate (pots/min)</td>
</tr>
<tr>
<td>STAPH(NRE)</td>
<td>upper limit of slag tapping rate (pots/min)</td>
</tr>
<tr>
<td>TOTSL(NRE)</td>
<td>number of slag pots tapped</td>
</tr>
<tr>
<td>RSLG(NRE)</td>
<td>rate of slag level increase (in/min)</td>
</tr>
<tr>
<td>SLRA(NRE)</td>
<td>conversion factor (ins slag/ladle)</td>
</tr>
<tr>
<td>POT(NRE)</td>
<td>conversion factor (ins slag/pot)</td>
</tr>
<tr>
<td>TSLAG(NRE)</td>
<td>number of converter slag ladles returned</td>
</tr>
<tr>
<td>TAW(ITUN)</td>
<td>arrival time of matte car at converter aisle (min)</td>
</tr>
<tr>
<td>TOTW(ITUN)</td>
<td>total time full ladle stands on matte car (min)</td>
</tr>
<tr>
<td>Variable</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>TRAIL(ITUN)</td>
<td>time duration of matte car transportation (min)</td>
</tr>
<tr>
<td>IFLOR(ITUN)</td>
<td>number of matte ladles on floor</td>
</tr>
<tr>
<td>ITROL(ITUN)</td>
<td>status of matte car</td>
</tr>
<tr>
<td>NSLAG(ITUN)</td>
<td>matte available at matte tunnels (automatic operation)</td>
</tr>
<tr>
<td>NRE = 1,3</td>
<td>reverberatory furnace numbers for R1, R2, R3</td>
</tr>
<tr>
<td>ITUN = 1,2</td>
<td>matte tunnel numbers for M1, M2</td>
</tr>
</tbody>
</table>
### Table C-3: Typical Simulation Report on Reverberatory and Matte Tunnel Submodels

#### Reverberatory Furnaces

<table>
<thead>
<tr>
<th></th>
<th>R1</th>
<th>R2</th>
<th>R3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Matte Level</td>
<td>19.54</td>
<td>19.76</td>
<td>23.16</td>
</tr>
<tr>
<td>Max. Level</td>
<td>21.00</td>
<td>21.00</td>
<td>25.00</td>
</tr>
<tr>
<td>Min. Level</td>
<td>16.00</td>
<td>15.00</td>
<td>19.00</td>
</tr>
<tr>
<td>Matte Tapped</td>
<td>35.00</td>
<td>37.00</td>
<td>38.00</td>
</tr>
<tr>
<td>Slag Level</td>
<td>43.36</td>
<td>55.52</td>
<td>49.74</td>
</tr>
<tr>
<td>Max. Level</td>
<td>57.00</td>
<td>56.00</td>
<td>57.00</td>
</tr>
<tr>
<td>Min. Level</td>
<td>43.00</td>
<td>43.00</td>
<td>43.00</td>
</tr>
<tr>
<td>Slag Tapped</td>
<td>95.71</td>
<td>88.51</td>
<td>46.74</td>
</tr>
<tr>
<td>Slag Returned</td>
<td>39.28</td>
<td>47.52</td>
<td>35.90</td>
</tr>
</tbody>
</table>

#### Matte Tunnels

<table>
<thead>
<tr>
<th></th>
<th>MT1</th>
<th>MT2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mt on Floor</td>
<td>3</td>
<td>3</td>
</tr>
<tr>
<td>Mt on Car</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Min Mt is Waiting</td>
<td>428.02</td>
<td>496.63</td>
</tr>
</tbody>
</table>
APPENDIX D

CONVERTER SUBMODEL

D.1 INTRODUCTION

The converter model described here was developed by L.K. Nenonen of the National Research Council and the description that follows was taken from Reference (N-4).

The converter model consists of:

(i) Analog circuitry to represent ongoing chemical activity during blowing period as well as detection of end points.

(ii) A digital computer program module (Submodule 3) which handles material additions to the converter.

(iii) A digital computer program module (Submodule 6) which handles material removals from the converter.

(iv) A digital computer program module (Submodule 7) which responds to end point events, converter breakdowns, collar pulling, etc.

All materials transported by the cranes are specified in terms of their weight and their Fe, Cu, and S contents, the balance being considered slag. When specific material additions are made, these compositions and weights are used in submodule 3 to calculate the corresponding quantities of FeS, Cu2S and slag if the converter is in its matte stage and Cu2S, Cu and slag if in the copper stage. Digital-Analog converters are used to transmit the effects of the material addition to the analog
computer. Similarly, when material is removed from a converter, using Submodule 6, the digital computer determines the corresponding quantities of FeS, Cu$_2$S, Cu or slag removed and updates the analog computer using the same Digital–Analog converters.
D.2 PROCESS CHEMISTRY AND CONVERTING RATES

The model of the matte stage behaviour accounts for the reactions:

\[ 2\text{FeS} + 3\text{O}_2 \rightarrow 2\text{FeO} + 2\text{SO}_2 \] ....(1)

\[ 2\text{FeO} + \text{SiO}_2 \rightarrow 2\text{FeO} \cdot \text{SiO}_2 \] ....(2)

which give:

\[ 2\text{FeS} + 3\text{O}_2 + \text{SiO}_2 \rightarrow 2\text{FeO} \cdot \text{SiO}_2 + 2\text{SO}_2 \] ....(3)

The assumptions made are as follows:

(1) Any free Cu entering with input materials oxidizes instantaneously, according to:

\[ 4\text{Cu} + 2\text{FeS} + \text{O}_2 \rightarrow 2\text{Cu}_2\text{S} + 2\text{FeO} \]

the resulting FeO combining with SiO\(_2\) to form fayalite according to (2).

(2) Any free Fe entering with input materials oxides instantaneously according to:

\[ 2\text{Fe} + \text{O}_2 \rightarrow 2\text{FeO} \]

the resulting FeO combining with SiO\(_2\) according to (2)

(3) Converter temperatures and fluxing conditions are such that magnetite is not produced according to:

\[ 6\text{FeO} + \text{O}_2 \rightarrow 2\text{Fe}_3\text{O}_4 \]

(4) Referring to equation (3), equilibrium conditions are assumed to occur at all times such that the rate of reaction is determined entirely by the rate of supply of oxygen.

Additions of free Cu and Fe are accommodated by the model, the necessary adjustments being made in the material addition portion of the computer program.
To calculate the rate of supply of oxygen to the reaction, two input variables are required:

\[ \text{AIR} = \text{Measured air flow rate entering the converter, SCFM} \]

\[ \text{EFF} = \text{Proportion of the air entering which actually reacts on its way through the converter. This efficiency factor accounts for loss of air through leakage and possible non-equilibrium conditions in the converter which lead to the escape of unreacted oxygen.} \]

Thus, the rate at which \( O_2 \) enters into reaction (3) is:

\[ R_0 = 0.21 \times \frac{\text{Pa}}{\text{RT}} \times \text{AIR} \times \text{EFF} \text{ pound-moles per minute} \]

\[ R_{02} = .000545 \times \text{AIR} \times \text{EFF} \text{ LBMM} \]

The rates of change of all other reactants and products of reaction (3) can be determined based on \( R_{02} \).

\[ R_{\text{FES}} = -\frac{2}{3} \times R_{02} \text{ LBMM} \]

\[ R_{\text{FES}} = -.00001597 \times \text{AIR} \times \text{EFF} \text{ TPM} \]

\[ R_{S_{02}} = +\frac{2}{3} \times R_{02} \text{ LBMM} \]

\[ R_{S_{02}} = +.00000884 \times \text{AIR} \times \text{EFF} \text{ TPM} \]

\[ R_{2\text{FE}_0.5\text{O}_2} = +\frac{1}{3} \times R_{02} \text{ LBMM} \]

\[ R_{2\text{FE}_0.5\text{O}_2} = +.0000185 \times \text{AIR} \times \text{EFF} \text{ TPM} \]

\[ R_{\text{SiO}_2} = -\frac{1}{3} \times R_{02} \text{ LBMM} \]

\[ R_{\text{SiO}_2} = -.00000548 \times \text{AIR} \times \text{EFF} \text{ TPM} \]

However, converter flux additions may not be made at the stoichiometric rate given by (8). To permit use of non-stoichiometric rates in the model, a coefficient \( FRAT \) is introduced such that the actual rate of silica addition is
RS102 = + .00000548 x AIR x EFF x FRAT TPM ....(9)

and when FRAT = 1, SiO₂ is added at exactly the rate demanded by reaction (2),
when FRAT > 1, excess SiO₂ is added, and when FRAT < 1, insufficient SiO₂ is added.
Since the flux is actually 68% SiO₂, the total rate of flux added is:

\[
RFLUX = \frac{RS102}{0.68} = \frac{.00000804 \times AIR \times EFF \times FRAT}{0.68} TPM ....(10)
\]
of which the balance of 32% is assumed to be inert materials which enter the slag.

In practice, flux is added in an intermittent fashion to achieve converter temperature control. However, flux additions are made continuously in the model at the rate given by (10) whenever the converter is blowing in the matte stage.

While the total flux addition per charge will be correct, the uniform distribution of flux addition over the entire matte stage will deviate somewhat from the distribution achieved in practice. The actual rate of change of slag in the converter can be determined from equations (7) and (10) as follows:

\[
RSL = 12FeO \times SiO₂ + (FRAT - 1) \times RS102 + RFLUX \times 0.32
\]
where term (1) accounts for fayalite production by (2)
(2) accounts for SiO₂ not involved in reaction (2), i.e., non-stoichiometric SiO₂
(3) accounts for the inert materials entering with the flux.

Substituting

\[
RSL = .0000185 \times AIR \times EFF + (FRAT - 1) \times .00000548 \times AIR \times EFF
+ .00000804 \times AIR \times EFF \times FRAT \times 0.32
= .000013 \times AIR \times EFF (1 + .62 FRAT) .... (11)
\]

Equations (5) and (11) are the key rate equations solved in the model for converter in the matte stage.
During the copper or finish stage, incoming oxygen reacts with \( \text{Cu}_2\text{S} \) as follows:

\[
\text{Cu}_2\text{S} + \text{O}_2 \rightarrow 2 \text{Cu} + \text{SO}_2
\]

Therefore, the rates are

\[
\text{RCU}_2\text{S} = -R_02 \text{ LBMM}
\]

\[
\text{RCU}_2\text{S} = -0.0000434 \times \text{AIR} \times \text{EFF} \times \text{TPM}
\]

and

\[
\text{RCU} = +2 \times R_02 \text{ LBMM}
\]

\[
\text{RCU} = 0.0000346 \times \text{AIR} \times \text{EFF} \times \text{TPM}
\]

Note that the rate of supply of oxygen to the reaction \( (12) \) is based on the same efficiency factor \( \text{EFF} \) as is used in the matte stage. Equations \( (13) \) and \( (14) \) are solved in the model for converters in the copper stage.

During the matte stage equations \( (5) \) and \( (11) \) are solved to produce FeS and slag.

Before solving or integrating these rate equations on an analog computer, decisions are required concerning amplitude scaling and time scaling. To transform the analog voltages to corresponding material quantities in tons, the variable \( \text{TPU(NCV)} \) is used, \( \text{NCV} \) being the converter number where \( \text{NCV} = 1 \) to 5 corresponds to converters 3 to 7. Simulation work to date has used \( \text{BETA} = 20 \), \( \text{TPU(NCV)} = 150 \) for all converters.

