

B.O.F. CONTROL

B.O.F. CONTROL

By

DOUGLAS J. BUCHANAN, B. Sc.

A Thesis

Submitted to the School of Graduate Studies

in Partial Fulfilment of the Requirements

for the Degree

Master of Engineering

McMaster University

May 1972

MASTER OF ENGINEERING
(Chemical Engineering)

McMASTER UNIVERSITY
Hamilton, Ontario

TITLE: B.O.F. Control

AUTHOR: Douglas J. Buchanan, B.Sc. (Chemistry and Mathematics)
(Roberts Wesleyan College, N. Chili, N.Y.)

SUPERVISOR: Dr. D. A. R. Kay

NUMBER OF PAGES: ix, 74

SCOPE AND CONTENTS:

A production Basic Oxygen Furnace installation at Dominion Foundries and Steel Ltd., Hamilton, Ontario is used to develop a method of dynamically predicting the time during steelmaking at which a specified bath carbon content will be reached.

The carbon evolution rate is calculated from continuous stack gas analysis and flow data. This information, combined with a known bath carbon content measured shortly before the end of the blow, is used to perform carbon mass balances. Bath carbon contents, calculated from these mass balances can be expressed as a function of time by a simple updated mathematical model which is then used to predict the end point time.

A new technique for determining stack gas flow rates by utilizing an injected chemical tracer is evaluated.

ACKNOWLEDGMENTS

The author wishes to express his gratitude to the following:

McMaster University for financial support received and Dofasco for sponsorship of the project.

Several members of the McMaster University Computing Centre for providing much needed advice with the programming.

Mr. Paul Curry for supplying the computer sub-routines used for plotting the graphs.

Dr. D. A. R. Kay, the research supervisor, for his continued guidance throughout all phases of the work.

Dominion Foundries and Steel Company, Ltd., for extensive use of plant equipment and generous assistance of the following departments:

Melt Shop Pipefitters

Research Department and Dr. B.A. Strathdee, Manager, Applied Research

Melt Shop Utilities personnel for valuable, experienced advice and co-operation for instrumentation work.

TABLE OF CONTENTS

CHAPTER 1. Basic Oxygen Furnace Steelmaking	1
1.1 Background	1
1.2 Chemistry of the Process	1
1.3 Equipment Design and Operation	2
CHAPTER 2. B.O.F. Control Systems	4
2.1 Advantages	4
2.2 General Principles	5
2.3 Control by a Chemical Tracer Technique	6
2.3.1 Stack Gas Flow	6
2.3.2 Chemical Analysis	6
CHAPTER 3. Control Equations	7
3.1 Carbon Evolution Rate	7
3.2 Bath Carbon Content	8
3.3 Modelling	8
3.4 Stack Lance Oxygen	9
CHAPTER 4. Experimental	10
4.1 Tracer Injection	10
4.2 Gas Sampling and Preparation	10
4.3 Instrumentation	11
4.3.1 Gas Analysis	11

4.3.2 Operating Variables	11
4.3.3 Time Delay	11
4.4 Data Collection and Processing	12
 CHAPTER 5. Results	 13
5.1 Presentation and Discussion	
5.1.1 Measured Variables	13
(a) Gas Analysis	13
(b) Lance Height and Oxygen Flow	14
(c) Methane Injection Temperature	14
5.1.2 Calculated Variables	14
(a) Stack Gas Flow	14
(b) Carbon Evolution Rate	15
(c) Bath Carbon Content	15
(d) PRL02	16
(e) UNUSDO2(t)	16
5.1.3 Mathematical Modelling	17
5.1.4 Variance Analysis	18
5.1.5 Sample Computer Printout	19
5.2 Conclusions	19
5.3 Recommendations	20
 APPENDIX I. Derivation of Control Equations and Error Analyses	 22
I.1 Propagation of Errors	22
I.1.1 Theory	22
I.1.2 Assumed Variance Values	23

1.1.3 Sample Calculation Data (HEAT #19923)	24
1.2 Flow Measurement	25
1.2.1 Methane Injection Rate	25
1.2.2 Stack Gas Flow	27
1.3 Carbon Evolution	28
1.3.1 Rate	28
1.3.2 Integration-Trapezoidal Rule	30
1.3.3 Bath Carbon Content	31
1.4 Oxygen Distribution	33
1.4.1 Unused Lance Oxygen	33
1.4.2 PRL02	36
APPENDIX II. Computer Program	38
II.1 Listing	38
II.2 Sample Printout	73

LIST OF TABLES

Table 2.1	Distribution of Total Oxygen	4a
Table 4.1	Time Delays	12
Table 5.1	Bath Carbon Contents	17d
Table 5.2	End Point Predictions	17d

LIST OF FIGURES

Figure 1.1(a) Bath Impurities During Blow	2a
Figure 1.1(b) Slag Analyses During Blow	2a
Figure 1.2 Sequence of Operations in a B.O.F.	2b
Figure 2.1 Bath Oxygen Content [O] as a Function of Bath Carbon Content [C]	4a
Figure 2.2(a) Slag Analysis During a No-Slop Heat	4b
Figure 2.2(b) Slag Analysis During a Slop Heat	4c
Figure 4.1 Tracer Injection Apparatus	10a
Figure 4.2 Sampling and Preparation System	10b
Figure 4.3 Gas Analysis and Data Collection	11a
Figure 5.1 CO, CO ₂ , and O ₂ Stack Gas Analyses	13a
Figure 5.2 Lance Height Above Static Bath During a Blow	14a
Figure 5.3 Lance Oxygen Flow	14b
Figure 5.4 Methane Injection Temperature	14c
Figure 5.5 Stack Gas Flow	14d
Figure 5.6 Carbon Evolution Rate	15a
Figure 5.7 Carbon Evolution Rate After Lance O ₂ Shut Off	15b
Figure 5.8 Bath Carbon Content (After Datum Point)	15c
Figure 5.9 PRL02	16a
Figure 5.10(a) Carbon Evolution (Slopping Heat)	16b
Figure 5.10(b) PRL02 (Slopping Heat)	16c
Figure 5.11 Lance O ₂ Flow During Slopping	16d
Figure 5.12 Lance O ₂ Used for CO Combustion in B.O.F.	17a
Figure 5.13(a) Predicted Bath Carbon Curves (HEAT #19926)	17b

Figure 5.13(b) Predicted Bath Carbon Curves (HEAT #19929) 17b

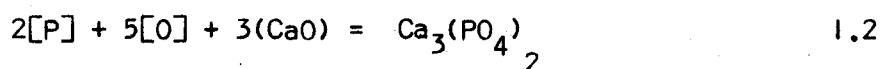
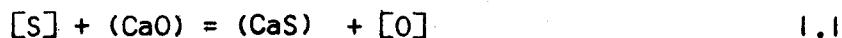
Figure 1.1 Flowmeter Calibration Chart 26a

CHAPTER I

BASIC OXYGEN FURNACE STEELMAKING

1.1 Background

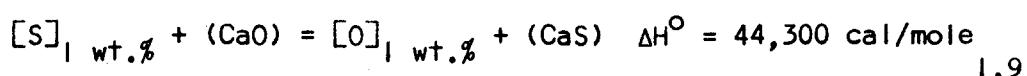
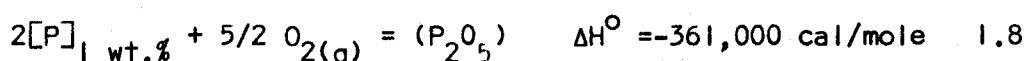
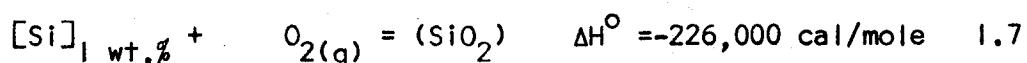
Over-reduced hot metal, containing approximately 4.5% C, 1.0% Si, 1.0% Mn, < 0.01% P, and 0.02% S by weight, is supplied by the blast furnace. Oxidation, using commercially pure (99.5%) oxygen is then carried out in the B.O.F. under chemically basic conditions to remove the first four elements. Equations 1.1 and 1.2⁽¹⁾, which describe desulphurization and dephosphorization respectively, demonstrate the necessity for basic conditions.



[] represents a component in the metal and () a component in the slag. Desulphurization, a reducing reaction, is relatively inefficient under oxidizing steelmaking conditions.

1.2 Chemistry of the Process

Steelmaking reactions at 1600°C may be summarized by the following equations^(1,2):



$[\square]_{1 \text{ wt.\%}}$ indicates that component activities are referred to the standard state of a 1 wt.% solution. The first six exothermic reactions generate enough heat to make Basic Oxygen Steelmaking an autogeneous process. Typical graphs of composition change in the bath and slag during the blow are illustrated in Figures 1.1(a) and 1.1(b)⁽³⁾ respectively.

1.3 Equipment Design and Operation

The furnace rotates about a fixed horizontal axis to allow for various operating positions such as charging of new materials, blowing and tapping of steel and slag. Figure 1.2 illustrates the sequence of operations. Oxygen is injected at supersonic velocities through a retractable water-cooled lance which is lowered

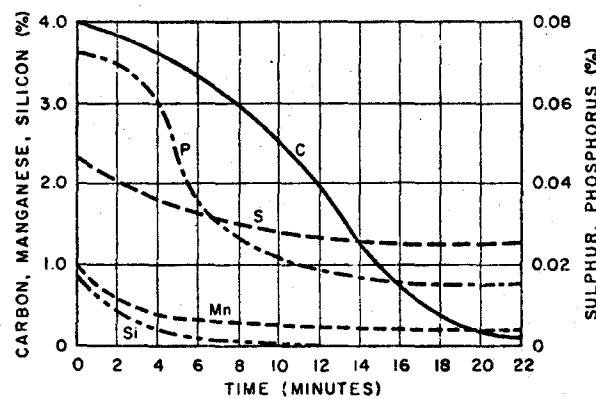


Figure 1.1(a) Bath Impurities During Blow

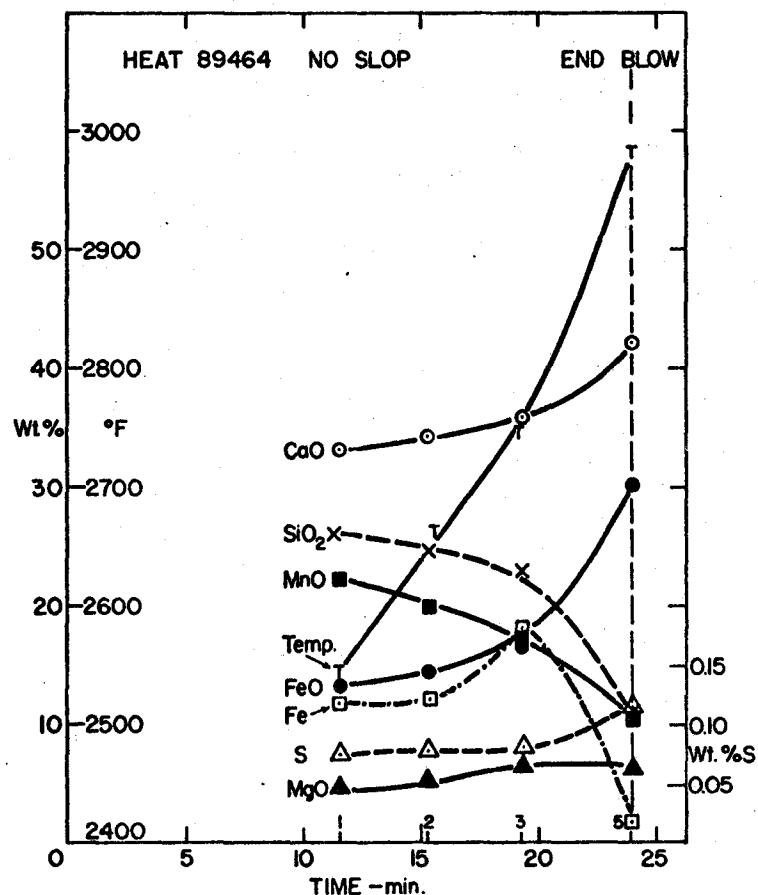


Figure 1.1(b) Slag Analyses During Blow

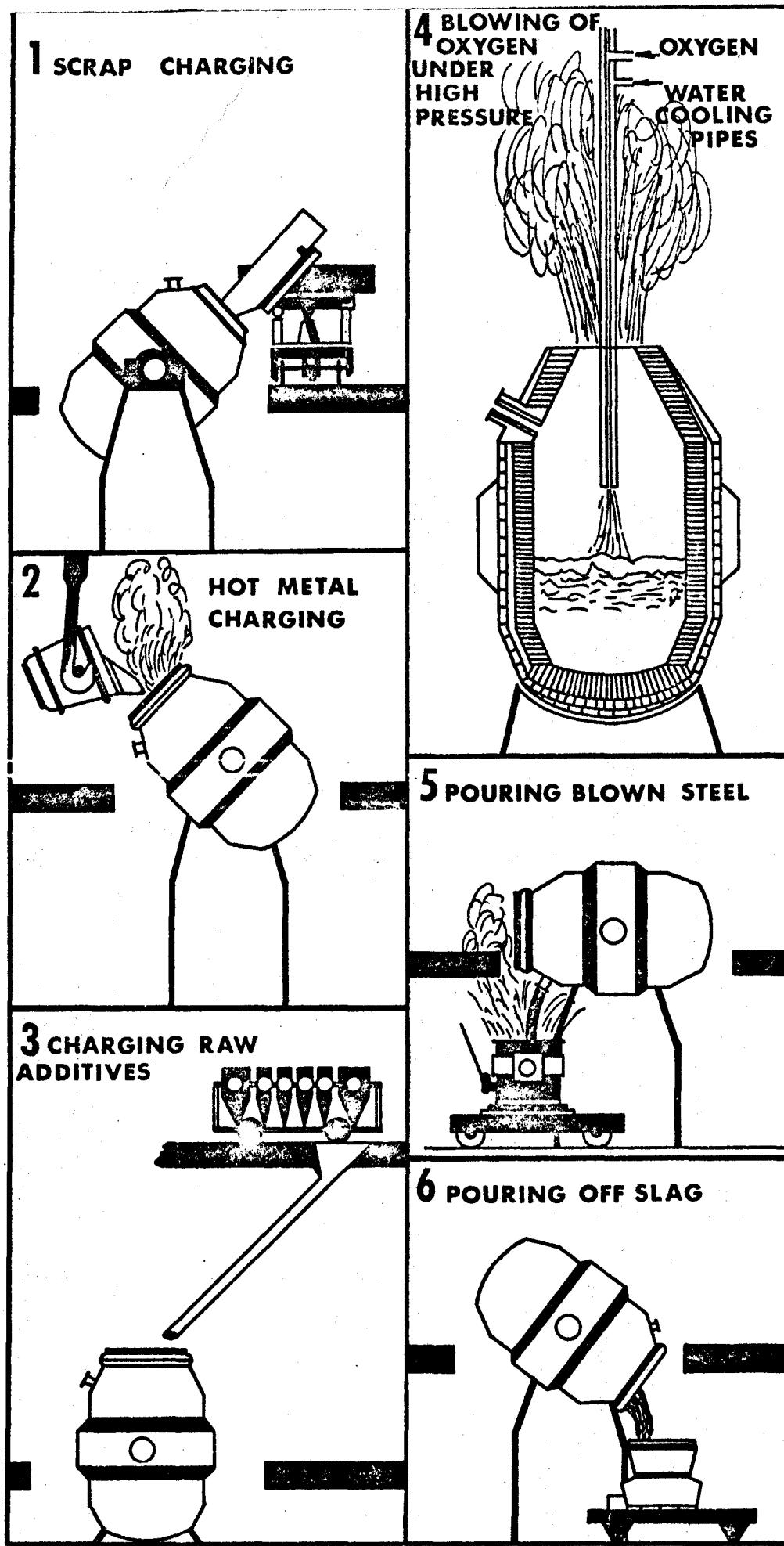


Figure 1.2 Sequence of Operations in a B.O.F.

vertically into the furnace. Effluent gases, collected by a furnace hood, pass into a wet gas cooling and scrubbing system. Fluxes such as limestone and dolomite are added in specified quantities to form a slag of desired basicity and fluidity. The basic slag, essential to sulphur and phosphorous removal, also collects oxidation products.