In the model program, the variables \( \text{AIR} \) and \( \text{EFF} \) are defined by elements of the two-dimensional array \( \text{CV} \) as follows:

\[
\text{CV(NCV,6)} = \text{AIR} \times 10^{-3}
\]

\[
\text{CV(NCV,7)} = \text{EFF}
\]

Substituting into the rate equations for RFES, RSL, the equations to be integrated become
RFES = \(-0.01597 \times CV(\text{NCV}, 6) \times CV(\text{NCV}, 7) \times \text{BETA/TPU(\text{NCV})}\) .... (15)

RSL = \(+0.013(1 + 0.62 \text{FRAT}) \times CV(\text{NCV}, 6) \times CV(\text{NCV}, 7) \times \text{BETA/TPU(\text{NCV})}\) .... (16)

Similarly, in the copper stage, the following equations must be integrated:

RCU2S = \(-0.0434 \times CV(\text{NCV}, 6) \times CV(\text{NCV}, 7) \times \text{BETA/TPU(\text{NCV})}\) .... (17)

RCU = \(+0.0346 \times CV(\text{NCV}, 6) \times CV(\text{NCV}, 7) \times \text{BETA/TPU(\text{NCV})}\) .... (18)

Actual solution or integration of these equations is achieved by the analog computer circuit of Figure D-1, identical for each converter model. Referring to Figure D-1, potentiometers, \(P_A, P_D\) are set to the values given by (15) and (16) while \(P_B\) and \(P_E\) are set to the values given by (17) and (18). \(P_F\) and \(P_G\) are set to 1/60 to convert units per minute to units per second, the computer time scales being based on units/second. These potentiometers are set at the start of a simulation period when all initial conditions are established.

Thus, once initial conditions have been established for a given blowing period via \(DA_A\) and \(DA_B\), control lines \(A, B\) are employed to switch the integrators to the operate mode where the outputs of \(I_A, I_B\) will change at rates determined by \(P_A, P_D\) in the matte stage or \(P_B, P_E\) in the copper stage.
D.3 RESPONSE OF BASIC CONVERTER MODEL TO DISCRETE EVENTS

Discrete events can influence these analog computer circuits via the DA or AD converters, comparators, control lines or computer mode control status lines. The types of discrete event which arise are:

(1) Material additions
(2) Material removals
(3) Start and end points of individual blowing periods
(4) Equipment maintenance and breakdowns

Such events can be triggered by operator commands, by commands from the digital computer automatic decision making algorithms, or by changes in state of comparators in the analog computer itself. In particular, when FeS or Cu₂S in a blowing converter reaches 1 ton, this is sensed by the corresponding comparator \( C_A \) in Figure D-1 which switches from a TRUE state to a FALSE state and in so doing, triggers a sequence of other events leading to the switching of the corresponding integrators \( I_A, I_B \) to the HOLD mode from the OPERATE mode. In the HOLD mode, voltages on \( I_A, I_B \) are held for recording until further action is taken arising from a subsequent material removal or addition event.

The following converter material additions are classified by elements of the IADD (J10, 4):

J10

(1) Hot Charge Matte from matte tunnel 1
(2) Wet Charge Matte from matte tunnel 2
(3) Transfer matte
(4) Copper pigs
(5) Scrap
(6) Reactor slag
(7) Flue dust
(8) Skulls
(9) High copper slag

Each category of addition is identified by a code J1O and is assigned four attributes: weight, % Cu, % Fe, and % S.

Compositions of material removed from the converters are also defined by elements of the LADD array:

<table>
<thead>
<tr>
<th>J1O</th>
</tr>
</thead>
<tbody>
<tr>
<td>(3)</td>
</tr>
<tr>
<td>(9)</td>
</tr>
<tr>
<td>(10)</td>
</tr>
<tr>
<td>(11)</td>
</tr>
</tbody>
</table>

Other converter model variables are given in Table D-1.

In the model, composition of transfer matte removed from a particular converter is calculated based on the composition of the converter matte, and the corresponding LADD array elements are updated. Thus, when the transfer matte is added to a second converter, these updated compositions are used. However, fixed compositions are used for high copper slag, regular slag, and blister copper.
Fifteen operating states are defined by the variable ISTAT(NCV) where NCV is the converter number. These states are as follows:

1. Filled for a matte blow and blowing
2. Blowing on copper
3. End of a matte blow, slag to be skimmed
4. End of a matte stage, slag to be skimmed
5. End of a copper stage, blister copper to be removed
6. Blowing on matte, able to take more matte
7. End of matte stage, slag skimmed, waiting for flue dust before starting copper stage.
8. End of copper stage, skull to be added before tapping blister copper
9. End of matte blow, slag has been skimmed, waiting for matte before blowing can start
10. Not used
11. Blowing on copper, cold copper needed
12. Collar being pulled
13. Converter down due to equipment breakdown or rebrickling
14. End of a matte blow, slag can be skimmed or more matte added and current blow continued
15. Gowing high, at least one ladle of slag can be skimmed

The classification of material addition events is given above and the event numbers for the nine possible categories is given in Appendix F. Corresponding to each event number a material addition event causes a branch to submodule 3 where the following operations are performed:
1. Branch to respective event statement number.

2. Establish legality of command. Depending on the converter status
   ISTAT (NCV) and the material category J1O the matrix I0(J1O, ISTAT)
   transfers action to one of the 9 categories of legal material addition.
   If an illegal command was issued a message is typed out and the command
   is ignored by branching to the RETURN statement.

3. Based on the material category the respective converter model variables
   and the analog circuit is updated to account for the addition. Based on
   the material compositions as given by array IADD the quantities are
   updated for a matte blowing stage or a copper blow.

4. During a matte stage every addition causes a check whether the accumulated
   quantity of copper (Cu2S) has reached a critical level after which a copper
   blow can be initialised. If this condition is satisfied a message is typed
   out which gives the operator the choice. If he decides to proceed with
   a copper blow the present blow will be the last matte blow, the going-high
   blow before proceeding with a copper stage.

   Two further events are treated in submodule 3, the start of a slag blow in
   a converter analog model and the start of a copper blow. The first occurs when
   sufficient matte has been added for a particular blow while a copper blow always
   starts when a ladle of flue dust has been added after all the slag is skimmed in the
   going-high blow.

   If a material removal event occurs a branch to submodule six follows, where
   the four categories of material removal are treated in a similar fashion as additions
   are recorded in submodule 3.
The difference being the checks of material quantities remaining after a removal. For each type of material removed one component is more critical than the other, i.e. Cu₂S for transfer matte, and slag is important in the slag skimming operation irrespective of other quantities. However, removal of a particular material category causes a removal of the correct proportions of each of its constituents. The composition is again given by the array IADD.

In addition to the removal events an event recording the presence of an empty ladle at a converter and an event to start a blow after skimming slag during the going-high blow is treated here. Whenever converter slag is skimmed from a converter this is rerouted to a converter instead of being returned to a reverb-furnace.

Submodule seven consists of ten types of converter events, these are also shown in Appendix F. These events respond to beginnings and end points of processes.

The end point of a blowing period is given by the analog model of a converter and depending on the converter status given by ISTAT(NCV) it will indicate the end of a copper or a slag blowing period. In the copper blow the demand for cold copper pigs is created in submodule seven by filing an event at time intervals, that are determined at the beginning of a copper blow. In a similar way an event is filed that indicates when enough slag has accumulated in the going high blow to warrant skimming. In both cases light indicators are turned on to which the operator can respond.

Collar pulling results in an addition of material to a converter which is treated by two events, the start and end of collar pulling. Converter breakdowns are treated by digital models utilising breakdown and repair statistics.
One further event to zero converter arrays IAFO, IRFO and MT is executed whenever a new converter cycle starts after removing all the copper or all transfer matte from a converter.

The three arrays together with the material quantities in the converters at any time, given by array CV, provide a complete status report on all converters. This information is automatically provided at the end of every converter cycle or upon request. A typical typeout is given on Table D-2.
FIGURE D-1 - ANALOG CIRCUIT FOR CONVERTER MODELS
<table>
<thead>
<tr>
<th>INDEX J</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>CV(I,J)</td>
<td>Converter No.</td>
</tr>
<tr>
<td>IADD(I,J)</td>
<td>Input material category</td>
</tr>
<tr>
<td>IO(I,J)</td>
<td>Input or output material category</td>
</tr>
<tr>
<td>IAFO(I,J)</td>
<td>Input material category</td>
</tr>
<tr>
<td>IRFO(I,J)</td>
<td>Input material category</td>
</tr>
<tr>
<td>MT(I,J)</td>
<td>Converter No.</td>
</tr>
<tr>
<td>ICV(I,J)</td>
<td>Converter No.</td>
</tr>
<tr>
<td>PCV(I,J)</td>
<td>Converter No.</td>
</tr>
<tr>
<td>IREP(I,J)</td>
<td>General Converter Data Storage array</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>INDEX I</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weight, % Cu, Fe, and S</td>
<td>Validity check and program directory for converter material additions or removals</td>
</tr>
<tr>
<td>Record of material additions during current converter charge</td>
<td>Record of material removals during current change cycle</td>
</tr>
<tr>
<td>Record of converter event time during current change cycle</td>
<td>Computer Hardware assignments</td>
</tr>
<tr>
<td>Potentiometer assignments</td>
<td>General Converter Data Storage array</td>
</tr>
</tbody>
</table>
**CONVERTER INPUT/OUTPUT STATISTICS AT TIME C**

<table>
<thead>
<tr>
<th>ADDITIONS</th>
<th>REMOVALS</th>
<th>STATUS</th>
</tr>
</thead>
<tbody>
<tr>
<td>NCV</td>
<td>DM</td>
<td>WM</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>6</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>7</td>
<td>5</td>
</tr>
<tr>
<td>7</td>
<td>9</td>
<td>3</td>
</tr>
</tbody>
</table>

**CONVERTER TIMES AT TIME1440.0**

STAGE * MATTE BLOW * CU * BRK * PULL * EP AT * PIG * MIN*

START BLOW 3+ 2+ ADD PER

M CU B E SB SNE B E SNE B SNE B SNE M CU INT LCU LM

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
<th>15</th>
<th>16</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**CONVERTER CONTENTS**

<table>
<thead>
<tr>
<th>NCV</th>
<th>CI2S</th>
<th>FES</th>
<th>SLAG</th>
<th>CU</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>38.6</td>
<td>15.1</td>
<td>19.0</td>
<td>.0</td>
</tr>
<tr>
<td>4</td>
<td>64.3</td>
<td>5.6</td>
<td>37.3</td>
<td>.0</td>
</tr>
<tr>
<td>5</td>
<td>.0</td>
<td>.0</td>
<td>.0</td>
<td>.0</td>
</tr>
<tr>
<td>6</td>
<td>66.6</td>
<td>9.5</td>
<td>34.2</td>
<td>.0</td>
</tr>
<tr>
<td>7</td>
<td>92.0</td>
<td>.0</td>
<td>6.4</td>
<td>14.1</td>
</tr>
</tbody>
</table>

**TABLE D-2  TYPICAL SIMULATION REPORT OF CONVERTER FURNACES**
APPENDIX E

ANODE FURNACE AND MISCELLANEOUS MODELS

E-1 ANODE FURNACE SUBMODEL

The submodels described below were developed by Dr. U.P. Graefe of the National Research Council and the description is taken from Reference G-4.