Shortly before turndown bath temperatures are taken using bomb thermocouples, the blow being terminated when the correct end point temperature is reached. After turndown another temperature measurement, together with a bath sample analysis, is taken to ensure that specifications have been met.

CHAPTER 2

B.O.F. CONTROL SYSTEMS

2.1 Advantages

An important variable in B.O.F. steelmaking is the end point carbon content. If this carbon content is above specification, approximately 10% of tap-to-tap time may be used in a reblow to remove more carbon. The problems of overblowing are illustrated by Table 2.1⁽⁴⁾ and Figure 2.1. Lower carbon levels correspond with sharply increasing oxygen saturation, i.e., overblowing requires extra deoxidation which is costly in materials and produces a greater number of undesirable oxide inclusions in the steel. Table 2.1 shows that as lower carbon contents are reached an increasing amount of oxygen combines with iron to form unrecoverable FeO in the slag, thereby reducing the iron yield. Overblown steels must also be recarburized in order to meet specifications.

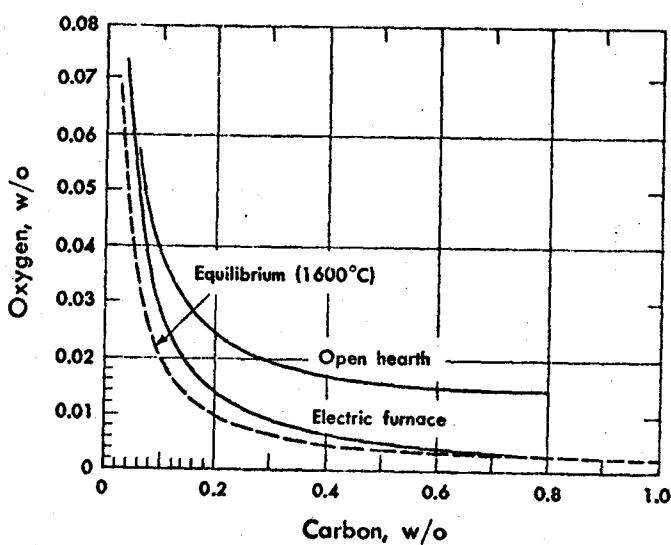
Another production problem is slopping -- the physical ejection of slag-metal-gas emulsion from the B.O.F. It occurs when oxygen from slag FeO supplements the lance oxygen for carbon elimination giving an accelerated, sometimes violent evolution of CO. Slopping further reduces the iron yield but more important, it indicates the refining process is out of control. Figures 2.2(a) and 2.2(b)⁽³⁾ contrast slag analyses of a no-slop and slop heat.

Loss of production time, excess oxygen consumption, oxide inclusion content and loss in iron yield are crucial factors to minimize if more efficient

TABLE 2.1

DISTRIBUTION OF TOTAL OXYGEN

wt.% C	$O_2 \rightarrow CO$	$O_2 \rightarrow FeO$	$O_2 \rightarrow [O]$
2.00	95	5	---
1.00	95	5	---
0.2	65	30	5
0.06	18	70	12

Figure 2.1 Bath Oxygen Content $[O]$ as a Function of Bath Carbon Content $[C]$

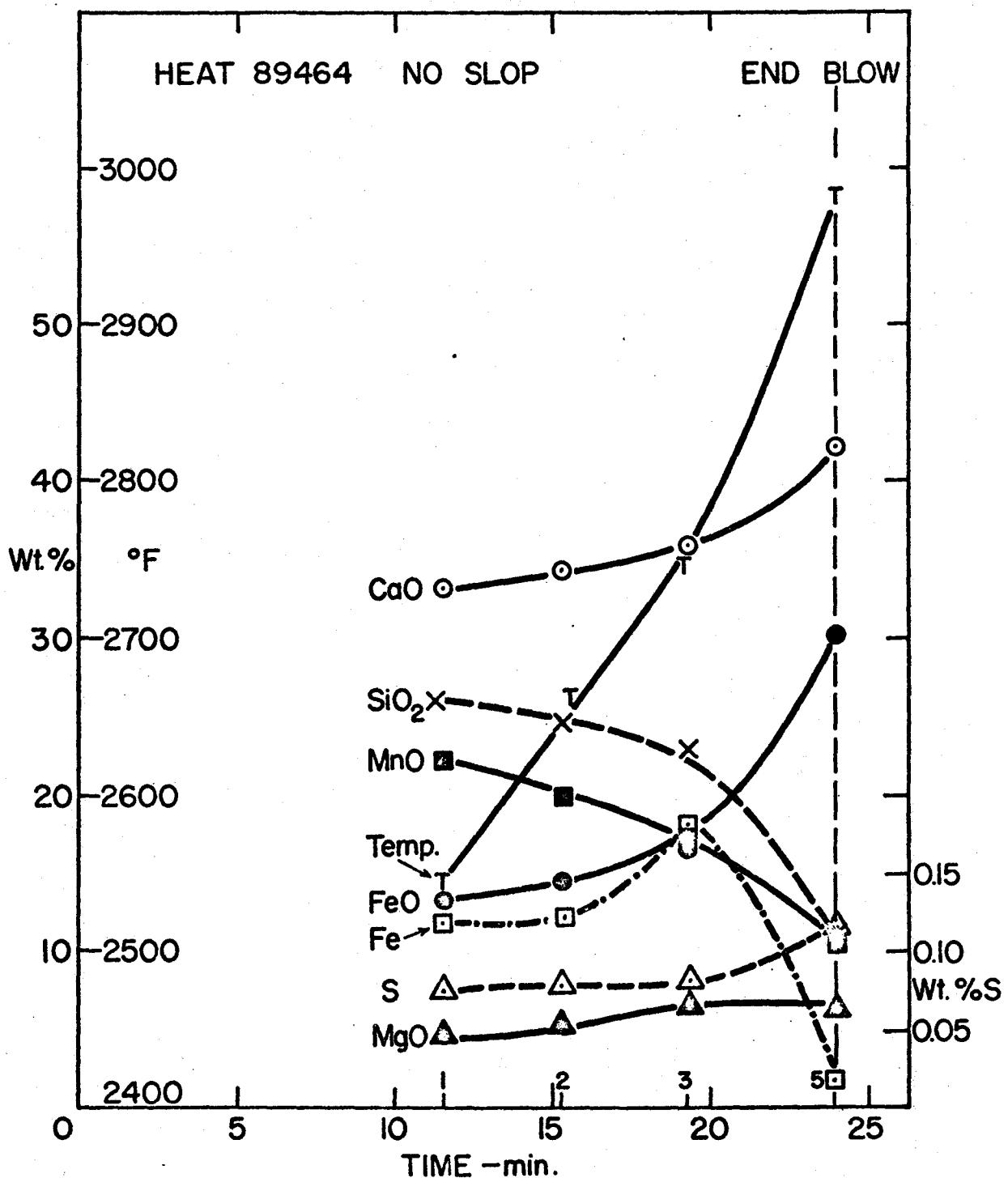


Figure 2.2(a) Slag Analysis During a No-Slop Heat

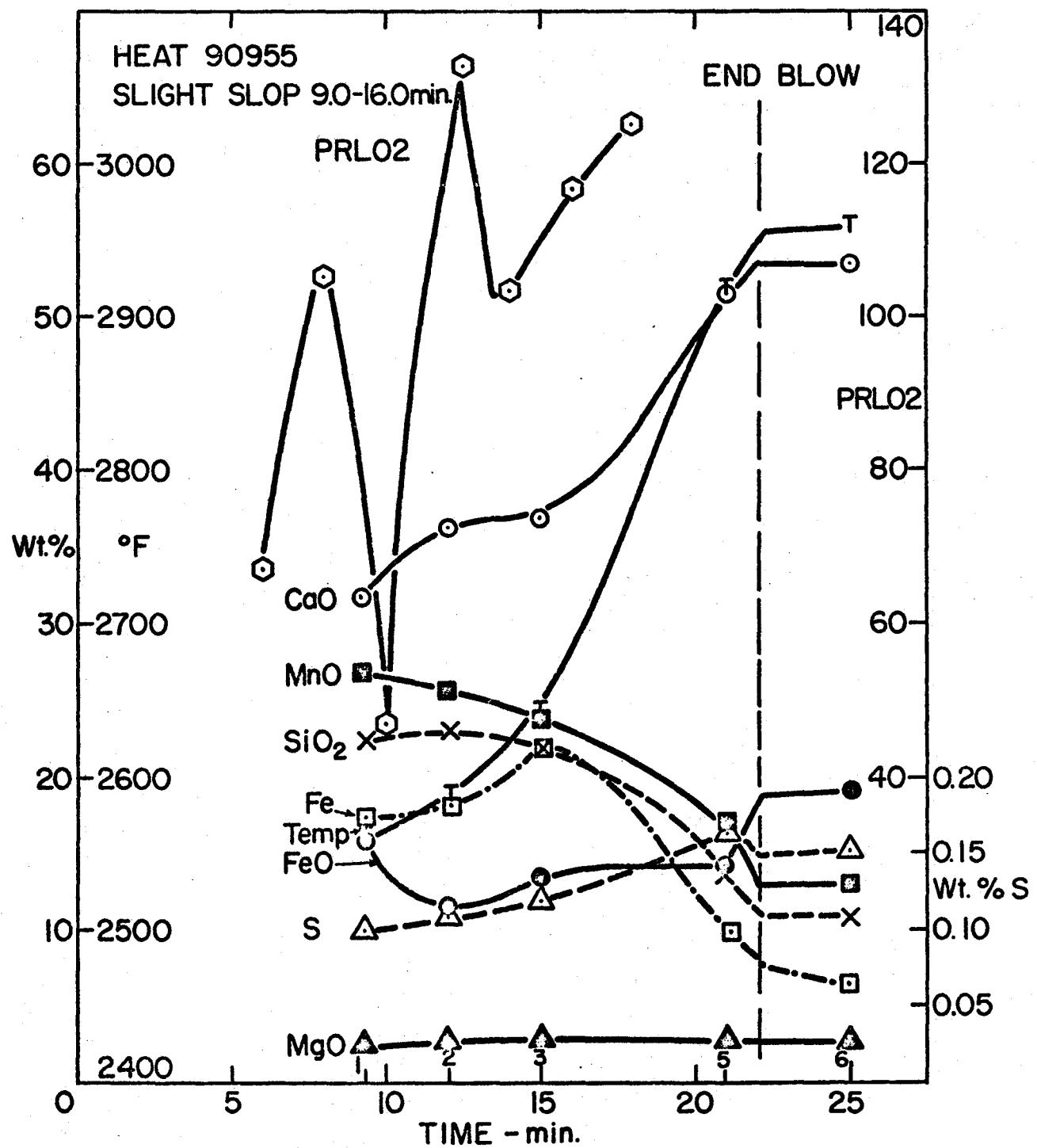


Figure 2.2(b) Slag Analysis During a Slop Heat

profitable steelmaking is to be realized.

2.2 General Principles

Methods of control can be grouped into two general categories: static and dynamic. A static approach uses an empirically based heat and mass balance made before the blow begins, to calculate the correct amount of input materials required to arrive at the desired end point specifications. Although this procedure is standard practice at most North American steel plants, it is unsatisfactory because it cannot compensate for refining irregularities (e.g., slopping) and cannot predict at what time specifications will be met.

In contrast, dynamic control takes into account such irregularities by on-line changes in operating variables (lance height and oxygen flow) as well as end point carbon and temperature predictions. The most fruitful dynamic end point carbon control system incorporates indirect carbon measurement by means of mass balances based on effluent gas analysis. Since carbon is evolved from a B.O.F. in gaseous form as CO and CO₂, the indirect carbon calculations are based on knowledge of (1) a reference carbon point, (2) bath weight, (3) stack gas flow measurement, and (4) stack gas analyses for CO and CO₂.

Hot metal carbon analysis is not a satisfactory reference point because the analytical error involved is greater than the allowable end point tolerances. In addition, error introduced by integration of the decarburization curve throughout the blow only compounds the problem. Approximate studies^(5,6) made no direct continuous measurements of stack gas flow, either assuming the flow was constant or proportional to the rated power of the exhaust fan system. More precise studies lead to the use of stack pressure differentials for flow measurement, e.g., an

orifice plate⁽⁷⁾. These methods share a common difficulty. Sample gas is analyzed on a dry, clean basis but flow data apply to actual stack conditions. Calculation of the CO and CO₂ mass flows therefore require a knowledge of water vapour and particulate matter contents -- added complications and sources of error.

For these reasons the optimism associated with B.O.F. control systems based on such flow measurements has proved to be unwarranted, the application in practice being singularly⁽⁴⁾ unsuccessful.

2.3 Control by a Chemical Tracer Technique

2.3.1 Stack Gas Flow

A new chemical tracer technique by-passes the problems associated with monitoring stack gas water vapour and particulate matter content, pressure and temperature.

Methane (as natural gas) is injected at a known rate into the stack. A stack gas sample is taken downstream from the injection point and analyzed. Since this analysis is on a clean, dry basis and the CH₄ injection rate may be expressed as scfm, stack gas flow rate data are also calculated on a clean, dry basis in scfm.

2.3.2 Chemical Analysis

As an alternative to hot metal carbon analysis, in-blow sampling and analysis for carbon has been used in this study to reduce analytical error. Also the error introduced by integrating the decarburization curve is reduced, since integration now takes place over a fraction of the total blowing time.

CHAPTER 3

CONTROL EQUATIONS

Detailed derivations and error propagation formulae for each equation are given in Appendix I. Variable names are identical with computer listings in Appendix II; [] after the equation number contains its location in the program.