The operation cycle of an anode furnace consists of filling the furnace with copper, a ladle or less at a time until its holding capacity is reached or until the operator decides to start refining. The refining time is assumed to be proportional to the number of ladles of copper in the furnace. Upon completion of refining, the furnace awaits a casting command from the operator. Casting can only take place if an anode wheel is available. Furnace 6 casts onto anode wheel No. 3 while furnaces 4 and 5 share anode wheel No. 4. The actual casting time depends on the furnace and the quantity of copper in the furnace. After completion of casting a certain amount of time is required to get the furnace ready to accept copper again.

The furnaces can thus be modelled by the following six types of events:

1. Event 30 to 32, Pouring of copper into anode furnace 4, 5 or 6 respectively
2. Event 34, Start the refining process
3. Event 35, Furnace ready for casting
4. Event 36, Start casting
5. Event 37, Finished casting

To prevent issuing the refined and cast commands at the wrong time during the anode furnace cycle, the furnace and anode wheel states are saved in the three
element array IAFL (one element per furnace) and the two element array IAWHL (one element per anode wheel). The states are coded as follows:

\[
\begin{align*}
IAFL(i) &= 0 \text{ when waiting for copper} \\
&= 1 \text{ when full} \\
&= 2 \text{ when refining} \\
&= 3 \text{ when finished refining} \\
&= 4 \text{ when casting} \\
&= 5 \text{ when finished casting}
\end{align*}
\]

\[
\begin{align*}
IAWHL (j) &= 0 \text{ when not casting} \\
&= 1 \text{ when casting}
\end{align*}
\]

where \( i = 1, 2, 3 \) for furnaces 4, 5 and 6 respectively

and \( j = 1, 2 \) for anode wheel 3 or 4 respectively

Pouring of copper into a furnace results in updating of the amount of copper held, provided of course, that the furnace is in state 0. The 'more copper required' and 'furnace full' lights are switched appropriately and the furnace state is set to 1 when the present level differs by less than one ladle from full capacity.

A refining command for a particular furnace results in a check to see if the furnace is indeed ready to refine. If so, appropriate status lights and furnace state are changed, and an event No. 35 is scheduled using a furnace dependent uniform distribution and the amount of copper in the furnace to determine the refining time.

After completion of the refining process event 35 is executed which notifies the operator that the particular anode furnace is ready for casting. The operator originated casting command results in checking for the correct furnace state and availability of an anode wheel. If the furnace is indeed ready to cast, and a wheel
is available appropriate status indicators are set and an event No. 37 is scheduled

to occur at a time derived from a furnace dependent uniform distribution and the

amount of copper held in the furnace.

Completion of casting results in setting appropriate status indicators and

anode wheel and furnace states. The total number of anodes cast is updated by the

number of anodes just cast, and event No. 38 is scheduled, using a furnace dependent

uniform distribution for the time of occurrence of this event.

When event No. 38 is executed the furnace status is reset to 0, the copper

content is set to zero, the 'furnace ready for copper' light is turned on, and the

operator is notified. Table E-1 shows the format of information as obtained from the

simulation model in response to the command INFOA (4, 5, 6). Also shown on Table E-1

is the information on the various other submodels described in the next sections.
E-2 PILOT PLANT SUB-MODEL

Operations of the pilot plant as seen from the converter aisle consists of the production, from time to time, of two copper cakes and pilot plant slag. The copper cakes have to be removed by a crane. A full ladle of pilot plant slag has to be picked up, and after emptying the slag into a converter, the empty slag ladle has to be returned. Thus, the whole pilot plant can be modelled through the following five events:

1. Event 1, copper cakes are ready to be picked up.
2. Event 3, copper cakes have been picked up
3. Event 4, slag is ready to be picked up
4. Event 6, slag has been picked up.
5. Event 7, an empty slag ladle has been returned to the pilot plant.

Briefly, when copper cakes are ready the operator is notified, the total number of cakes is updated and the next event No. 1 is filed using uniform distribution for the cake production time.

The pick-up of cakes involves checking to see that cakes are indeed available for pick-up, and notifying the operator when all available cakes have been picked up.

When slag is ready the operator is notified of this fact and the next such event is scheduled at a future time determined from an appropriate uniform distribution.
Pick-up of slag involves some bookkeeping and changing the light indicator status for the pilot plant.

The return of the slag ladle just causes the 'empty ladle required' light to be turned off.
Copper pigs are produced by pouring blister copper into moulds. There are 30 such moulds in the pig bay and one ladle of copper produces about five pigs. After cooling for approximately three hours the pigs are ready for pick-up. They may be dumped either into a converter or transported to the pig pile. Pigs on the pile may be picked up later for dumping into a converter. Thus the copper pig submodel can be represented by five events.

1. Event 21. pigs have been poured
2. Event 27. Some pigs have cooled sufficiently to be picked up
3. Event 23. Pick up of pigs from pig bay.

Before pigs can be poured, a check is made to see if there are enough moulds available. If, after filling all available moulds there is still some copper left, the operator is notified of the amount of copper left. As soon as more than 26 moulds are filled, the 'moulds available' light is turned off. Pouring of copper pigs also results in scheduling an event No. 27 to occur at a fixed time TPCOOL minutes from now.

When the pigs have cooled sufficiently event No. 27 occurs which increments the number of cold pigs available and notifies the operator of the fact that cool pigs are available.

Pick-up of pigs from the pig bay results in a check to see if the required
number of pigs is available. If all pigs have been picked up, the 'pigs available' light is turned off, and the 'moulds available' light is turned on as soon as less than 26 moulds are occupied.

Dumping of pigs or copper cakes onto the pig pile results in updating the number of pigs on the pig pile. Note that no distinction is made here between copper pigs or copper cakes. As long as there is at least one pig on the pig pile, the 'pigs available' light is on. Pick-up of pigs from the pig pile results in a check to see if the required number of pigs is available, and decrementing the number of pigs held on the pig pile.
No attempt is being made to keep track of individual ladles. Whenever a ladle is filled with matte, any type of slag, transfer matte, or copper, a variable NLC which counts the number of ladle usages, is incremented by 1. When this variable reaches the value NFL which is an average number of ladle usages between bumping, a light is turned on to inform the operator that a ladle (any ladle) requires bumping. The operator may now deposit a ladle near the bumping block for cooling. This constitutes event No. 13. After having cooled for some time, the operator may command a crane to bump the ladle, so gaining a skull in the progress. This operation is performed in event No. 11.

This submodel keeps track of the number of skulls available at the bumping block. Execution of event No. 11 adds a skull and execution of event No. 17, the 'completion of a skull pick-up' removes a skull from the supply. Light No. 5 which indicates the availability of skulls, is switched off and on appropriately.
TABLE E-1. TYPICAL SIMULATION REPORT ON ANODE FURNACE AND MISCELLANEOUS SUBMODELS.

ANODE STATISTICS

<table>
<thead>
<tr>
<th></th>
<th>A4</th>
<th>A5</th>
<th>A6</th>
<th>TOTAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANODES CAST</td>
<td>0</td>
<td>1787</td>
<td>7</td>
<td>1794</td>
</tr>
<tr>
<td>FURNACE CAPACITY</td>
<td>14.58</td>
<td>15.53</td>
<td>17.08</td>
<td></td>
</tr>
<tr>
<td>COPPER IN FURNACE</td>
<td>0</td>
<td>0</td>
<td>16.88</td>
<td>16.8</td>
</tr>
<tr>
<td>TOTAL LDLS ADDED</td>
<td>0</td>
<td>15.38</td>
<td>23.88</td>
<td>38.4</td>
</tr>
<tr>
<td>FURNACE STATUS</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>TIME OF STATUS, START</td>
<td>0</td>
<td>625</td>
<td>1626</td>
<td></td>
</tr>
<tr>
<td>ANODE WHEEL STATUS</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

PIG PILE

YO OF PIGS = 16

PIG BAY

YO OF PIGS = 21
COOL PIGS = 21
TOWT. PIGS POURED = 41

JUMPING BLOCK

COOLING LADLES = 1
SKULLS = 7
TOWT. SKULLS PROD. = 16

PILOT PLANT

CAKES READY = 0
TOTAL NO OF CAKES = 0
LADLES OF SLAG RDY = 0
TOTAL LDLS OF SLAG PROD = 0
NO INFO AVAILABLE

PROTOTYPE PLANT

CAKE RDY = 0
TOTAL NO OF CAKES = 3
LDLS OF CU RDY = 0
TOTAL LDLS OF CU PROD = 0
APPENDIX F

In this Appendix the following reference lists for the overall model are given:

F.1 LIST OF SUBROUTINES
F.2 LIST OF EVENTS
F.3 LIST OF VARIABLES IN LABLED COMMON BLOCKS
F.4 LIST OF LIGHT INDICATORS FOR DISPLAY UNIT
F.5 LIST OF LABLED PAUSE STATEMENTS IN THE PROGRAMME
F.6 LIST OF SENSE SWITCH FUNCTIONS
<table>
<thead>
<tr>
<th>NAME</th>
<th>FUNCTION PERFORMED</th>
<th>SUBMOD. OF FILE NAME</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANODE</td>
<td>Searches for best anode furnace to receive Cu</td>
<td>9</td>
<td>24</td>
</tr>
<tr>
<td>ANTIME</td>
<td>Provides duration of anode furnace event</td>
<td>CRANE</td>
<td>29</td>
</tr>
<tr>
<td>BELLs</td>
<td>Rings bell of teletypewriter</td>
<td>CRANE</td>
<td>32</td>
</tr>
<tr>
<td>CDCK</td>
<td>Checks correctness of next event in crane tasks</td>
<td>CRANE</td>
<td>29</td>
</tr>
<tr>
<td>CFILe</td>
<td>Encodes crane task variables and files crane events</td>
<td>CRANE</td>
<td>32</td>
</tr>
<tr>
<td>CHECK</td>
<td>Checks priority of crane event and crane no.</td>
<td>CRANE</td>
<td>32</td>
</tr>
<tr>
<td>CLOCK</td>
<td>Converts simulation time (TIME) to hours and minutes</td>
<td>CRANE</td>
<td>33</td>
</tr>
<tr>
<td>CMMDG</td>
<td>Generates and files commands</td>
<td>4</td>
<td>11</td>
</tr>
<tr>
<td>CONCHK</td>
<td>Checks converter status</td>
<td>4</td>
<td>12</td>
</tr>
<tr>
<td>CONTNL</td>
<td>Sets and resets control lines</td>
<td>CRANE</td>
<td>32</td>
</tr>
<tr>
<td>CONV</td>
<td>Finds best converter to receive material</td>
<td>9</td>
<td>24</td>
</tr>
<tr>
<td>COREAD</td>
<td>Enables reading from buffer array core</td>
<td>CRANE</td>
<td>33</td>
</tr>
<tr>
<td>CRANE</td>
<td>Finds best crane</td>
<td>9</td>
<td>23</td>
</tr>
<tr>
<td>CRA1</td>
<td>Establishes feasibility of move by crane 1</td>
<td>CRANE</td>
<td>30</td>
</tr>
<tr>
<td>CRA2</td>
<td>Establishes feasibility of move by crane 2</td>
<td>CRANE</td>
<td>30</td>
</tr>
<tr>
<td>CRA3</td>
<td>Establishes feasibility of move by crane 3</td>
<td>CRANE</td>
<td>30</td>
</tr>
<tr>
<td>CVFORC</td>
<td>Produces forecast of matte requirements of converters</td>
<td>11</td>
<td>27</td>
</tr>
<tr>
<td>NAME</td>
<td>FUNCTION PERFORMED</td>
<td>SUBMOD OF FILE NAME</td>
<td>PAGE</td>
</tr>
<tr>
<td>----------</td>
<td>-----------------------------------------------------------------------------------</td>
<td>---------------------</td>
<td>------</td>
</tr>
<tr>
<td>DATAN</td>
<td>Initialises GASP simulation, reads GASP data</td>
<td>GASP</td>
<td>34</td>
</tr>
<tr>
<td>DRAND</td>
<td>Produces a pseudo random number</td>
<td>CRANE</td>
<td>33</td>
</tr>
<tr>
<td>ERROR</td>
<td>Types ERROR if GASP error occurs</td>
<td>GASP</td>
<td>35</td>
</tr>
<tr>
<td>EVNTS</td>
<td>Branches to a corresponding event of a submodel</td>
<td>MAIN</td>
<td>3</td>
</tr>
<tr>
<td>EXIT</td>
<td>Calls EXIT after GASP error occurs</td>
<td>GASP</td>
<td>35</td>
</tr>
<tr>
<td>EXEC</td>
<td>Reads any submodule from disc</td>
<td>CRANE</td>
<td>33</td>
</tr>
<tr>
<td>FILEM</td>
<td>Files GASP type event in any file</td>
<td>GASP</td>
<td>34</td>
</tr>
<tr>
<td>FIND</td>
<td>Finds event in a file</td>
<td>CRANE</td>
<td>29</td>
</tr>
<tr>
<td>FLEFND</td>
<td>Checks if all submodules exist on disc</td>
<td>MIAN</td>
<td>3</td>
</tr>
<tr>
<td>FNDCEV</td>
<td>Finds any crane event issued but not executed</td>
<td>9</td>
<td>24</td>
</tr>
<tr>
<td>GASP</td>
<td>Performs simulation control</td>
<td>MAIN</td>
<td>1</td>
</tr>
<tr>
<td>GETEV</td>
<td>Removes crane event from file and decodes event variables</td>
<td>CRANE</td>
<td>29</td>
</tr>
<tr>
<td>GOWT</td>
<td>Sets analog components of crane models</td>
<td>CRANE</td>
<td>30</td>
</tr>
<tr>
<td>GWAIT</td>
<td>Files crane events with increasing rank</td>
<td>CRANE 3</td>
<td>31</td>
</tr>
<tr>
<td>LADLE</td>
<td>Submodel of ladles - records usage of filled ladles</td>
<td>CRANE</td>
<td>32</td>
</tr>
<tr>
<td>LIGHT</td>
<td>Sets or resets light indicators on display unit</td>
<td>CRANE</td>
<td>32</td>
</tr>
<tr>
<td>LOCFND</td>
<td>Finds device location from corresponding code</td>
<td>CRANE</td>
<td>29</td>
</tr>
<tr>
<td>MATTE</td>
<td>Provides information on present availability of matte</td>
<td>9</td>
<td>24</td>
</tr>
<tr>
<td>NAME</td>
<td>FUNCTION PERFORMED</td>
<td>SUBMODEL PAGE</td>
<td>PAGE</td>
</tr>
<tr>
<td>--------</td>
<td>-------------------------------------------------------------</td>
<td>----------------</td>
<td>------</td>
</tr>
<tr>
<td>MTFORC</td>
<td>Produces forecast of matte availability</td>
<td></td>
<td></td>
</tr>
<tr>
<td>QRBADR*</td>
<td>Reads analog to digital converter value</td>
<td></td>
<td></td>
</tr>
<tr>
<td>QRSSL*</td>
<td>Reads logical signal of sense line</td>
<td></td>
<td></td>
</tr>
<tr>
<td>QRCPL*</td>
<td>Reads comparator logic state</td>
<td></td>
<td></td>
</tr>
<tr>
<td>QSIC*</td>
<td>Selects analog computer mode Initial Condition</td>
<td></td>
<td></td>
</tr>
<tr>
<td>QSECN*</td>
<td>Selects analog time scale SEC.NORMAL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>QSDLY*</td>
<td>Causes analog delay period</td>
<td></td>
<td></td>
</tr>
<tr>
<td>QWFLR*</td>
<td>Sets function relay true or false</td>
<td></td>
<td></td>
</tr>
<tr>
<td>QWJDAR*</td>
<td>Sets value on digital to analog converter</td>
<td></td>
<td></td>
</tr>
<tr>
<td>QWPR*</td>
<td>Sets value on potentiometer</td>
<td></td>
<td></td>
</tr>
<tr>
<td>QSH*</td>
<td>Selects analog computer mode</td>
<td></td>
<td></td>
</tr>
<tr>
<td>QSOP*</td>
<td>Selects analog computer mode</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RD</td>
<td>Accepts operator commands from typewriter input</td>
<td></td>
<td></td>
</tr>
<tr>
<td>REV</td>
<td>Selects best reverberatory furnace for conv. slag</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RMOVE</td>
<td>Removes GASP type event from a file</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RNORM</td>
<td>Finds a normal variate of a given distribution</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SET</td>
<td>Updates filing pointers for GASP array NSET</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SETEST</td>
<td>Accepts specific operator command for automatic MAIN operation (debug)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SETIME</td>
<td>Sets timing circuit</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SLTEST</td>
<td>Reads all logic links of analog submodels and sense switch (1)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SORT</td>
<td>Finds best furnace if tie exists</td>
<td></td>
<td></td>
</tr>
<tr>
<td>STATS</td>
<td>Produces typeout of information on any submodel</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NAME</td>
<td>FUNCTION PERFORMED</td>
<td>SUBMOD# OR FILE NAME</td>
<td>PAGE</td>
</tr>
<tr>
<td>---------</td>
<td>-----------------------------------------------------------------------------------</td>
<td>----------------------</td>
<td>------</td>
</tr>
<tr>
<td>TESTFL</td>
<td>Finds first time event and resets timing circuit if required</td>
<td>CRANE</td>
<td>29</td>
</tr>
<tr>
<td>TCSORT</td>
<td>Sorts material requirements chronologically</td>
<td>11</td>
<td>27</td>
</tr>
<tr>
<td>TISTAT</td>
<td>Finds converters in a particular status</td>
<td>9</td>
<td>24</td>
</tr>
<tr>
<td>TIME</td>
<td>Computes simulation time from analog timing</td>
<td>CRANE</td>
<td>31</td>
</tr>
<tr>
<td>TKODE</td>
<td>Delays crane operation for synchronisation of &quot;two crane&quot; operation</td>
<td>MAIN</td>
<td>2</td>
</tr>
<tr>
<td>TWOOCR</td>
<td>Selects combination of two cranes for a task</td>
<td>9</td>
<td>23</td>
</tr>
<tr>
<td>UNFRM</td>
<td>Finds uniform variate of a given distribution</td>
<td></td>
<td>33</td>
</tr>
</tbody>
</table>

System linkage routines used to set and read analog components.
## LIST OF EVENTS