3.1 Carbon Evolution Rate

Stack gas analyses for CO_2 , CO, and CH_4 combined with methane injection rates permit calculation of the carbon evolution rate.

$$\text{SFLOW}(t) = \frac{\text{FLWCH4}}{\text{CH4}} \times 100 \quad 3.1 [2. 567]$$

$$\text{CEV}(t) = \frac{0.031(\text{CO}_2 + \text{CO}) \times \text{SFLOW}(t)}{100} \quad 3.2 [2. 612]$$

where at time t (minutes):

SFLOW(t) = stack gas flow (scfm)

FLWCH4 = methane injection rate (scfm)

CH4 = stack gas methane content (%)

CEV(t) = carbon evolution rate (lb/min)

CO₂ = stack gas carbon dioxide content (%)

CO = stack gas carbon monoxide content (%)

3.2 Bath Carbon Content

The bath carbon content (%) at time t can be expressed as:

$$\text{BCC } (t) = \frac{100((\text{DATMPT} \times \text{TOTWT})/100 - \text{CEVD})}{\text{TOTWT}}$$

3.3 [2. 698]

where DATMPT = carbon datum point analysis (%)

TOTWT = estimated bath weight (assumed constant from sampling time until turndown - 1b)

CEVD = carbon evolved from time of datum point (1b)

3.3 Modelling

Updating a least squares fit of the polynomial

$$\text{BCC } (t) = a + bt + ct^2$$

3.4

proves to be an extremely successful way of relating bath carbon content with time. Rearranging 3.4 gives

$$\text{TPRED3} = \frac{-b - \sqrt{b^2 - 4c(a - \text{STALC})}}{2c}$$

3.5

where STALC = end point carbon analysis (%)

TPRED3 = predicted end point time (minutes)

The predicted time left (seconds) at time t is

$$\text{TLEFT3} = (\text{TPRED3} - t) 60 - \text{MAXLAG}$$

3.6 [2.1115]

where MAXLAG = maximum time delay of control system (seconds)

3.4 Stack Lance Oxygen

PRL02 (percentage recovered lance oxygen) is a variable used to estimate the proportion of input lance oxygen used in decarburization. It is defined as the percentage of input lance oxygen recovered in the stack. Preliminary work⁽³⁾ has shown PRL02 to be an effective indicator of the degree of slopping.

$$\text{PRL02 } (\text{t}) = \frac{[1.27 (\text{CO}_2 + \text{O}_2) + 0.77 \text{ CO} - 26.50] \times \text{SFLOW } (\text{t})}{\text{FLW02 } (\text{t})/100} \cdot 3.7 [2.592]$$

where O₂ = stack gas oxygen content (%)

FLW02(t) = input lance oxygen flow (scfm)

CHAPTER 4

EXPERIMENTAL

4.1 Tracer Injection

Methane, supplied from bottles of compressed natural gas, was injected at constant pressure through a flowmeter and injection probe into the stack gas (Figure 4.1). The probe was designed to ensure that the methane would be well-mixed in the turbulent flow -- the basic assumption in this tracer technique. Pressure, temperature, and rotameter readings were recorded in order to correct injection flow rates to standard conditions. Stack gas temperature readings (under 200°F) indicated no possibility of CH₄ combustion.

4.2 Gas Sampling and Preparation

Approximately 100 feet downstream from the injection point and around a 120° pipe elbow, stack gas was continuously sampled through a probe identical to that used for methane injection. 25 cu. ft/hr of saturated stack gas sample was drawn under reduced pressure through a gas-solid separator, for removal of gross particulate matter, to a sample preparation unit (Figure 4.2). In this unit, a heated filter (~ 300°F) removed most of the remaining particulate matter; subsequent passage through a refrigerated dryer reduced the water vapour content to approximately 5000 ppm (35°F dewpoint). Treatment by two more filters (10 and 3 µm) and CaSO₄ dessicant served as final preparation

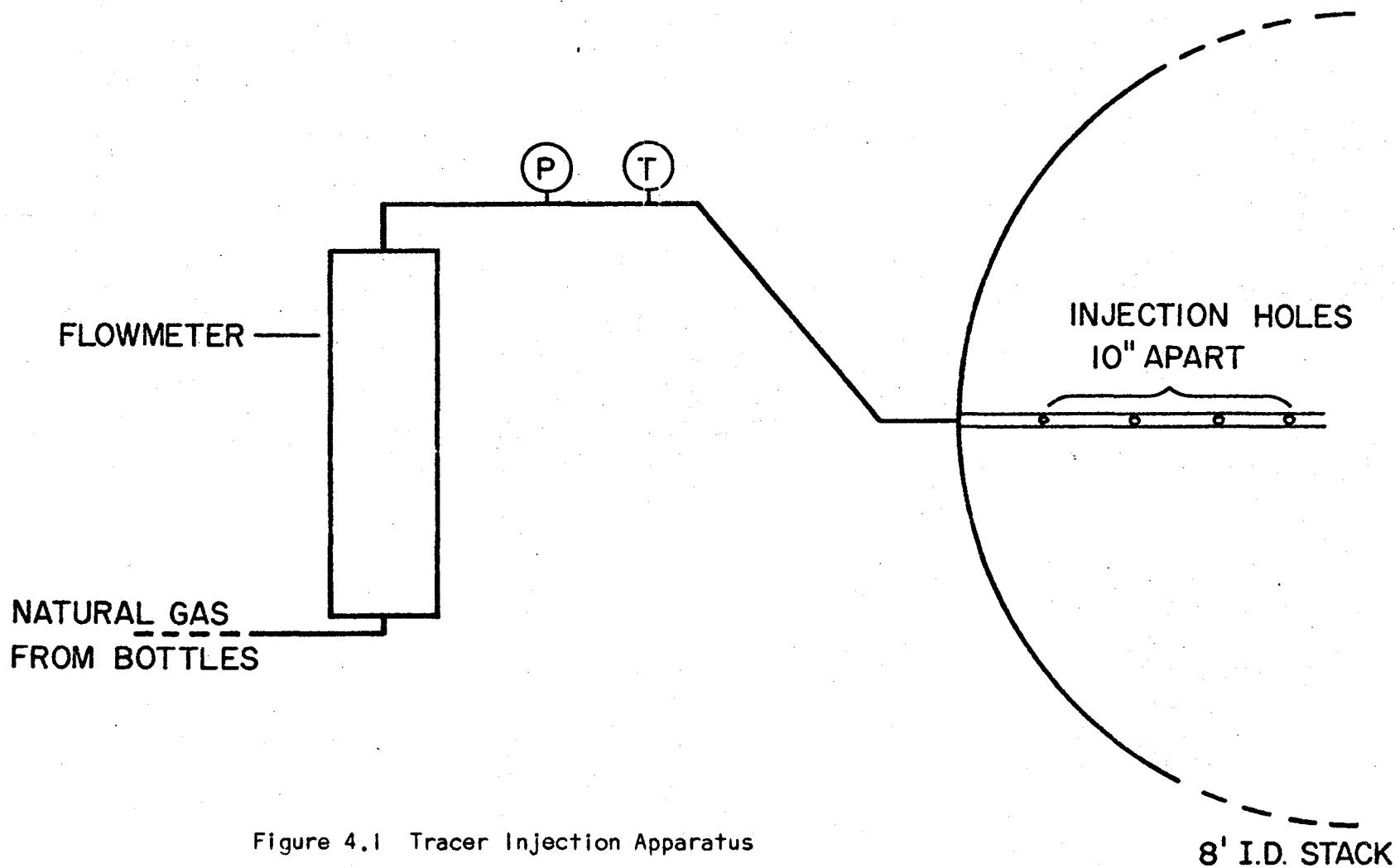


Figure 4.1 Tracer Injection Apparatus

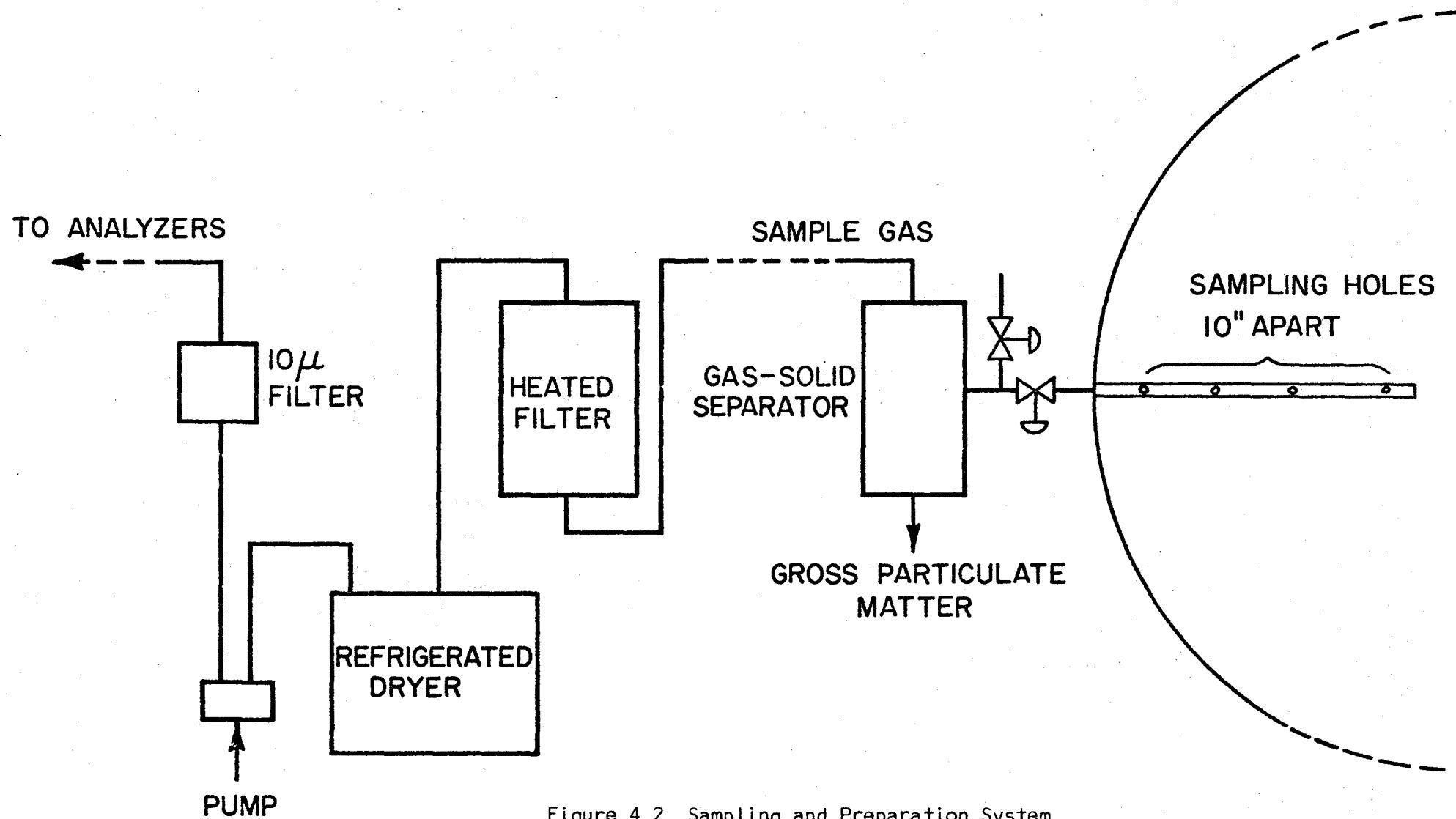


Figure 4.2 Sampling and Preparation System

before the sample passed, under pressure (5 psig), to the gas analysis unit.

4.3 Instrumentation

4.3.1 Gas Analysis

Three MSA infrared analyzers (for 0-30 % CO₂, 0-20% CO, 0-500 ppm CH₄) connected in series and a Beckman polarographic oxygen analyzer (0-25%) in parallel with the first three (Figure 4.3) provided continuous sample gas analysis. Two flowmeters monitored the parallel flows of 4 cu. ft/hr, excess sample being vented to atmosphere. Each analyzer was calibrated before every experimental run with zero and span gases.

4.3.2 Operating Variables

The lance height above the static metal bath and lance oxygen flow were also recorded in order to examine their influence on PRL02 and decarburization.

4.3.3 Time Delay

Transport time from the sample point to the entrance of the sample preparation unit was calculated from known volumetric sample flows and the diameter of the sample line. The remaining time delays for CO, CO₂ and CH₄ were calculated from 90% response curves of standards injected at the entrance of the preparation and analyzer units. The O₂ response time was estimated by correlations with processed gas analysis data.

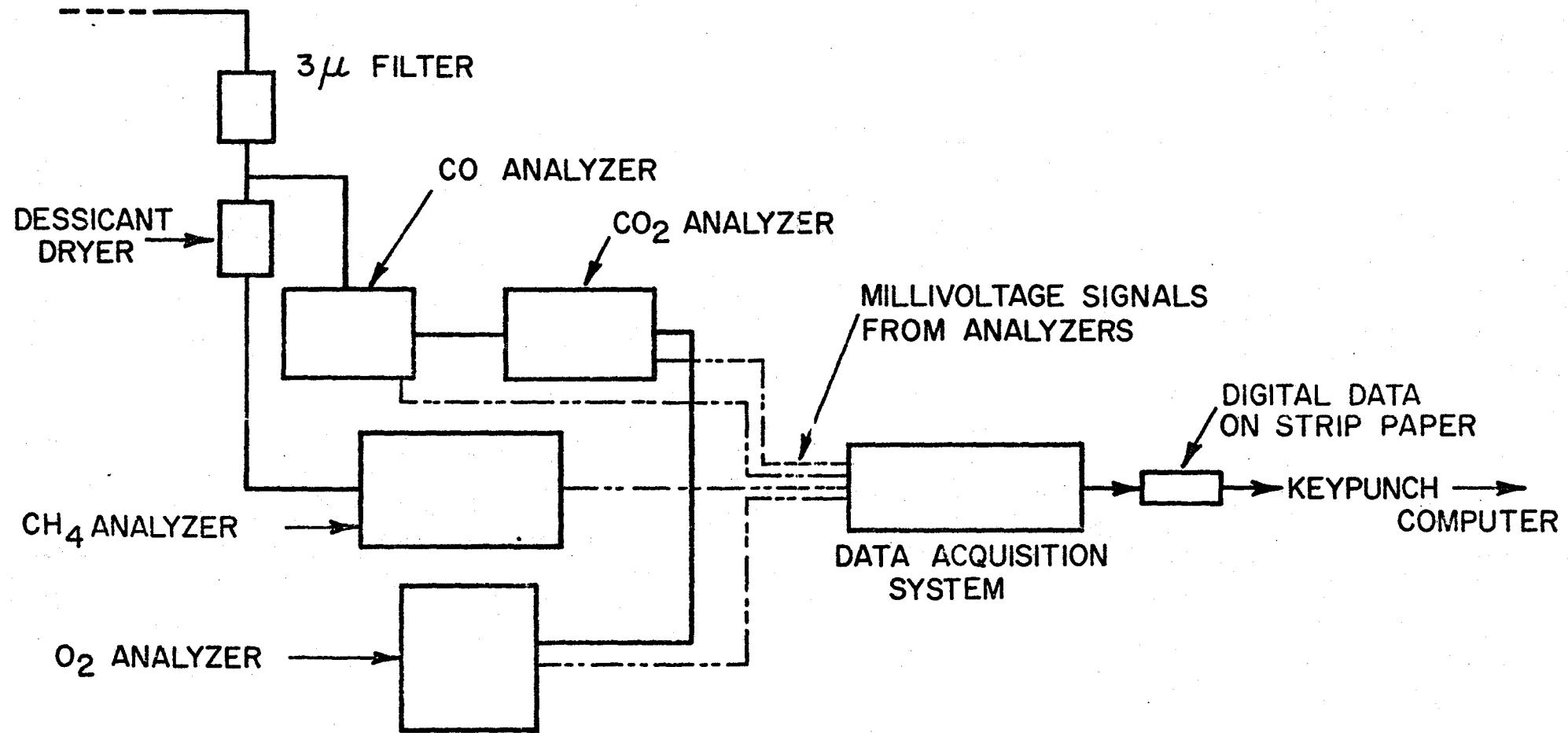


Figure 4.3 Gas Analysis and Data Collection

TABLE 4.1

TIME DELAYS

Gas	Time (seconds)
CO	52
CO ₂	51
CH ₄	54
O ₂	59

4.4 Data Collection and Processing

An Esterline Angus D-2020 data acquisition system monitored and printed the six channels of millivoltage data in digital forms on a strip paper approximately every 3 seconds. After the information was keypunched onto data cards, it was processed by a program run on the CDC 6400 computer at McMaster University. Appendix II provides a complete listing of the program.

CHAPTER 5

RESULTS

5.1 Presentation and Discussion

5.1.1 Measured Variables

(a) Gas Analysis

Figure 5.1 illustrates the CO, CO₂ and O₂ analyses throughout HEAT #19923. A sharp decrease in CO₂ content and increase in O₂ content at the 20.2 minute mark show the effect of temporarily halting the blow to obtain a carbon datum point; the re-blow at the 24.3 minute mark for 1.3 minutes is also clearly shown by a CO₂ peak and an O₂ drop.

The marked shifts of component analyses from the 14 - 20 minute mark are characteristic of double furnace operation⁽³⁾. When the waste gas system is set for such operation, the total mass flow must now be divided between each furnace off-gas duct which join a common header feeding into a 3-fan venturi scrubber system. With a lower mass flow on the furnace being studied, less air is drawn through the open hood to dilute the CO and CO₂ contents; consequently these analyses sharply increase and the O₂ content decreases.

Closer scrutiny near the 14 minute mark reveals smaller peaks on all 3 curves followed by relatively stable analyses until the sampling turndown (20 minute mark). These peaks indicate a brief period of abnormally high carbon evolution which suggests that some degree of slopping has occurred here; this

GAS ANALYSES -- HEAT NO. 19923

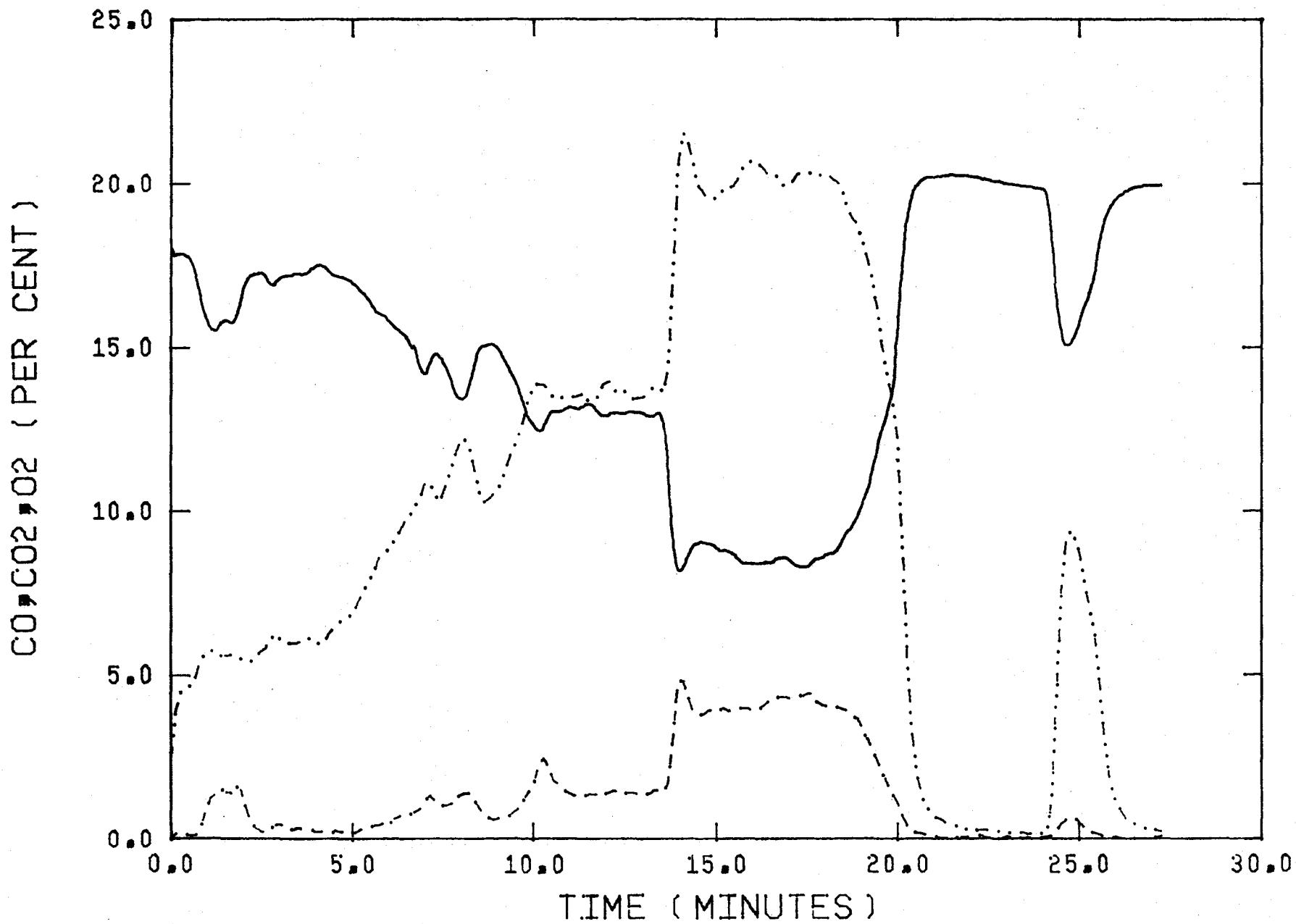


Figure 5.1 CO , CO_2 and O_2 Stack Gas Analyses

becomes more clear in the CEV and PRL02 curves to be discussed shortly.

(b) Lance Height and Oxygen Flow

Figures 5.2 and 5.3 show these two operating variables during the blow and indicate more accurately when the turndowns and reblow occurred.

(c) Methane Injection Temperature

The unusual temperature curve (Figure 5.4) was typical of all the heats studied and is explained as follows: the adiabatic cooling effects of the sudden large pressure drops (> 2000 psia at first) through the gas bottle regulators caused adiabatic cooling. As more gas was injected, the pressure differential (and adiabatic cooling effects) decreased, thereby allowing heat transfer from the surrounding atmosphere to predominate until ambient temperatures were reached once again.

5.1.2 Calculated Variables

(a) Stack Gas Flow

Figure 5.5 shows quite clearly how sharply the stack flow decreased when a second furnace duct was opened to the fan system at the 14 minute mark. The constant flow shortly before and after the reblow is a result of temporarily shutting off the tracer injection to conserve a low natural gas supply; to avoid calculating an infinitely large flow (with no tracer present) and a consequently distorted CEV curve, the flow was set at a suitable level. This