<table>
<thead>
<tr>
<th>Event No.</th>
<th>Submodule</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>Copper cakes ready</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>Copper cakes ready at prototype plant</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>Cakes have been picked up</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>Slag ready</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>Copper cakes picked up at prototype</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>Slag has been picked up</td>
</tr>
<tr>
<td>7</td>
<td>2</td>
<td>Empty slag ladle has been returned</td>
</tr>
<tr>
<td>8</td>
<td>2</td>
<td>CU ready at prototype</td>
</tr>
<tr>
<td>9</td>
<td>2</td>
<td>CU picked up at prototype</td>
</tr>
<tr>
<td>10</td>
<td>2</td>
<td>Ladle returned to prototype</td>
</tr>
<tr>
<td>11</td>
<td>2</td>
<td>Completion of bumping</td>
</tr>
<tr>
<td>12</td>
<td>2</td>
<td>Pick up of scrap or flue dust</td>
</tr>
<tr>
<td>13</td>
<td>2</td>
<td>Ladle deposited for cooling</td>
</tr>
<tr>
<td>14</td>
<td>2</td>
<td>Anode slag picked up at A4</td>
</tr>
<tr>
<td>15</td>
<td>2</td>
<td>Anode slag picked up at A5</td>
</tr>
<tr>
<td>16</td>
<td>2</td>
<td>Anode slag picked up at A6</td>
</tr>
<tr>
<td>17</td>
<td>2</td>
<td>Skull has been picked up</td>
</tr>
<tr>
<td>18</td>
<td>2</td>
<td>Pull collar on A4</td>
</tr>
<tr>
<td>19</td>
<td>2</td>
<td>Pull collar on A5</td>
</tr>
<tr>
<td>20</td>
<td>2</td>
<td>Pull collar on A6</td>
</tr>
<tr>
<td>Event No.</td>
<td>Submodule</td>
<td>Event</td>
</tr>
<tr>
<td>----------</td>
<td>-----------</td>
<td>--------------------------------------------</td>
</tr>
<tr>
<td>21</td>
<td>2</td>
<td>Pigs have been poured.</td>
</tr>
<tr>
<td>22</td>
<td>2</td>
<td>Pick anode collar at A4</td>
</tr>
<tr>
<td>23</td>
<td>2</td>
<td>Picked up pigs at pig bay</td>
</tr>
<tr>
<td>24</td>
<td>2</td>
<td>Dumped pigs on pig pile</td>
</tr>
<tr>
<td>26</td>
<td>2</td>
<td>Picked up pigs at pig pile</td>
</tr>
<tr>
<td>27</td>
<td>2</td>
<td>Some pigs have cooled</td>
</tr>
<tr>
<td>28</td>
<td>2</td>
<td>Pick anode collar at A5</td>
</tr>
<tr>
<td>29</td>
<td>2</td>
<td>Pick anode collar at A6</td>
</tr>
<tr>
<td>30</td>
<td>2</td>
<td>Poured copper into anode F. No.4</td>
</tr>
<tr>
<td>31</td>
<td>2</td>
<td>Poured copper into anode F. No.5</td>
</tr>
<tr>
<td>32</td>
<td>2</td>
<td>Poured copper into anode F. No.6</td>
</tr>
<tr>
<td>33</td>
<td>2</td>
<td>Mould available at pig bay</td>
</tr>
<tr>
<td>34</td>
<td>4</td>
<td>Start refining</td>
</tr>
<tr>
<td>35</td>
<td>2</td>
<td>Furnace 4, 5 or 6 ready for casting</td>
</tr>
<tr>
<td>36</td>
<td>4</td>
<td>Start casting on a wheel No. 4 or 3</td>
</tr>
<tr>
<td>37</td>
<td>2</td>
<td>Finished casting on a wheel No. 4 or 3</td>
</tr>
<tr>
<td>38</td>
<td>2</td>
<td>Furnace 4, 5 or 6 ready for copper input</td>
</tr>
<tr>
<td>40</td>
<td>8</td>
<td>Scheduling and forecasting for automatic operation</td>
</tr>
<tr>
<td>41</td>
<td>8</td>
<td>Matte level rises in reverb.furnace</td>
</tr>
<tr>
<td>42</td>
<td>8</td>
<td>Empty ladle arrives at matte Tun. No. 1</td>
</tr>
<tr>
<td>43</td>
<td>8</td>
<td>Empty ladle arrives at matte Tun. No.2</td>
</tr>
<tr>
<td>44</td>
<td>8</td>
<td>Full ladle arrives at Tun. No. 1</td>
</tr>
<tr>
<td>45</td>
<td>8</td>
<td>Full ladle arrives at Tun. No.2</td>
</tr>
<tr>
<td>46</td>
<td>8</td>
<td>Crane removes full ladle from No. 1</td>
</tr>
<tr>
<td>Event No.</td>
<td>Submodule</td>
<td>Event</td>
</tr>
<tr>
<td>----------</td>
<td>-----------</td>
<td>--------------------------------------------------------------</td>
</tr>
<tr>
<td>47</td>
<td>8</td>
<td>Crane removes full ladle from No.2</td>
</tr>
<tr>
<td>48</td>
<td>8</td>
<td>Slag is skimmed from reverb.</td>
</tr>
<tr>
<td>49</td>
<td>8</td>
<td>Converter slag is returned to reverb.1</td>
</tr>
<tr>
<td>50</td>
<td>8</td>
<td>Converter slag is returned to reverb.2</td>
</tr>
<tr>
<td>51</td>
<td>8</td>
<td>Converter slag is returned to reverb.3</td>
</tr>
<tr>
<td>52</td>
<td>8</td>
<td>Deposit full ladle of matte near Mt. 1</td>
</tr>
<tr>
<td>53</td>
<td>8</td>
<td>Deposit full ladle of matte near Mt. 2</td>
</tr>
<tr>
<td>55</td>
<td>8</td>
<td>Crane failure</td>
</tr>
<tr>
<td>56</td>
<td>8</td>
<td>Daily maintenance scheduled</td>
</tr>
<tr>
<td>57</td>
<td>8</td>
<td>Execute daily maintenance</td>
</tr>
<tr>
<td>58</td>
<td>8</td>
<td>End of shift</td>
</tr>
<tr>
<td>59</td>
<td>8</td>
<td>Time display is updated</td>
</tr>
<tr>
<td>60-64</td>
<td>3</td>
<td>Add scrap</td>
</tr>
<tr>
<td>65-69</td>
<td>3</td>
<td>Add reactor slag</td>
</tr>
<tr>
<td>70-74</td>
<td>3</td>
<td>Add pigs</td>
</tr>
<tr>
<td>75-79</td>
<td>3</td>
<td>Add high copper slag</td>
</tr>
<tr>
<td>80-84</td>
<td>3</td>
<td>Add wet charge reverb matte</td>
</tr>
<tr>
<td>90-94</td>
<td>3</td>
<td>Add dry charge reverb matte</td>
</tr>
<tr>
<td>95-99</td>
<td>3</td>
<td>Add transfer matte</td>
</tr>
<tr>
<td>104</td>
<td>3</td>
<td>Start copper blow</td>
</tr>
<tr>
<td>105-109</td>
<td>3</td>
<td>Add flue dust</td>
</tr>
<tr>
<td>110-114</td>
<td>3</td>
<td>Add skull</td>
</tr>
<tr>
<td>115-119</td>
<td>3</td>
<td>Start slag blow</td>
</tr>
<tr>
<td>170-174</td>
<td>6</td>
<td>Remove high copper slag (automatic mode only)</td>
</tr>
<tr>
<td>175-179</td>
<td>6</td>
<td>Remove regular slag</td>
</tr>
<tr>
<td>Event No.</td>
<td>Submodule</td>
<td>Event</td>
</tr>
<tr>
<td>-----------</td>
<td>-----------</td>
<td>------------------------------------</td>
</tr>
<tr>
<td>180-184</td>
<td>6</td>
<td>Remove transfer matte</td>
</tr>
<tr>
<td>185-189</td>
<td>6</td>
<td>Remove blister copper</td>
</tr>
<tr>
<td>190-194</td>
<td>6</td>
<td>Return empty slag ladle to converter</td>
</tr>
<tr>
<td>200</td>
<td>6</td>
<td>Start blowing after going-high skim</td>
</tr>
<tr>
<td>250-254</td>
<td>7</td>
<td>End of slag stage</td>
</tr>
<tr>
<td>255-259</td>
<td>7</td>
<td>Start converter breakdown</td>
</tr>
<tr>
<td>260-264</td>
<td>7</td>
<td>End of converter breakdown</td>
</tr>
<tr>
<td>265-269</td>
<td>7</td>
<td>Cold CU needed</td>
</tr>
<tr>
<td>270-274</td>
<td>7</td>
<td>Indicate that slag may be skimmed on going high</td>
</tr>
<tr>
<td>275-279</td>
<td>7</td>
<td>Start collar pulling</td>
</tr>
<tr>
<td>280-284</td>
<td>7</td>
<td>End collar pulling</td>
</tr>
<tr>
<td>285-289</td>
<td>7</td>
<td>End of a blowing period</td>
</tr>
<tr>
<td>290-294</td>
<td>7</td>
<td>End of finish stage</td>
</tr>
<tr>
<td>295-299</td>
<td>7</td>
<td>Zero arrays for next charge</td>
</tr>
</tbody>
</table>
### Variables in 'Named Common' Blocks

**List of Variables in Common Block BLOK1**

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PO(3)</td>
<td>Present crane position.</td>
</tr>
<tr>
<td>WT(3)</td>
<td>Present crane task time.</td>
</tr>
<tr>
<td>IP(3)</td>
<td>Present crane task priority.</td>
</tr>
<tr>
<td>SP(3)</td>
<td>Crane speed for initialization.</td>
</tr>
<tr>
<td></td>
<td>Crane delay time during simulation.</td>
</tr>
<tr>
<td>POO(3)</td>
<td>Previous crane position.</td>
</tr>
<tr>
<td>WTO(3)</td>
<td>Previous crane task time.</td>
</tr>
<tr>
<td>IPO(3)</td>
<td>Previous crane task priority.</td>
</tr>
<tr>
<td>NC(3)</td>
<td>Crane number when selecting crane tasks from files.</td>
</tr>
<tr>
<td>NP(3)</td>
<td>Time when next ladle of matte is available.</td>
</tr>
<tr>
<td>JWT(3)</td>
<td>State of crane (delayed).</td>
</tr>
<tr>
<td>KWT(3)</td>
<td>State of crane (waiting).</td>
</tr>
<tr>
<td>TCRAN(3)</td>
<td>Crane busy time during simulation.</td>
</tr>
</tbody>
</table>
List of Variables in Common Block BLOK2

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ICAPA(3)</td>
<td>100 * ladle capacity of anode furnaces</td>
</tr>
<tr>
<td>ILADA(3)</td>
<td>100 * number of ladles of copper in anode furnaces</td>
</tr>
<tr>
<td>IBUFX(12)</td>
<td>X-buffer for reading operator commands.</td>
</tr>
<tr>
<td>IBUFY(12)</td>
<td>Y-buffer for reading operator commands.</td>
</tr>
<tr>
<td>NRANK(3)</td>
<td>Rank number with which last crane command was filed.</td>
</tr>
<tr>
<td>ISTRT(3)</td>
<td>Matte tunnel number from which crane picked matte.</td>
</tr>
<tr>
<td>LOCDEV(31)</td>
<td>Co-ordinates of components in converter aisle (feet, from south end)</td>
</tr>
<tr>
<td>IWTNGT(17,2)</td>
<td>Default waiting time for tasks in seconds.</td>
</tr>
<tr>
<td>IPRA (17,2)</td>
<td>Default priorities for tasks.</td>
</tr>
<tr>
<td>NLC</td>
<td>Number of full ladles carried since last bumping a ladle.</td>
</tr>
<tr>
<td>NFL</td>
<td>Number of ladle fillings before bumping is required.</td>
</tr>
<tr>
<td>NSKJ</td>
<td>Number of skulls at bumping block.</td>
</tr>
<tr>
<td>NCOOL</td>
<td>Number of cooling ladles at bumping block.</td>
</tr>
<tr>
<td>NPPB</td>
<td>Number of pigs in the pig bay.</td>
</tr>
<tr>
<td>NCOLP</td>
<td>Number of cool pigs in the pig bay.</td>
</tr>
<tr>
<td>NPPP</td>
<td>Number of pigs in the pig pile.</td>
</tr>
<tr>
<td>NCTOT</td>
<td>Number of cakes now ready at the pilot plant.</td>
</tr>
<tr>
<td>NCPPL</td>
<td>Number of cakes produced at pilot plant during simulation, including initial value of NCTOT.</td>
</tr>
<tr>
<td>NTANC</td>
<td>Total number of anodes cast.</td>
</tr>
<tr>
<td>ICNTRW</td>
<td>Control word to set control lines.</td>
</tr>
<tr>
<td>IER</td>
<td>Miscellaneous, usually error word in linkage routines.</td>
</tr>
<tr>
<td>ISAVE(3)</td>
<td>100 * content of ladles carried by cranes.</td>
</tr>
<tr>
<td>Code</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>NAPL</td>
<td>Total number of skulls produced during simulation.</td>
</tr>
<tr>
<td>NOIIL(3)</td>
<td>Cranes have been oiled if = 1.</td>
</tr>
<tr>
<td>IFLOR(2)</td>
<td>Ladles of matte on floor.</td>
</tr>
<tr>
<td>ITROL(2)</td>
<td>Status of ladle on matte car.</td>
</tr>
<tr>
<td>JJ</td>
<td>Dummy for cranes.</td>
</tr>
<tr>
<td>II</td>
<td>Dummy for cranes</td>
</tr>
<tr>
<td>IDLE(3)</td>
<td>Crane status, 0 is idle, 1 is busy.</td>
</tr>
<tr>
<td>KAPUT(3)</td>
<td>Crane status, 0 is operational, 1 is down for repair.</td>
</tr>
<tr>
<td>NEXT(3)</td>
<td>Event code for next event to be executed by a crane.</td>
</tr>
<tr>
<td>LSTR(3)</td>
<td>Rank of last event removed from crane file.</td>
</tr>
<tr>
<td>NSLAG(3)</td>
<td>Number of ladles of slag returned to reverber.</td>
</tr>
<tr>
<td>IAFFL(3)</td>
<td>Anode furnace status.</td>
</tr>
<tr>
<td>IAWHL(2)</td>
<td>Anode wheel status.</td>
</tr>
<tr>
<td>NEVC(5)</td>
<td>Converter blow number.</td>
</tr>
<tr>
<td>IADD(11, 4)</td>
<td>Weight and composition of converter input material.</td>
</tr>
<tr>
<td>ICV(5, 9)</td>
<td>Computer hardware assignments for converter models.</td>
</tr>
<tr>
<td>ISTAT(5)</td>
<td>Current converter status.</td>
</tr>
<tr>
<td>JSTAT(5)</td>
<td>Converter status at time of breakdown.</td>
</tr>
<tr>
<td>NSTAT(5)</td>
<td>Converter status, not used presently.</td>
</tr>
<tr>
<td>NAPLC(3)</td>
<td>Number of anodes cast per ladle of copper.</td>
</tr>
<tr>
<td>IHGS(5)</td>
<td>100 * ladle content of converter slag standing near converter.</td>
</tr>
<tr>
<td>IRFO(5, 5)</td>
<td>Quantities of materials removed from converter during the current charge.</td>
</tr>
<tr>
<td>IAF0(5, 15)</td>
<td>Quantities of materials added to the converters during the current charge.</td>
</tr>
</tbody>
</table>
MT(5, 20)  Converter blowing/non blowing and starting times during current charge.