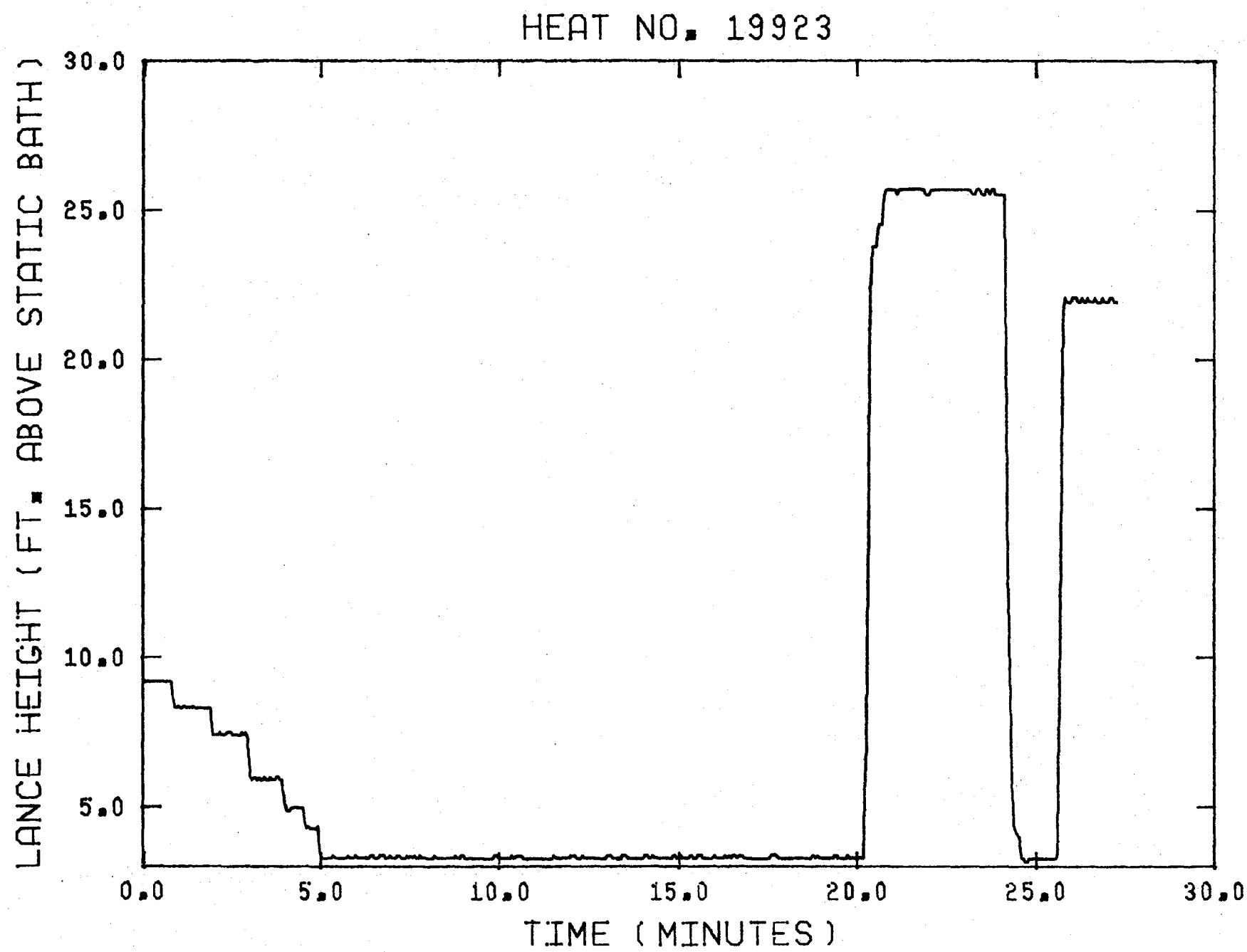


Figure 5.2 Lance Height Above Static Bath During a Blow

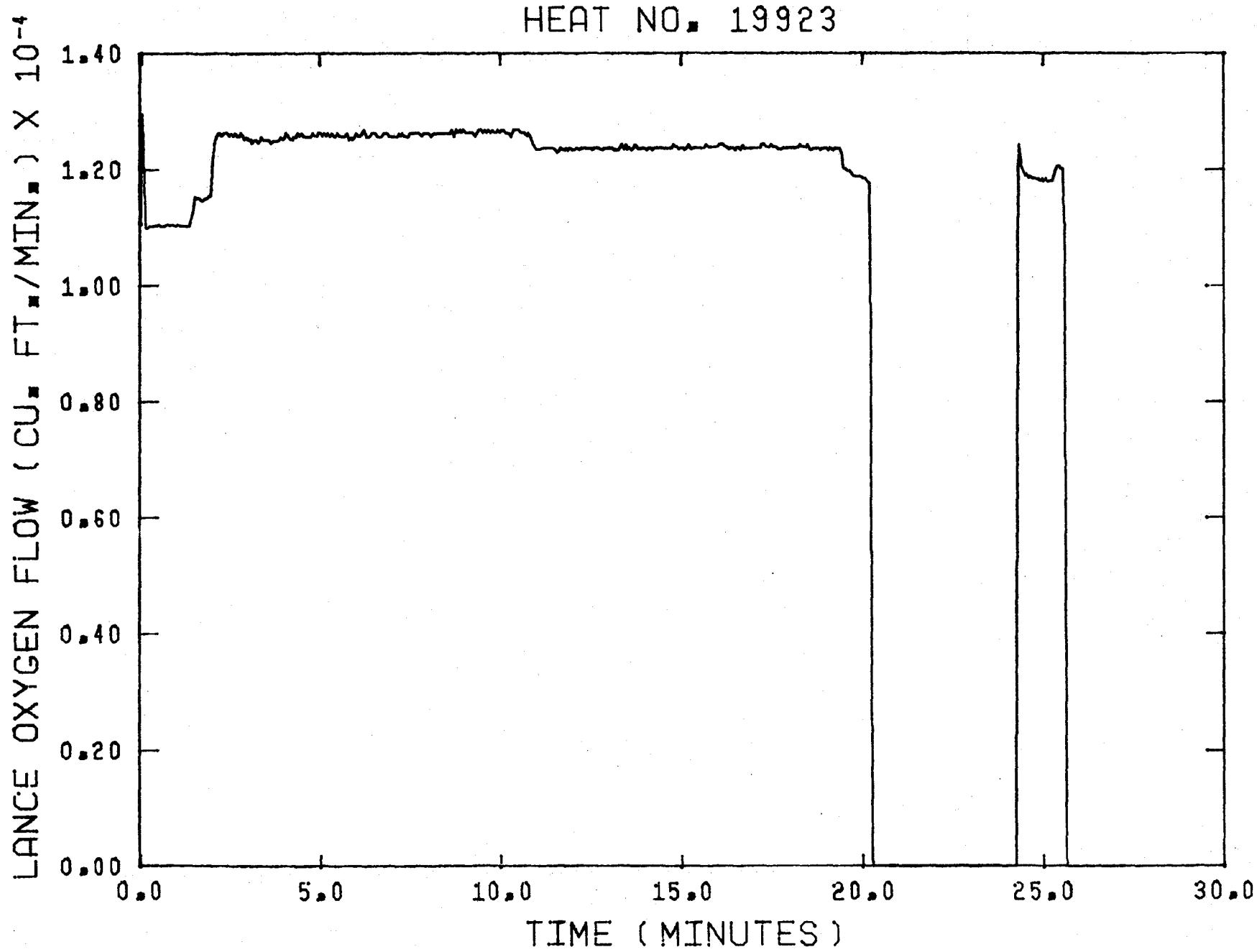


Figure 5.3 Lance Oxygen Flow

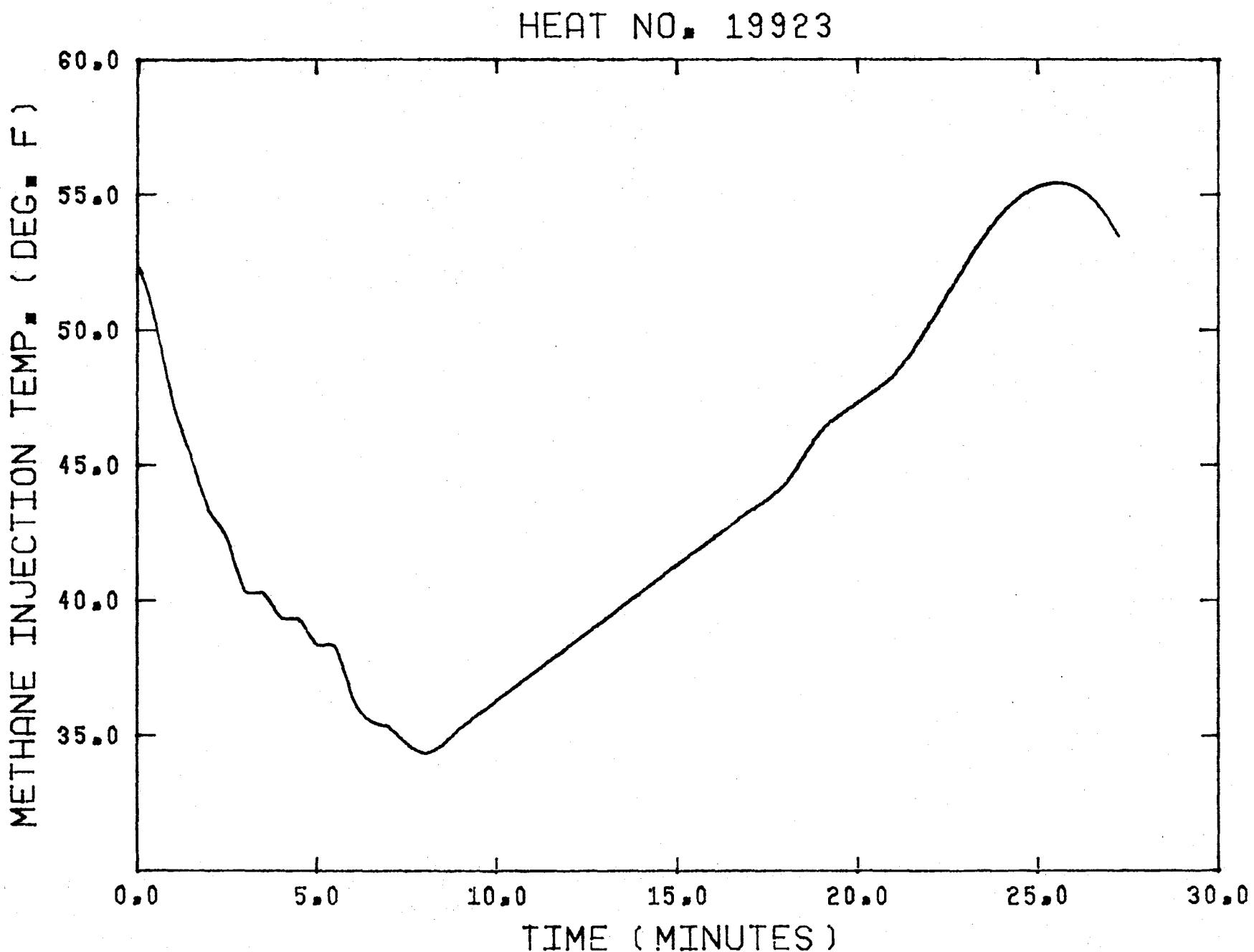


Figure 5.4 Methane Injection
Temperature

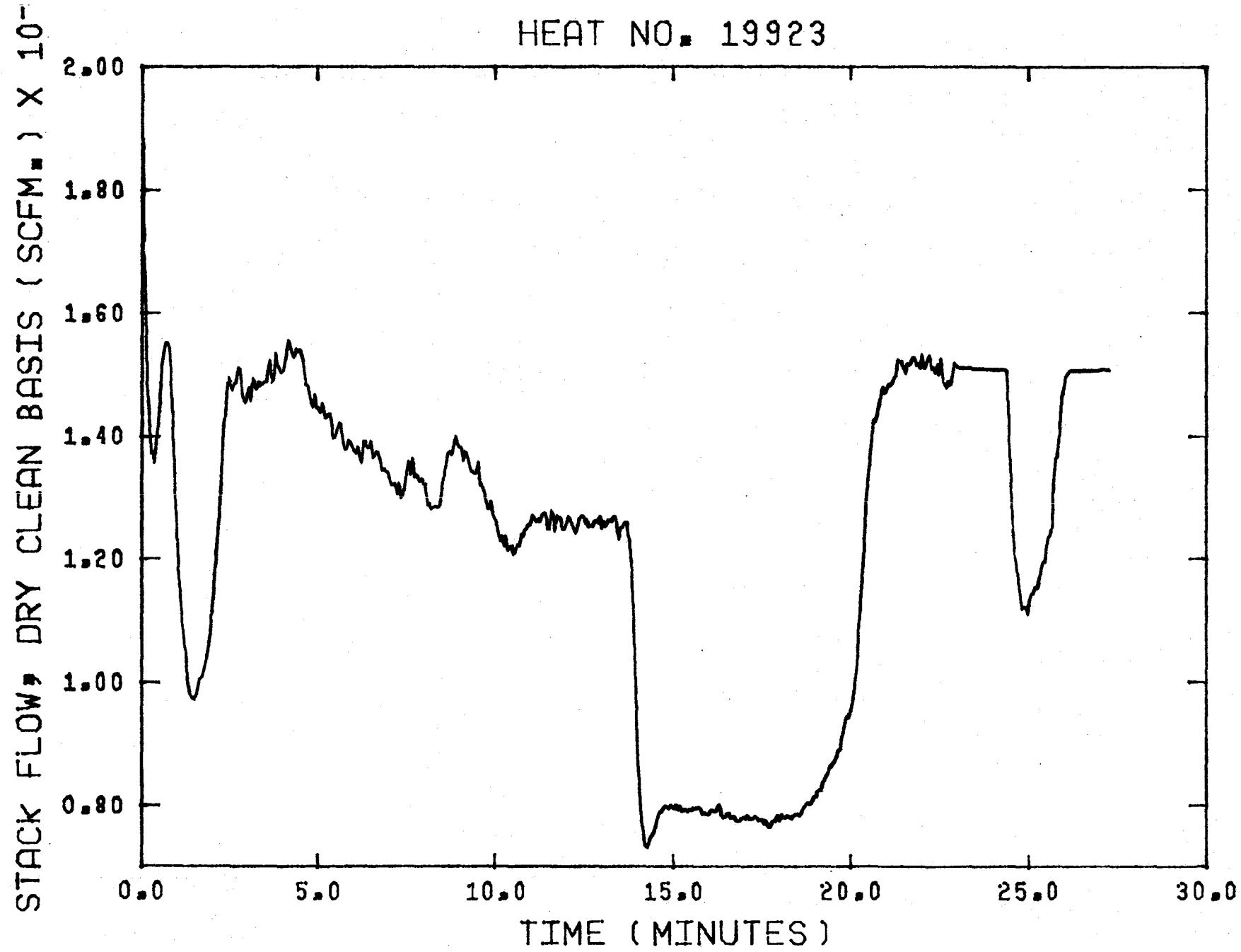


Figure 5.5 Stack Gas Flow

only affects to a minor extent the carbon evolution rate when the furnace is turned down. Stack flow errors throughout the blow averaged < 3.5 %, a sample calculation being given in Appendix I, equations 1.11 and 1.12.

(b) Carbon Evolution Rate

The decarburization curve (Figure 5.6) is calculated from the data shown in Figures 5.1 and 5.5 using equation 3.2. The sharp peak at the 14 minute mark indicates that mild slopping occurred over a total period of less than 20 - 30 seconds.

It is clearly demonstrated by Figure 5.7 that decarburization still occurs after the lance oxygen is shut off. The amount of carbon evolved from turndown to sampling depends on the carbon level at turndown and the amount of time taken from turndown to sampling. In this particular case ~ 2 points of carbon were evolved after each turndown.

(c) Bath Carbon Content

The bath carbon content for HEAT # 19923 from the datum point to end point sampling times, shown by Figure 5.8, dropped from 0.245 % (chemical analysis), to 0.095 % C (mass balance) whereas the actual end point was 0.12 % C (chemical analysis). The dashed-dotted lines indicate approximate confidence intervals - they gradually diverge as error accumulates.

Since sampling times were not recorded they had to be assumed. The datum point time was taken halfway between the turndown (oxygen off) and reblow (oxygen on). Final sampling times were taken as 1.6 minutes after the final turndown, the statistical average⁽⁸⁾.

HEAT NO. 19923

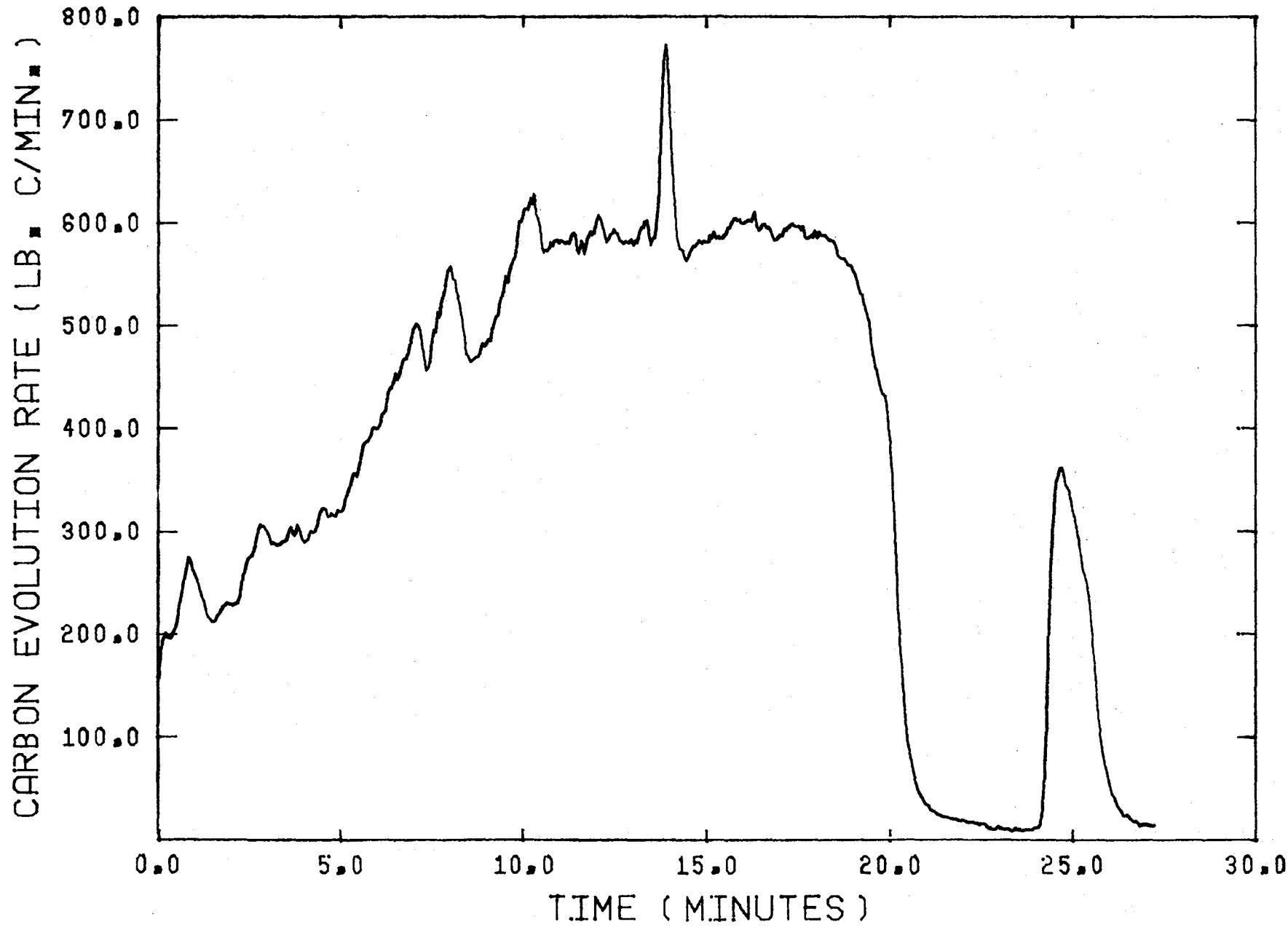


Figure 5.6 Carbon Evolution Rate

HEAT NO. 19923 -- DATUM PT. TO END PT. SAMPLE TIME

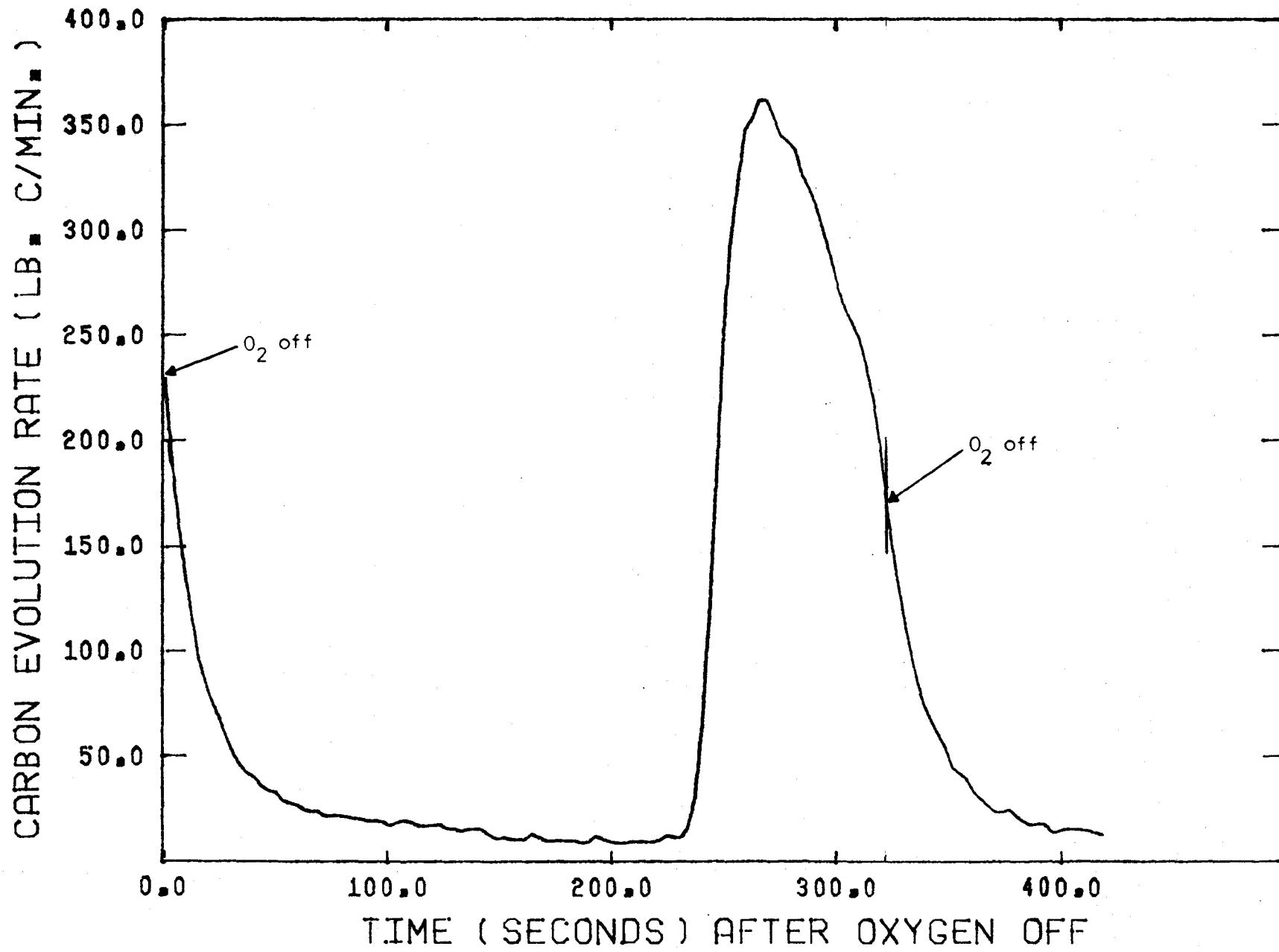


Figure 5.7 Carbon Evolution Rate After Lance O₂ Shut Off

HEAT NO. 19923 -- DATUM PT. TO END PT. SAMPLE TIME

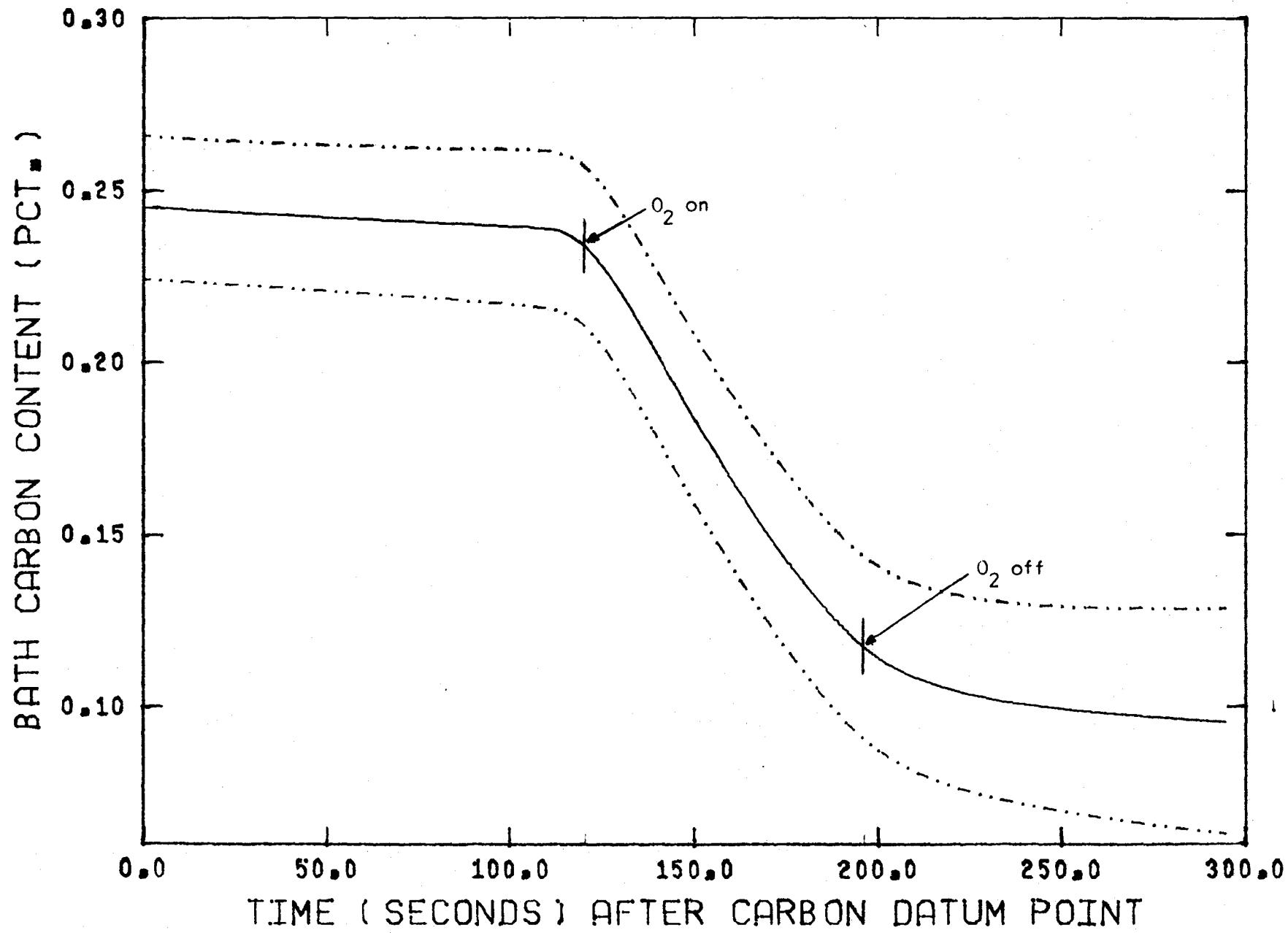


Figure 5.8 Bath Carbon Content (after datum point)

(d) PRL02

The percentage recovered lance oxygen (the percentage of input lance oxygen recovered as CO and CO₂ in the stack) illustrated by Figure 5.9 indicates the lance oxygen efficiency for carbon removal. Negative values result from inaccurate time delays and O₂ analyses. The PRL02 curve is similar to the decarburization curve (Figure 5.6) because both are calculated, partly or in whole, from the stack gas flow, CO and CO₂ contents.

The particularly high PRL02 values around the 14 minute mark can be explained by the fact that oxygen from slag FeO supplements lance O₂ to give a brief period of enhanced carbon evolution. This substantiates the findings of previous work⁽³⁾ which indicates that slopping becomes noticeable when PRL02 values exceed 70 - 80 %. Figures 5.10(a) and 5.10(b) show a better example of slopping and Figure 5.11 shows the operator's strategy to halt it.

Appendix I (1.4.2(b)) contains a sample PRL02 calculation; the average error of PRL02 values is ~ 13 %.

(e) UNUSDO2(t)

The amount of lance oxygen used for CO combustion within the furnace is calculated in Appendix I (1.4.1). Although variance analysis indicates that such calculated quantities are very inaccurate (average relative error of ~ 140 %) they may prove valuable in eliminating excess refractory wear in the corbel (or mouth) area of the B.O.F. Values of UNUSDO2(t) (and hence heat generated in CO-CO₂ combustion) averaged over several heats would be correlated with lance height and oxygen flow. The best settings of these two variables (i.e., to minimize UNUSDO2(t)) would be determined in order to reduce furnace lining costs. However, in order to use UNUSDO2(t) for control

HEAT NO. 19923

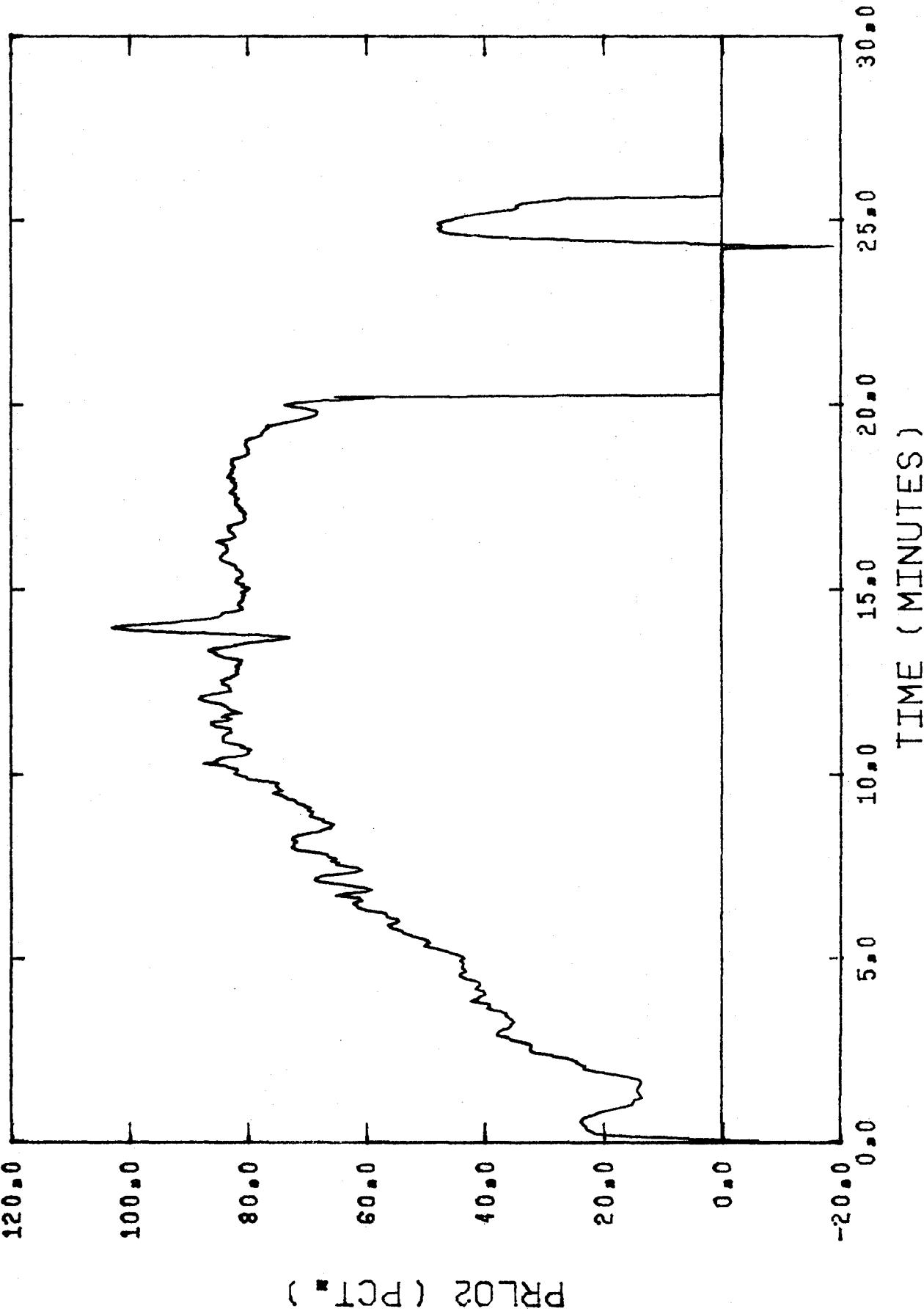


Figure 5.9 PRLO2

HEAT NO. 19926

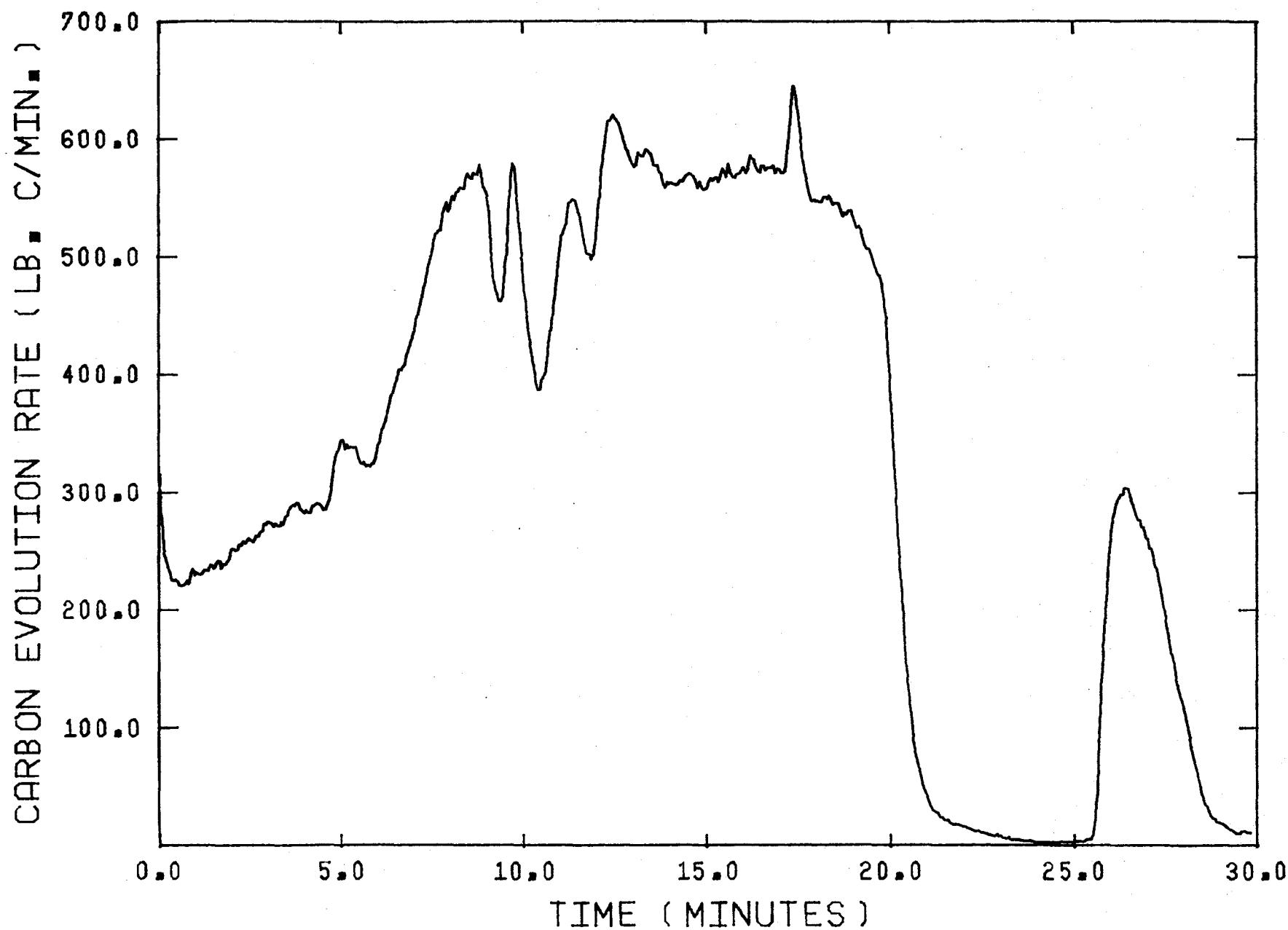


Figure 5.10(a) Carbon Evolution (Stopping Heat)

HEAT NO. 19926

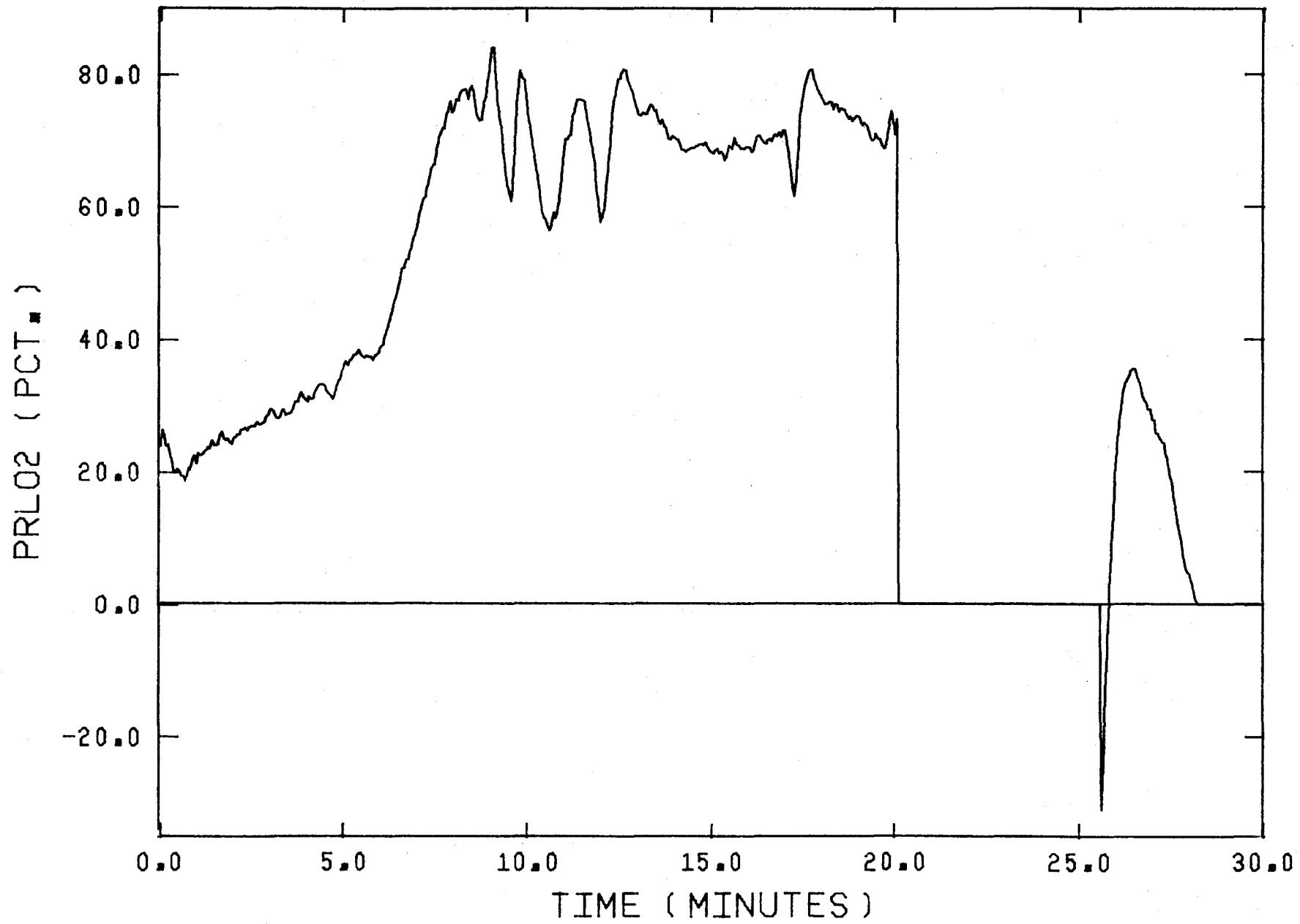


Figure 5.10(b) PRL02 (Stopping Heat)