NCV  Converter number 1 to 5.

NTLAD(3)  $100 \times$ total number of ladles of copper added to anode furnaces.

NANCST(3)  Total number of anodes cast.

IATIME(3)  Time when present anode furnace state was begun.

NTPPB  Total number of pigs poured in pig bay.

NCPT  Total number of cakes produced by prototype.

IRESL  Number of ladles of reactor slag presently available.

IPTCU  $= 1$ if copper presently available from prototype.

IPK(6, 2)  Equivalenced with JQ(12) to store material quantities for automatic operation (AO).

IDP(6, 2)  Equivalenced with JPR(12) to store priorities for automatic operation (AO).

NCPTT  Number of copper cakes presently available from prototype.

LCPU  Total ladles of copper produced by prototype.

LRS  Total ladles of slag produced by pilot plant.

IRS  Reverb furnace which received ladle of slag last.

IPRIT(5)  Priorities of converters to receive matte (automatic operation AO)

INHOUR  Initializes subroutine clock at start of simulation.

NACOL(3)  Number of anode collars available from anode furnaces.

ITCOR(15)  Ranking variable used in forecasting of time of matte requirements (AO)

IBLOW(15)  Blow number for matte requirements (AO).

MOP(5, 8)  Matte allocation to converters per blow (AO).

MTREQ(5)  Quantity of first matte requirement after forecast (AO).
<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MTAV(2)</td>
<td>Quantity of matte available at time of forecast (AO).</td>
</tr>
<tr>
<td>REP(5, 6)</td>
<td>Reserved for converter status report.</td>
</tr>
<tr>
<td>MTOR(15)</td>
<td>Matte requirement within forecasting period (AO).</td>
</tr>
<tr>
<td>IGN0(15)</td>
<td>Converter number corresponding to MTOR(AO).</td>
</tr>
</tbody>
</table>
List of Variables in Common Block BLOK3

TPIA, TPIB  Lower and upper limits of uniform distribution (LUD) of time between copper cakes.

TIPBLA, TPISLB Lower and upper LUD of time between slag ready.

TPCOOL Time in minutes for pigs to cool.

TAF6A, TAF6B) Lower and upper LUD of time for refining per ladle in.
TAF5A, TAF5B) anode furnaces 6, 5 and 4.
TAF4A, TAF4B) anode furnaces 6, 5 and 4.

TSCT4A, TCST4B) Lower and upper LUD of time for casting per ladle for
TCST5A, TCST5B) anode furnaces 4 and 5.

TAFR5A, TAFR4B) Lower and upper LUD of time to get anode furnace 4 or 5
TAFR5A, TAFR5B) ready after casting.

XMAX Maximum crane displacement in feet from south end.

BETA Speed-up factor for simulation.

TMAX Maximum crane task time in minutes.

TIMEX Maximum time interval between time events.

TLAST Time of execution of last time event.

TOL Length of shortest started crane task that is not interrupted by another crane task.

W Closest distance between cranes.

RMLEV(3) Relative matte level in reverb furnace (inches)

YMMAX(3) Upper limit of matte level in reverb furnace (inches)

YMMIN(3) Lower limit of matte level in reverb furnace (inches)

RMAT(3) Rate of matte level increase in (min/inch).

TR(3) Interval between matte level updating (minutes).
TLO(3), THI(3)  Mean and variance of matte tapping time in minutes.
TOLA(3)  Total number of ladles of matte tapped.
TAW(2)  Arrival time of matte car at converter aisle (minutes).
TOTW(2)  Total time full ladle stands on matte car (minutes).
TRAIL(2)  Time duration of matte car transportation (minutes).
RATI(3)  Conversion factor (inches of matte/ladle).
SLEV(3)  Relative slag level in furnace in inches.
SMAX(3), SMIN(3) Upper and lower limit of slag level (inches).
STAPL(3), STAPH(3) Upper and lower limit of slag tapping rate (pots/min.).
TOTSL(3)  Number of pots of slag tapped.
RSLG(3)  Rate of slag level increase (inches/min).
SLRA(3)  Conversion factor (inches of slag/ladle).
POT(3)  Conversion factor (inches of slag/pot).
TDOWN(3)  Total break-down duration of cranes.
TDL(3), TDH(3)  Lower and upper LUD of crane down time (minutes)
CRL(3), CRH(3)  Lower and upper LUD of operational time of cranes (minutes).
TCST6A, TCST6B  Upper and lower LUD for casting time on anode furnace 6.
TAFF6A, TAFF6B  Lower and upper LUD for getting anode furnace 6 ready after casting.
HOURS  Clock display time (hours).
TSLAG(3)  Ladles of converter slag returned to reverbs.
CV(5, 9)  Converter contents, blowing rates and capacities.
TPU(5)  Scale factor for converter model voltages.
TREQ(5)  Time of first matte requirement after forecast.
TAV(2)  Time of next matte arrival at converter aisle.
TCEND(5)  Forecasted time for start of copper blow in converters (AO).
TCOR(15)  Times of matte requirements within forecasting period (AO).
TCONT(5)  Reserved for automatic operation (AO).
List of Variables in Common Block BLOK4

NCR  Crane number

POS  Target position of crane

IPR  Priority of crane event.

TW   Waiting time of crane event.

KODE Various uses.

NXEVT Event to be executed at end of crane task.

IS(10) Buffer to store sense line and comparator states when switching to frozen time mode.

LOGVAL Logical variable, various uses.

NFRNC Anode furnace number.

NRE  Reverb furnace number

List of Variables in Common Block SAVE

NSET (6, 60) GASP filing array

MODSCT(13) Track and sector number of submodules on disk.

List of Variables in Common Block ARG

IX   Event number to be executed.

1DIX  First command syllable

ID2X  Second command syllable

ISET Type of input command (PK, DP, PD)

JSEQ(12) Type of input command (material)

JCMND(12) Used to communicate command and crane information between submodules 9 and 10.

NCRAN(12) Dummy, used in submodule 9 for converter number storage.

ICO(5) Flag to load submodules 9 to 12.

MODUL
## F.4 List of Model Display Lights

<table>
<thead>
<tr>
<th>Light Number</th>
<th>State of Process When Light is On</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Reset all lights</td>
</tr>
<tr>
<td>1</td>
<td>Copper cakes available at PI</td>
</tr>
<tr>
<td>2</td>
<td>Slag ready at PI</td>
</tr>
<tr>
<td>3</td>
<td>Ladle requires bumping</td>
</tr>
<tr>
<td>4</td>
<td>Ladle cooling prior to bumping</td>
</tr>
<tr>
<td>5</td>
<td>Skull available at bumping block</td>
</tr>
<tr>
<td>6</td>
<td>Pigs available at pig bay</td>
</tr>
<tr>
<td>7</td>
<td>Pigs available at pig pile</td>
</tr>
<tr>
<td>8</td>
<td>A4 more copper required</td>
</tr>
<tr>
<td>9</td>
<td>A5 more copper required</td>
</tr>
<tr>
<td>10</td>
<td>A6 more copper required</td>
</tr>
<tr>
<td>11</td>
<td>A4 full and waiting</td>
</tr>
<tr>
<td>12</td>
<td>A6 full and waiting</td>
</tr>
<tr>
<td>14</td>
<td>A4 ready for casting</td>
</tr>
<tr>
<td>15</td>
<td>A5 ready for casting</td>
</tr>
<tr>
<td>16</td>
<td>A6 ready for casting</td>
</tr>
<tr>
<td>17</td>
<td>Empty moulds in pig bay</td>
</tr>
<tr>
<td>18</td>
<td>Empty slag ladle required at PI</td>
</tr>
<tr>
<td>19</td>
<td>Full ladle of matte near MTI</td>
</tr>
<tr>
<td>LIGHT NUMBER</td>
<td>STATE OF PROCESS WHEN LIGHT IS ON</td>
</tr>
<tr>
<td>--------------</td>
<td>----------------------------------</td>
</tr>
<tr>
<td>20</td>
<td>Full ladle of matte near MT2.</td>
</tr>
<tr>
<td>21</td>
<td>Empty ladle required at matte tunnel 1.</td>
</tr>
<tr>
<td>22</td>
<td>Full ladle waiting at matte tunnel 1.</td>
</tr>
<tr>
<td>23</td>
<td>Same as 21 for tunnel 2.</td>
</tr>
<tr>
<td>24</td>
<td>Same as 22 for tunnel 2.</td>
</tr>
<tr>
<td>25</td>
<td>Matte available in No. 1.</td>
</tr>
<tr>
<td>26</td>
<td>Matte level too high in No. 1 (reaches slag hole).</td>
</tr>
<tr>
<td>27</td>
<td>Slag level is high.</td>
</tr>
<tr>
<td>28</td>
<td>Same as 25 for No. 2.</td>
</tr>
<tr>
<td>29</td>
<td>Same as 26 for No. 2.</td>
</tr>
<tr>
<td>30</td>
<td>Same as 27 for No. 2.</td>
</tr>
<tr>
<td>31</td>
<td>Same as 25 for No. 3.</td>
</tr>
<tr>
<td>32</td>
<td>Same as 26 for No. 3.</td>
</tr>
<tr>
<td>33</td>
<td>Same as 27 for No. 3.</td>
</tr>
<tr>
<td>34</td>
<td>Saved for possible 3rd tunnel.</td>
</tr>
<tr>
<td>35</td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>Crane 1 operational.</td>
</tr>
<tr>
<td>37</td>
<td>Crane 1 busy.</td>
</tr>
<tr>
<td>38</td>
<td>Crane 2 operational.</td>
</tr>
<tr>
<td>39</td>
<td>Crane 2 busy.</td>
</tr>
<tr>
<td>40</td>
<td>Crane 3 operational.</td>
</tr>
<tr>
<td>41</td>
<td>Crane 3 busy.</td>
</tr>
<tr>
<td>42-46</td>
<td>Matte blow, converter.</td>
</tr>
<tr>
<td>47-51</td>
<td>Copper blow, converter.</td>
</tr>
<tr>
<td>LIGHT NUMBER</td>
<td>STATE OF PROCESS WHEN LIGHT IS ON</td>
</tr>
<tr>
<td>-------------</td>
<td>--------------------------------------------------------</td>
</tr>
<tr>
<td>52-56</td>
<td>End of blow, converter</td>
</tr>
<tr>
<td>57-61</td>
<td>Additions required, converter</td>
</tr>
<tr>
<td>62-66</td>
<td>Empty ladle to be returned to converter</td>
</tr>
<tr>
<td>67-71</td>
<td>Skimming required, converter</td>
</tr>
<tr>
<td>72</td>
<td>Anode wheel No. 4 casting</td>
</tr>
<tr>
<td>73</td>
<td>Anode wheel No. 3 casting</td>
</tr>
<tr>
<td>74</td>
<td>Cakes ready at P.T.</td>
</tr>
<tr>
<td>75</td>
<td>Copper ready at P.T.</td>
</tr>
<tr>
<td>76</td>
<td>Empty ladle to be returned to P.T.</td>
</tr>
<tr>
<td>77-79</td>
<td>Spare</td>
</tr>
<tr>
<td>80-89</td>
<td>Not used</td>
</tr>
<tr>
<td>90-99</td>
<td>Used as control lines</td>
</tr>
</tbody>
</table>
### F.5 LIST OF PAUSES