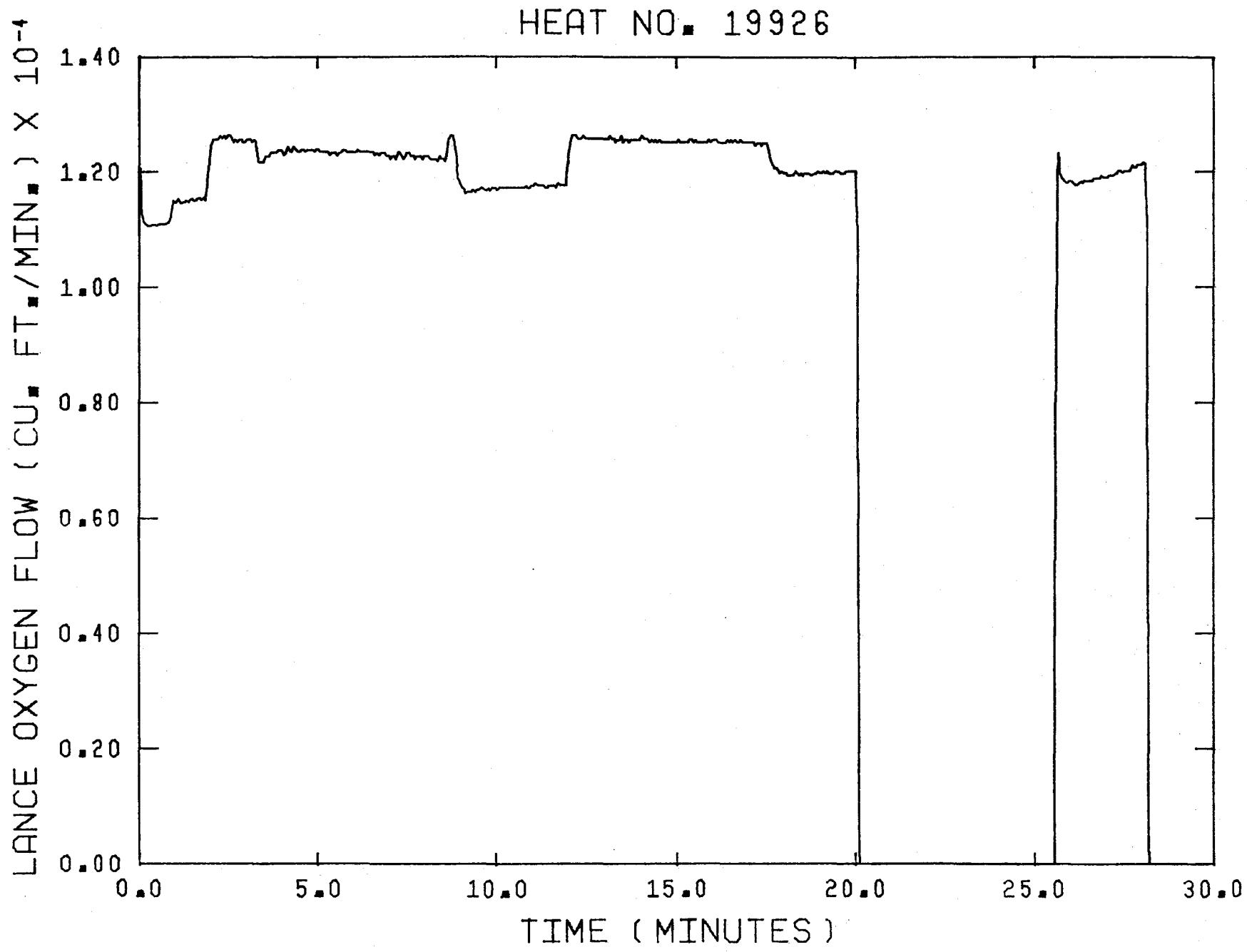


Figure 5.11 Lance O_2 Flow During Stopping

purposes it will have to be known much more accurately, i.e., error introduced in O_2 , CO_2 and CO (in decreasing order of importance) must be reduced by an order of magnitude, e.g., from 1 % to 0.1 % of full scale reading. Figure 5.12 provides an example of $UNUSDO_2(t)$ for HEAT # 19923 - the total amount unused for this heat was $\sim 7,000 \pm 10,000 \text{ ft}^3$ lance O_2 .

5.1.3 Mathematical Modelling

Unfortunately the reblow time for HEAT # 19923 was too short (with respect to delay time) to permit modelling and prediction before turndown. However, data from HEATS # 19926 and # 19929 indicate that updating a least squares fit of the form

$$BCC(t) = a + bt + ct^2 \quad (3.4)$$

(where $BCC(t)$ = bath carbon content (%) at time t minutes) predicts end point carbon contents (measured by gas analysis) within acceptable error limits ($\pm 0.01 \%$) 25 seconds before turndown. These predictions take into account the fact that 1 - 2 points of carbon can be removed from the time the oxygen supply is shut off until the bath is sampled. Figures 5.13(a) and 5.13(b) contrast predicted carbon content curves with the actual measured curves. Detailed prediction results are tabulated in Table 5.2. Because datum carbon analyses were unavailable for these two heats they were assigned values (Table 5.1) such that the calculated end point value would equal the actual analyzed value - thus Table 5.2 lists results calculated assuming no error in carbon evolution data. The predicted end points are lower than the measured end points (mass balance) but the fact that approximately 1- 2 points of a carbon