<table>
<thead>
<tr>
<th>PAUSE NUMBER</th>
<th>CAUSE</th>
<th>ACTION TAKEN WHEN CLEARED</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Invalid command entered</td>
<td>Command ignored</td>
</tr>
<tr>
<td>2</td>
<td>Invalid anode furnace no.</td>
<td>Command ignored</td>
</tr>
<tr>
<td>3</td>
<td>Invalid crane number</td>
<td>Command ignored</td>
</tr>
<tr>
<td>4</td>
<td>Event code outside range 1-295</td>
<td>No event is executed</td>
</tr>
<tr>
<td>5</td>
<td>No such event</td>
<td>No event is executed</td>
</tr>
<tr>
<td>6</td>
<td>Set monostable times 4-0, 4-1 switch serves on display unit ON.</td>
<td>Execution continues</td>
</tr>
<tr>
<td>11</td>
<td>Next time target further than TIMEX minutes away (subr. SETIME).</td>
<td>Set next time target TIMEX minutes from now</td>
</tr>
<tr>
<td>17</td>
<td>ISET greater than allowable value in DISK 9</td>
<td>Call ignored</td>
</tr>
<tr>
<td>21</td>
<td>Subr. CONV</td>
<td>Call ignored</td>
</tr>
<tr>
<td>41</td>
<td>Incorrect branch to CRA3 routine</td>
<td>Branches to CRA1 routine</td>
</tr>
<tr>
<td>55</td>
<td>In event 55 an incorrect crane failure state</td>
<td>Crane failed</td>
</tr>
<tr>
<td>74</td>
<td>Error in FIND, column No. less than 1. (GASP error 89)</td>
<td>Not recoverable</td>
</tr>
<tr>
<td>75</td>
<td>Error in SET (GASP error 88)</td>
<td>Not recoverable</td>
</tr>
<tr>
<td></td>
<td>Error in FNDCEV, column No. less than 1</td>
<td>Not recoverable</td>
</tr>
<tr>
<td>76</td>
<td>Error in REMOVE, column No. less than 1 (GASP error 97)</td>
<td>Not recoverable</td>
</tr>
<tr>
<td>77</td>
<td>Overlap in filling array, Subr. FILEM (GASP error 87)</td>
<td>Not recoverable</td>
</tr>
<tr>
<td>PAUSE NUMBER</td>
<td>CAUSE</td>
<td>ACTION TAKEN WHEN CLEARED</td>
</tr>
<tr>
<td>--------------</td>
<td>--------------------------------------------</td>
<td>---------------------------</td>
</tr>
<tr>
<td>150</td>
<td>Pause allows setting analog components to 'fix' cranes</td>
<td>Execution continues</td>
</tr>
<tr>
<td>222</td>
<td>Converter status is larger than 15</td>
<td>ISTAT (NCV) = 1</td>
</tr>
</tbody>
</table>
### F.6 List of Sense Switch Functions

<table>
<thead>
<tr>
<th>Sense Switch No.</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Push to enter a command</td>
</tr>
<tr>
<td>2</td>
<td>Push to suppress typeout from module 2, 8, 'crane idle' messages and light messages</td>
</tr>
<tr>
<td>3</td>
<td>Push to suppress typeout of ATRIB from modules 3, 6, 7.</td>
</tr>
<tr>
<td>4</td>
<td>Push to suppress typeout from modules 3, 6, 7.</td>
</tr>
<tr>
<td>5</td>
<td>Reset to suppress typeout in subroutine TCODE.</td>
</tr>
<tr>
<td>6</td>
<td>Push before going to start program during error recovery, and release after ENTER has been typed. In automatic mode, push if single crane operation is desired, release if two crane tasks are desired.</td>
</tr>
<tr>
<td>7</td>
<td>Push for automatic mode of operation.</td>
</tr>
<tr>
<td>8</td>
<td>Suppress typeout from modules 9, 10, 11 and 12.</td>
</tr>
</tbody>
</table>
FIRST SUBMODULE

SECK FOR FIRST SUBMODULE

SUBROUTINE DISK

THIS IS THE FIRST SUBMODULE. ITS FILE NAME IS UUPIC.
IT READS IN DATA INTO COMMON BLOCK 1 AND 2, SETS POTS
AND BRINGS IN PART VARIABLES.

COMMON FRED, DREAD
COMMON TRED, SERE (1, 1, 1, 1)
COMMON NRED, SERE (1, 1, 1, 1)
COMMON VRED, SERE (1, 1, 1, 1)
COMMON MRED, SERE (1, 1, 1, 1)
COMMON ARED, SERE (1, 1, 1, 1)
COMMON FRED, DREAD
COMMON TRED, SERE (1, 1, 1, 1)
COMMON NRED, SERE (1, 1, 1, 1)
COMMON VRED, SERE (1, 1, 1, 1)
COMMON MRED, SERE (1, 1, 1, 1)
COMMON ARED, SERE (1, 1, 1, 1)
COMMON FRED, DREAD
COMMON TRED, SERE (1, 1, 1, 1)
COMMON NRED, SERE (1, 1, 1, 1)
COMMON VRED, SERE (1, 1, 1, 1)
COMMON MRED, SERE (1, 1, 1, 1)
COMMON ARED, SERE (1, 1, 1, 1)
COMMON FRED, DREAD
COMMON TRED, SERE (1, 1, 1, 1)
COMMON NRED, SERE (1, 1, 1, 1)
COMMON VRED, SERE (1, 1, 1, 1)
COMMON MRED, SERE (1, 1, 1, 1)
COMMON ARED, SERE (1, 1, 1, 1)
COMMON FRED, DREAD
COMMON TRED, SERE (1, 1, 1, 1)
COMMON NRED, SERE (1, 1, 1, 1)
COMMON VRED, SERE (1, 1, 1, 1)
COMMON MRED, SERE (1, 1, 1, 1)
COMMON ARED, SERE (1, 1, 1, 1)
COMMON FRED, DREAD
COMMON TRED, SERE (1, 1, 1, 1)
COMMON NRED, SERE (1, 1, 1, 1)
COMMON VRED, SERE (1, 1, 1, 1)
COMMON MRED, SERE (1, 1, 1, 1)
COMMON ARED, SERE (1, 1, 1, 1)
COMMON FRED, DREAD
COMMON TRED, SERE (1, 1, 1, 1)
COMMON NRED, SERE (1, 1, 1, 1)
COMMON VRED, SERE (1, 1, 1, 1)
COMM...
C CODE 24 CUMBER OF PICS ON CALES ON PIG FILL
24 WAYW-YWAY(ES)(E)1/100
ISAVENCF(10)
CALL LIGHT(12)
J .700
C EVENT 26. COMPLETION OF PIC'S PICK-UP AT PIG FILL
26 WAYW-YWAY(ES)(E)10
IF(BWPYF) .GT. 0.00
BWPYF .LLE. 0.01 TYPE 1407
ISAVENCF(10)-WWAY(ES)(E)1/100
CALL LIGHT(12)
J .700
IF(BWPYF .GT. 0.10 CALL LIGHT(12)
IF(BWPYF .LLE. 0.10 CALL LIGHT(12)
J .700
C EVENT 27. A NUMBER OF PICS HAVE COOLED SUFFICIENTLY
C TO BE PICKED UP
27 WAYW-NPINC(S)(E)1/100
IF(BWPYF .GT. 0.10 CALL LIGHT(12)
C EMPTY CANOE ONLY AFTER AT LEAST 10 PICS HAVE ACCUMULATED
C J .700
C EVENT 30. 31. OR 32 POURING A LOAD OF COPPER INTO AN
C NOOD FURNACE
30 WAYW-NFNC
IF(BWPYF .GT. 0.10 CALL LIGHT(12)
C NOODS AVAILABLE AT PIC DRY
33 WAYW-LIGHT(12)
J .700
C EVENT 35. NOODS COOL IN FURNACE 6, 8, OR 6 IS READY FOR CASTING
35 WAYW-LIGHT(12)
J .700
C EVENT 37. COMPLETION OF CASTING ON NOOD WEAR 4 OR 6
C J .700
37 WAYW-LIGHT(12)
IF(BWPYF .GT. 0.10 CALL LIGHT(12)
IF(BWPYF .LT. 0.10 CALL LIGHT(12)
C EVENT 38. NOODS ON FURNACE
38 WAYW-LIGHT(12)
J .700
C EVENT 40. NOODS ON FURNACE
40 WAYW-LIGHT(12)
J .700
C EVENT 41. NOODS ON FURNACE
41 WAYW-LIGHT(12)
J .700
C EVENT 42. NOODS ON FURNACE
42 WAYW-LIGHT(12)
J .700
C EVENT 43. NOODS ON FURNACE
43 WAYW-LIGHT(12)
J .700
C EVENT 44. NOODS ON FURNACE
44 WAYW-LIGHT(12)
J .700
C EVENT 45. NOODS ON FURNACE
45 WAYW-LIGHT(12)
J .700
C EVENT 46. NOODS ON FURNACE
46 WAYW-LIGHT(12)
J .700
C EVENT 47. NOODS ON FURNACE
47 WAYW-LIGHT(12)
J .700
C EVENT 48. NOODS ON FURNACE
48 WAYW-LIGHT(12)
J .700
C EVENT 49. NOODS ON FURNACE
49 WAYW-LIGHT(12)
J .700
C EVENT 50. NOODS ON FURNACE
50 WAYW-LIGHT(12)
J .700
C EVENT 51. NOODS ON FURNACE
51 WAYW-LIGHT(12)
J .700
C DETERMINE IF CONVERTER CAN GO HIGH
C IF CONVERTER CAN GO HIGH
C CLEAR START TIME AND CONVERTER TO HIGH BLOW
C IF CONVERTER CANNOT GO HIGH
C CLEAR START TIME AND CONVERTER TO LOW BLOW
C DECIDE TO GO TO HIGH OR LOW BLOW
C IF CONVERTER GOES HIGH GO TO 5422
C IF CONVERTER GOES LOW GO TO 5424
C DECIDE AUTOMATIC DECISION ON CONVERTER
C LARGE DENOM.-TOT. TINTH.
C LARGE TINTH-NEV.
C NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C LARGE TINTH-NEV.
C END SLUG BLOW NOT ENOUGH SLAG TO SEIR
2900 ESTATINC=9
ISET=7
CALL LIGHTEI="NCV+1"
J = 700
C END OF A FINISH STAGE IN C18-2901
2900 ESTATINC=9
ISET=7
CALL LIGHTEI="NCV+1"
J = 700
C ZERO INFO 2900 ANALYSIS PRIOR TO START OF NEW CHARGE
2900 NCVR=FLOAT(XTRAE[M],1,1)
NCVR
call 2900 JCI=1,1
IEND=JEND=0
2901 continue
C ATRIB=MOSPE,1,1,2,140,
CORE=
IESTATINC=12
IESTATINC=12
CALL LIGHTEI","C0"
C CONVING STATE REPORT
990 JEST=ESTATINC
E0-PE
WRITEID="991 NEW, JEST, JCI=0,0"
WRITEID="992" (XTRAE[1],1,1)
J = 700
991 FORMAT(10)
992 FORMAT(1,2)
2000 FORMAT(1,14,14,
1001 FORMAT(14)
END
EVENT CODE 42

EVENT CODE 44

EVENT CODE 46

EVENT CODE 48

EVENT CODE 49

EVENT CODE 50

EVENT CODE 51

EVENT CODE 52

EVENT CODE 53

EVENT CODE 54

EVENT CODE 55

EVENT CODE 56

EVENT CODE 57

EVENT CODE 58

EVENT CODE 59

EVENT CODE 60

EVENT CODE 61

EVENT CODE 62

EVENT CODE 63

EVENT CODE 64

EVENT CODE 65

EVENT CODE 66

EVENT CODE 67

EVENT CODE 68

EVENT CODE 69

EVENT CODE 70

EVENT CODE 71

EVENT CODE 72

EVENT CODE 73

EVENT CODE 74

EVENT CODE 75

EVENT CODE 76

EVENT CODE 77

EVENT CODE 78

EVENT CODE 79

EVENT CODE 80

EVENT CODE 81

EVENT CODE 82

EVENT CODE 83

EVENT CODE 84

EVENT CODE 85

EVENT CODE 86

EVENT CODE 87

EVENT CODE 88

EVENT CODE 89

EVENT CODE 90

EVENT CODE 91

EVENT CODE 92

EVENT CODE 93

EVENT CODE 94

EVENT CODE 95

EVENT CODE 96

EVENT CODE 97

EVENT CODE 98

EVENT CODE 99

EVENT CODE 100

EVENT CODE 101

EVENT CODE 102

EVENT CODE 103

EVENT CODE 104

EVENT CODE 105

EVENT CODE 106

EVENT CODE 107

EVENT CODE 108

EVENT CODE 109

EVENT CODE 110

EVENT CODE 111

EVENT CODE 112

EVENT CODE 113

EVENT CODE 114

EVENT CODE 115

EVENT CODE 116

EVENT CODE 117

EVENT CODE 118

EVENT CODE 119

EVENT CODE 120

EVENT CODE 121

EVENT CODE 122

EVENT CODE 123

EVENT CODE 124

EVENT CODE 125

EVENT CODE 126

EVENT CODE 127

EVENT CODE 128

EVENT CODE 129

EVENT CODE 130

EVENT CODE 131

EVENT CODE 132

EVENT CODE 133

EVENT CODE 134

EVENT CODE 135

EVENT CODE 136

EVENT CODE 137

EVENT CODE 138

EVENT CODE 139

EVENT CODE 140

EVENT CODE 141

EVENT CODE 142

EVENT CODE 143

EVENT CODE 144

EVENT CODE 145

EVENT CODE 146

EVENT CODE 147

EVENT CODE 148

EVENT CODE 149

EVENT CODE 150

EVENT CODE 151

EVENT CODE 152

EVENT CODE 153

EVENT CODE 154

EVENT CODE 155

EVENT CODE 156

EVENT CODE 157

EVENT CODE 158

EVENT CODE 159

EVENT CODE 160

EVENT CODE 161

EVENT CODE 162

EVENT CODE 163

EVENT CODE 164

EVENT CODE 165

EVENT CODE 166

EVENT CODE 167

EVENT CODE 168

EVENT CODE 169

EVENT CODE 170

EVENT CODE 171

EVENT CODE 172

EVENT CODE 173

EVENT CODE 174

EVENT CODE 175

EVENT CODE 176

EVENT CODE 177

EVENT CODE 178

EVENT CODE 179

EVENT CODE 180

EVENT CODE 181

EVENT CODE 182

EVENT CODE 183

EVENT CODE 184

EVENT CODE 185

EVENT CODE 186

EVENT CODE 187

EVENT CODE 188

EVENT CODE 189

EVENT CODE 190

EVENT CODE 191

EVENT CODE 192

EVENT CODE 193

EVENT CODE 194

EVENT CODE 195

EVENT CODE 196

EVENT CODE 197

EVENT CODE 198

EVENT CODE 199

EVENT CODE 200
ELEVENTH SUBMODULE

SUBJECT: ELEVENTH SUBMODULE

SUBMITTED DATE: [Blank]

SUBMITTED BY: [Blank]

THIS IS THE ELEVENTH SUBMODULE. ITS FILE NAME IS [UPPER1].

IT IS USED FOR FORECASTING AND AUTOMATICALLY SCHEDULING ACTIVITIES IN THE CONVERTER APRIL.

SUBJECT:

This manual provides instructions and guidelines for the eleventh submodule, which is used for forecasting and automatically scheduling activities in the converter April. It is crucial for anyone involved in the converter process to understand and follow these guidelines to ensure smooth operation and efficiency. The manual is intended to help users navigate the complexities of the submodule, ensuring that all necessary tasks are completed accurately and on schedule.

The submodule is designed to facilitate the planning and execution of various activities within the converter, such as the addition, removal, and management of materials and processes. It is essential for users to familiarize themselves with the submodule's features and functionalities to maximize its potential.

For the convenience of users, the submodule includes a comprehensive set of procedures, checklists, and guidelines. These resources are available in both digital and hardcopy formats, making it accessible to all stakeholders.

In conclusion, the eleventh submodule is a vital tool for those involved in the converter process. Its effective use can lead to improved efficiency, reduced errors, and a more streamlined workflow. Users are encouraged to familiarize themselves with the submodule's features and to utilize its resources to the fullest extent possible.

Upon completion of the submodule review, users should be able to:

1. Understand the structure and purpose of the eleventh submodule
2. Identify key components and their functions
3. Execute tasks and processes within the submodule
4. Troubleshoot and address any issues that arise
5. Continuously improve their understanding and usage of the submodule

By following the guidelines and procedures outlined in this manual, users can enhance their productivity and ensure the success of their projects within the converter environment.

ACKNOWLEDGEMENTS

The team behind the development and maintenance of this submodule extends their gratitude to the following contributors:

[List of contributors]

FOR THE ELEVENTH SUBMODULE

[Date]

C 21 GO TO (22,145,11)
C 22 KAT = 2
C
C 23 MSA IS TO BE INSERTED AFTER HAPI. MAKE HAPI THE PREDECESSOR OF
C HAPI AND HAPI THE SUCCESSOR OF MSA.
C 24 MSA(MSA,MHA) = HAPI
C HAPI(MSA,HAPI) = HAPI
C
C 25 IF HAPI IS NOT Reset TO 2, THERE IS NO SUCCESSOR OF MSA. POINTERS
C ARE UPDATED AT STATEMENT 9. IF KAT = 2, IT HAS Reset AND THE
C SUCCESSOR OF MSA IS HAPI.
C
C 26 GO TO (21,7,145,11)
C
C 27 REMOVEAL OF AN ITEM FROM FILE JG.
C
C 9.0 = Z.
C
C 28 UPDATE POINTERS SYSTEM TO ACCOUNT FOR REMOVAL OF MLK (JG). COLUMN
C PREVIOUSLY REMOVED IS ALWAYS SET TO MLK(JG) BY SUBROUTINE MOVE.
C
C 29 MLK = MLK(JG)
C
C 30 IF JG(JG) = HAPI, BE AND COLUMN REMOVED. LET JG EQUAL SUCCESSOR
C OF COLUMN REMOVED AND JG EQUAL PREDECESSOR OF COLUMN ADDED.
C 31 IF JG(JG) = KAT, MLK HAS LAST ENTRY. IF JG = HAPI, MLK HAS FIRST Entry.
C 32 MLK WAS NOT FIRST OR LAST ENTRY. UPDATE POINTERS SO THAT JG IS
C SUCCESSOR OF JG(JG) AND JG(JG) IS PREDECESSOR OF JG.
C
C 33 DO 10 I = 1,10
C 34 MLK = MLK(JG)
C 35 J = MLK(JG)
C 36 MLK(JG) = MLK(JG(JG))
C 37 J(JG) = MLK(JG)
C 38 MLK = MLK(JG(JG))
C 39 J(JG(JG)) = MLK
C 40 MLK(JG(JG)) = MLK(JG)
C 41 J(JG) = J(JG(JG))
C 42 MLK(JG(JG)) = MLK(JG)
C 10 CONTINUE
C
C 43 UPDATE POINTERS.
C
C 44 MSA(JG(JG)) = HAPI
C 45 HAPI(JG(JG)) = MLK(JG(JG))
C 46 IF (MHA(MSA)) = Z.
C 47 IF (HAPI(JG(JG))) = MLK
C 48 MLK(JG(JG)) = MLK(JG(JG))
C 49 MLK(JG(JG)) = MLK(JG(JG))
C 50 MLK(JG(JG)) = MLK(JG(JG))
C 51 MLK(JG(JG)) = MLK(JG(JG))
C
C 52 DO 10 I = 1,10
C 53 MLK = MLK(JG(JG))
C 54 J = MLK(JG(JG))
C 55 MLK(JG(JG)) = MLK(JG(JG))
C 56 J(JG(JG)) = MLK(JG(JG))
C 57 MLK(JG(JG)) = MLK(JG(JG))
C 58 J(JG) = J(JG(JG))
C 59 MLK(JG(JG)) = MLK(JG(JG))
C 60 CONTINUE
C
C 61 UPDATE FILE STATISTICS
C
C 62 MLK(JG(JG)) = MLK(JG(JG))
C 63 J(JG) = J(JG(JG))
C 64 MLK(JG(JG)) = MLK(JG(JG))
C 65 MLK(JG(JG)) = MLK(JG(JG))
C 66 MLK(JG(JG)) = MLK(JG(JG))
C 67 MLK(JG(JG)) = MLK(JG(JG))
C 68 DO 10 I = 1,10
C 69 MLK = MLK(JG(JG))
C 70 J = MLK(JG(JG))
C 71 MLK(JG(JG)) = MLK(JG(JG))
C 72 J(JG(JG)) = MLK(JG(JG))
C 73 MLK(JG(JG)) = MLK(JG(JG))
C 74 J(JG) = J(JG(JG))
C 75 MLK(JG(JG)) = MLK(JG(JG))
C 76 CONTINUE
C
C 77 CALL UPDATE(MSA(JG),JG)
C 78 CALL UPDATE(JG(JG),JG)
C 79 CALL UPDATE(JG(JG),JG)
C 80 CALL UPDATE(JG(JG),JG)
C 81 CALL UPDATE(JG(JG),JG)
C 82 CALL UPDATE(JG(JG),JG)
C
C 83 RETURN
C
C