HEAT NO. 19923

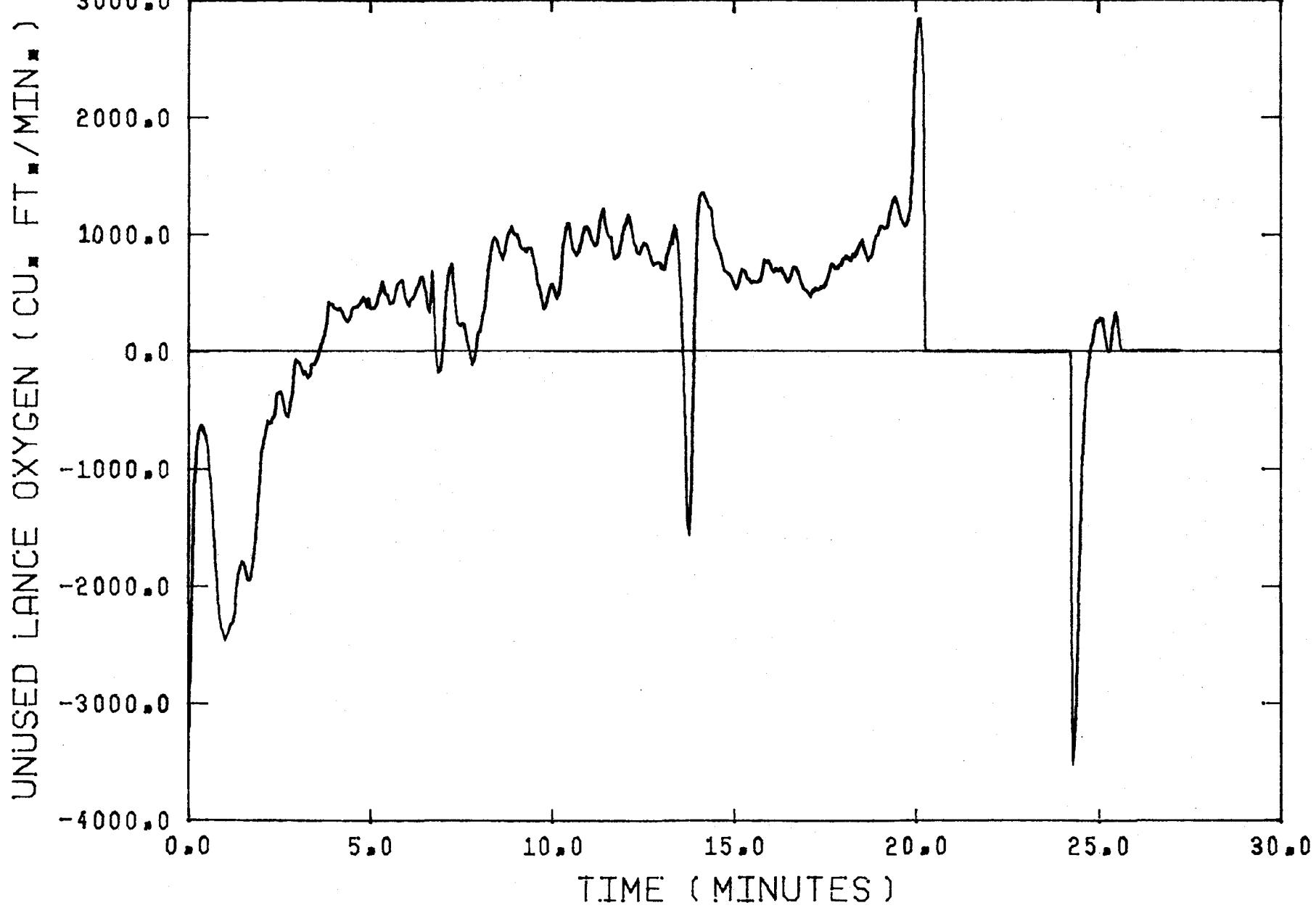


Figure 5.12 Lance O_2 Used for CO Combustion in B.O.F.
2

HEAT NO. 19926 -- END POINT PREDICTIONS

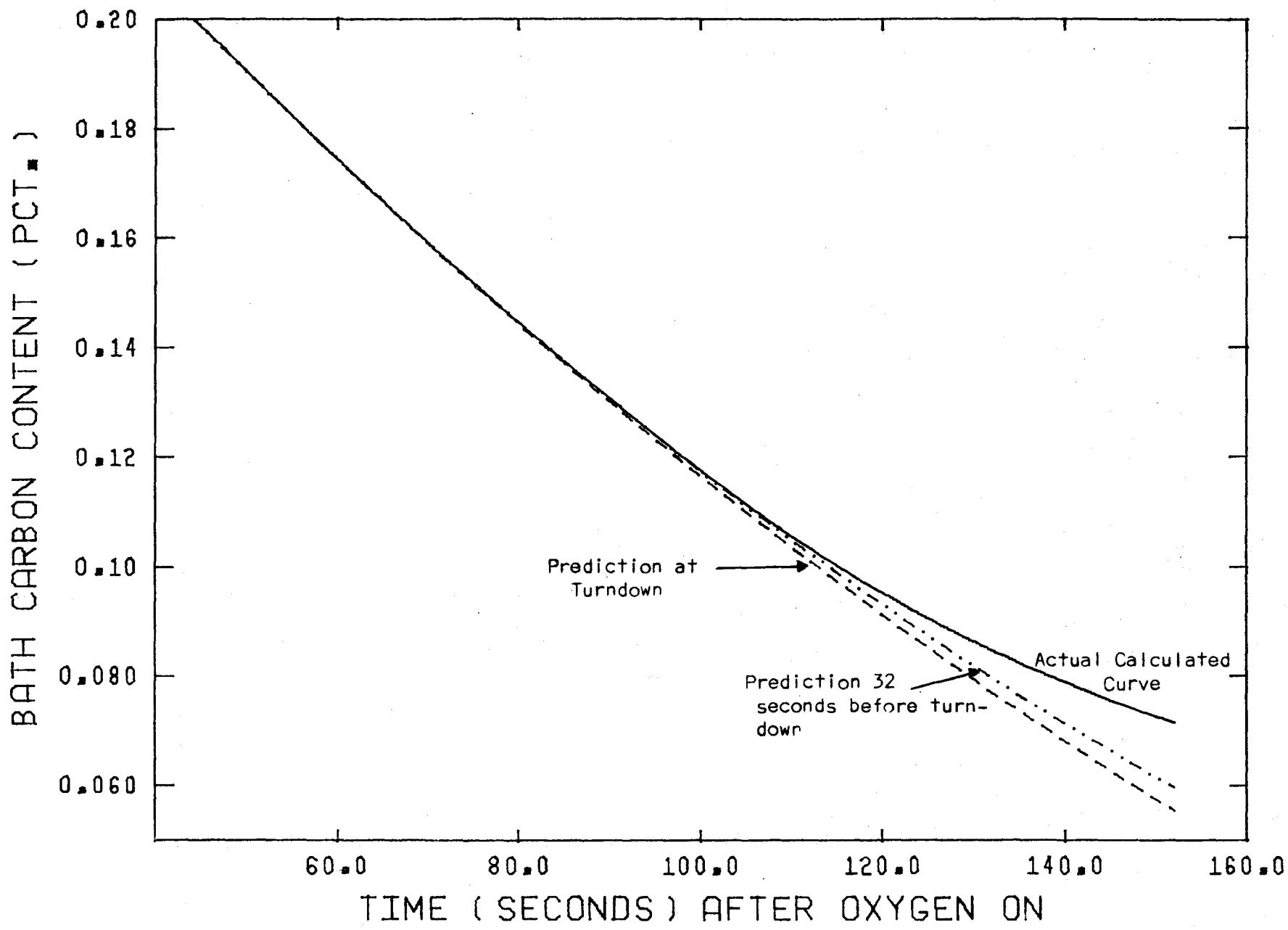


Figure 5.13(a) Predicted Bath Carbon Curves HEAT # 19926

HEAT NO. 19929 -- END POINT PREDICTIONS

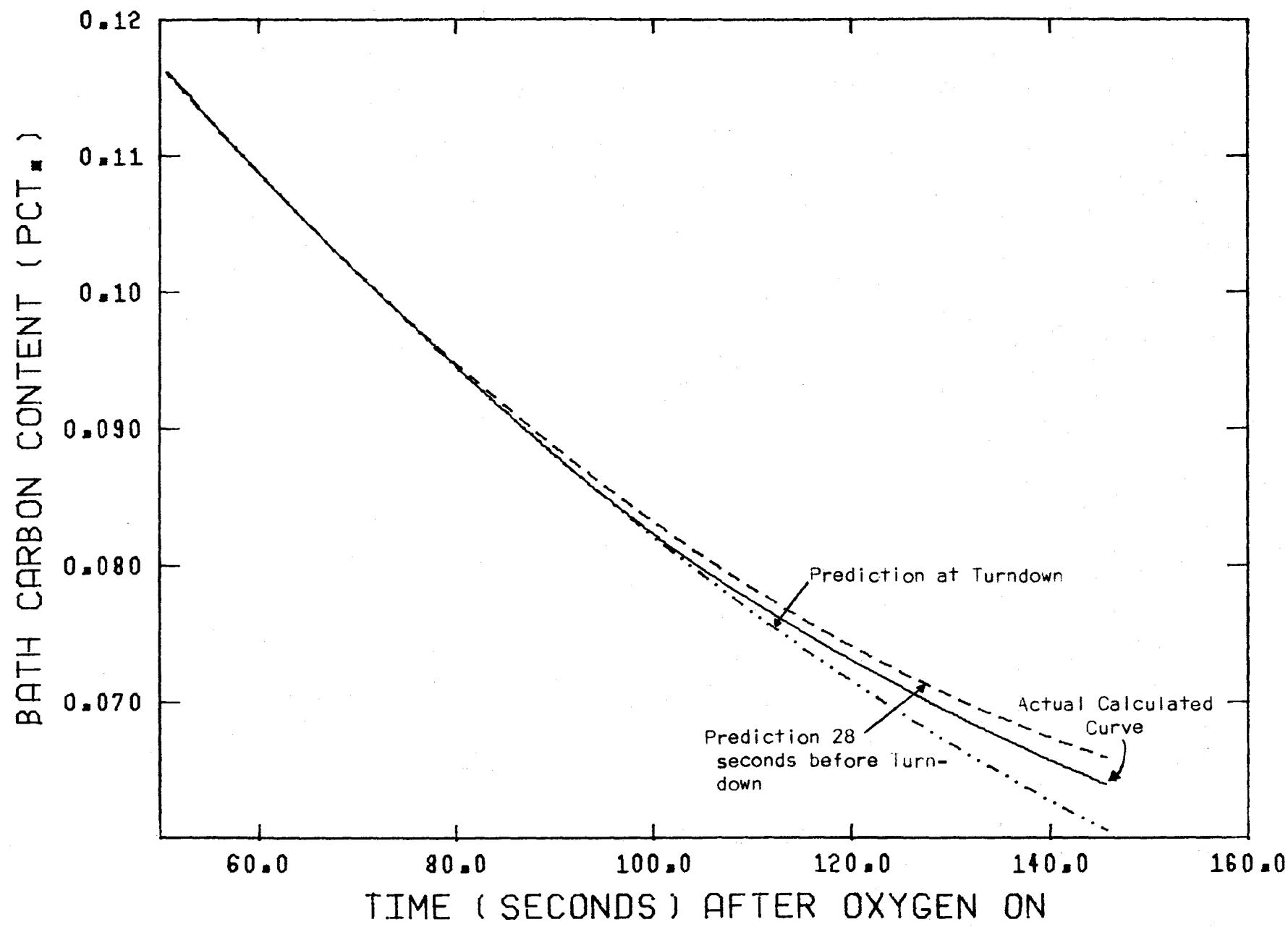


Figure 5.13(b) Predicted Bath Carbon Curves HEAT # 19929

TABLE 5.1

17d

BATH CARBON CONTENTS

	HEAT # 19929	HEAT # 19926
1. Datum point C (assumed)	0.254	0.158
2. End point C (Chemical analysis) at sampling time	0.050	0.055
3. End point C (Mass balance) at		
1) O ₂ off	0.062	0.071
2) sampling time (1.6 minutes after O ₂ off)	0.050	0.055

TABLE 5.2

END POINT PREDICTIONS

o. of fitted data Points	HEAT # 19929			HEAT # 19926		
	Time left (Until O ₂ off)		BCC(t) at O ₂ off- predicted	Time left (until O ₂ off)		BCC(t) at O ₂ off- predicted
	Actual	Predicted		Actual	Predicted	
4	32	21	0.058	32	25	0.066
5	28	46	0.064	28	23	0.067
6	25	39	0.064	25	19	0.066
7	22	27	0.063	22	16	0.067
8	19	22	0.063	19	11	0.065
9	16	19	0.063	16	7	0.063
10	13	12	0.062	13	2	0.062
11	9	4	0.060	9	-2	0.060
12	6	-3	0.059	6	-6	0.060
13	3	-7	0.058	3	-9	0.060
14	0	-10	0.058	0	-12	0.060

are evolved after lance 0₂ is shut off compensates for this. A more detailed study relating this amount of carbon evolved to the carbon level at 0₂ off and the time taken until the bath is sampled, will be necessary to improve the predictive model.

Table 5.2 combined with computer calculations indicate that controlling the furnace by this method would have led to a turndown approximately 6 seconds and 11 seconds too early for HEATS # 19926 and # 19929 respectively; the carbon end point would have been 0.004 % higher for each heat. The solid line curves in Figures 5.12(a) and 5.12(b) (the actual calculated data points) can be fitted essentially perfectly for control purposes as time delay is decreased; thus a shorter time delay will permit more accurate end point predictions.

5.1.4 Variance Analysis

It must be emphasized that the variance analysis derived in Appendix I is an approximation (Viz. the assumed variances in Appendix I (1.1.2)). While it provides estimates of confidence intervals for calculated values, at the moment it serves a more important function by indicating which sources of error deserve the most attention in order to arrive at more accurate results. The four largest sources in order of decreasing importance are VDATMPT, VC02, VCO, and VCH4; VDATMPT is by far the greatest. Equations in Appendix I qualitatively show which data input errors are presently most important; a later extension of this present work will indicate the same information quantitatively, e.g., the percentage of BCC(t) error derived from methane injection pressure error, temperature, etc.

5.1.5 Sample Computer Printout

Appendix II, II.2 contains a sample printout of computer calculations from HEAT # 19923 beginning at the datum point sample time. $CEV(t)$, $CEVD$, and $BCC(t)$ figures (and estimates of their errors - absolute and relative) are listed in detail.

5.2 Conclusions

1. A very accurate method of measuring stack gas flow rates, fundamental to every control equation, utilizing an injected chemical tracer has been developed.
2. A sample mathematical model has been developed, based on data from two B.O.F. heats, which predicts the bath carbon curve (calculated from gas analysis) against time within acceptable error limits ($\pm 0.01 \%$) starting 25 seconds before turndown.
3. Comparison of the end point carbon contents (calculated and analysed) from HEAT # 19923 indicates that B.O.F. control by off gas analysis and an in-blow carbon measurement is a practical possibility.
4. Correlation of lance height and oxygen flow with corbel refractory wear, using the variable $UNUSDO2(t)$, will be possible if gas analysis errors are reduced.
5. Earlier and more accurate end point predictions will be calculated as the time delay of the gas analysis system is reduced.
6. Calculated decarburization curves indicate that slopping elimination will be possible when the time delay is reduced by a factor of 3.

7. For practical B.O.F. control, the average time (4 minutes at present) for carbon datum point sampling should be reduced.

5.3 Recommendations

1. Several dozen more experimental heats must be studied to confirm these preliminary results.
2. Investigate simple alternative mathematical models to determine if more accurate end point predictions are possible.
3. In-blow carbon analysis must be performed by the 19 minute mark to allow end point predictions for all heats.
4. Preliminary correlation of lance height and oxygen flow with $UNUSDO_2(t)$ should be attempted (in spite of the large estimated errors involved) to determine if further expenditure to reduce gas analysis errors is justifiable.
5. A synchronized carbon datum point sampling procedure must be established with B.O.F. operating personnel to minimize loss of production time.
6. Various methods of reducing gas analysis time delays should be thoroughly studied and, wherever possible, used to design a more effective control system.

REFERENCES

1. Ward, R.G., An Introduction to the Physical Chemistry of Iron and Steel-Making, London: Edward Arnold (Publishers) Ltd., 1962.
2. Making, Shaping, and Treating of Steel, The, United States Steel Corporation, Pittsburgh: Herbick and Held, 1971.
3. Kay, D.A.R., Buchanan, D., Mecke, J., Jaskula, H., "B.O.F. Slipping and Fume Formation", Dofasco Internal Report, Research Dept., (December 1970).
4. Kay, D.A.R., private communication with the author.
5. Meyer, H.M., Dukelow, D.A., and Fischer, M.M., "Static and Dynamic Control of the Basic Oxygen Process", Journal of Metals (June 1964) 501-507.
6. Bicknese, Eugene H., "Use of Top Gas Analysis for Predicting Metal Carbon Levels", Open Hearth Proceedings (1970), 172-181.
7. Glasgow, J.A., and Porter, W.F., "Development and Operation of B.O.F. Dynamic Control", Journal of Metals (August 1967) 81-87.
8. Dawson, H., Manufacturing Controls Dept., Dofasco. Private communication to the author.
9. Redbourne, R., Dofasco Internal Report (February 18, 1971).
10. Himmelblau, D.M., Process Analysis by Statistical Methods, Toronto: John Wiley and Sons, Inc., 1968.

APPENDIX I

DERIVATION OF CONTROL EQUATIONS AND ERROR ANALYSES

Variable names in these equations and the computer program (Appendix II) are identical. A variable name prefixed by a "V" becomes the variance of that variable, e.g., VCEV(t) is the variance of the carbon evolution rate CEV(t). When the original variable name is seven characters (maximum on the CDC6400) its variance name is shortened to a recognizable form of the original, e.g., variance of UNUSDO2(t) is VUNUSD. [] after the equation number contains its location in the program.

I.1 Propagation of Errors

I.1.1 Theory

The variance of a function $f(x_1, x_2, x_3, \dots, x_n)$ can be approximated by equation I.1⁽¹⁰⁾.

$$\text{Var}\{f(x_1, x_2, x_3, \dots, x_n)\} = \sum_{i=1}^n \left[\frac{\partial f(x_1, x_2, x_3, \dots, x_n)}{\partial x_i} \right]^2 \text{Var}\{x_i\} \quad (I.1)$$

for example 1) $y = ax + bz \quad \frac{\partial y}{\partial x} = a, \quad \frac{\partial y}{\partial z} = b$

$$\text{Var } \{y\} = a^2 \text{Var } \{x\} + b^2 \text{Var } \{z\} \quad 1.2$$

$$2) \quad y = \frac{ax}{z} \quad \frac{\partial y}{\partial x} = \frac{a}{z}, \quad \frac{\partial y}{\partial z} = -\frac{ax}{z^2}$$

$$\text{Var } \{y\} = \frac{a^2}{z^2} \text{Var } \{x\} + \frac{a^2 x^2}{4z^2} \text{Var } \{z\}$$

$$= \frac{a^2}{z^2} \left[\text{Var } \{x\} + \frac{x^2}{z^2} \text{Var } \{z\} \right] \quad 1.3$$

$$3) \quad y = axz \quad \frac{\partial y}{\partial x} = az, \quad \frac{\partial y}{\partial z} = ax$$

$$\text{Var } \{y\} = a^2 [z^2 \text{Var } \{x\} + x^2 \text{Var } \{z\}] \quad 1.4$$

1.1.2 Assumed Variance Values

VP3 = $(0.5/3.0)^2$ = variance of injected tracer pressure P3 (psig)

VROTFLOW = $(0.1/3.0)^2$ = variance of natural gas injection rate ROTFLOW (cfm)

VNATGAS = $(0.005/3.0)^2$ = variance of natural gas methane analysis NATGAS

VC02 = $(0.3/3.0)^2$ = variance of stack gas CO₂ analysis CO2(%)

VCO = $(0.2/3.0)^2$ = variance of stack gas CO analysis CO(%)

VCH4 = $(0.0005/3.0)^2$ = variance of stack gas CH₄ analysis CH4(%)

VO2 = $(0.25/3.0)^2$ = variance of stack gas O₂ analysis O2(%)

VCH4T = $(2.0/3.0)^2$ = variance of injected tracer temperature CH4T(°F)

VFLW02 = $(600.0/3.0)^2$ = variance of lance oxygen flow FLW02(scfm)
 VTOTWT = $(5000.0/3.0)^2$ = variance of bath weight TOTWT(lb)
 VDATMPT = $(0.02/3.0)^2$ = variance of carbon datum point analysis DATMPT(%)

The assumed variances for stack gas analyses are derived from the manufacturer's specifications - 1% error of full scale analyzer reading.
 Lance oxygen flow variance is obtained from a Dofasco internal report⁽⁹⁾.
 All other variances are reasonable estimates gained by plant experience.

1.1.3 Sample Calculation Data (HEAT #19923)

a) Constants

Average composition of dry air:

$$\text{AIRN2} = 78.09 \%$$

$$\text{AIRO2} = 20.95 \%$$

$$\text{AIRAR} = 0.93 \%$$

$$\text{AIRCO2} = 0.03 \%$$

$$\text{ROTF LW} = 12.75 \text{ cfm}$$

$$\text{NATGAS} = 0.936$$

$$T_1 = 70^{\circ}\text{F} = 530^{\circ}\text{R}$$

$$P_1 = 14.7 \text{ psia}$$

$$P_2 = 57.2 \text{ psia}$$

$$\text{AMOLVOL} = 359.1 \text{ ft}^3/\text{mole } (32^{\circ}\text{F}, 1 \text{ atm.})$$

$$= 386.8 \text{ ft}^3/\text{mole } (70^{\circ}\text{F}, 1 \text{ atm.})$$

TCYCLE = 3.168 seconds
 DATMPT = 0.245 % C
 TOTWT = 319,040 lb
 N2FACTR = AIRN2/(AIRN2 + AIRAR + AIRCO2) = 0.988

b) Data at 25.61 minute work (final oxygen off)

CH4T = 55.4° F = 515.4° R
 CH4 = 0.0177 %
 CO2 = 4.2 %
 CO = 0.08 %
 O2 = 18.4 %

1.2 Flow Measurement

1.2.1(a) Methane Injection Rate

$$\begin{aligned}
 \text{INJRATE} &= \text{ROTFLW} \times \text{NATGAS} \\
 &= (12.75 \text{ cfm})(0.936) = 11.93 \text{ cfm}
 \end{aligned}$$

1.5 [2. 557]

Equation 1.1 gives

$$\begin{aligned}
 \text{VINJRAT} &= \text{ROTFLW}^2 \times \text{VNATGAS} + \text{NATGAS}^2 \times \text{VROTFLW} \\
 &= 0.00045 + 0.00097 \text{ cfm}^2 \\
 &= 0.001431 \text{ cfm}^2
 \end{aligned}$$

1.6 [2. 558]

- INJRATE = methane injection rate (cfm)
 ROTFLW = natural gas injection rate (cfm) from Brooks calibration chart (Figure 1.1) with rotameter reading at 100 mm.
 NATGAS = typical methane analysis of Matheson natural gas

1.2.1(b)

$$SF = \left(\frac{T_1 \times P_2}{CH_4 T \times P_1} \right)^{1/2} \quad 1.7[2. 559]$$

$$= 1.84$$

$$\frac{\partial (SF)}{\partial (CH_4 T)} = -\frac{1}{2} \left(\frac{T_1 \times P_2}{CH_4 T^3 \times P_1} \right)^{1/2} \quad \frac{\partial (SF)}{\partial (P_2)} = \frac{1}{2} \left(\frac{T_1}{CH_4 T \times P_1 \times P_2} \right)^{1/2}$$

$$VSF = \left(\frac{T_1 \times P_2}{4 \times CH_4 T^3 \times P_2} \right) VCH_4 T + \left(\frac{T_1}{4 \times CH_4 T \times P_1 \times P_2} \right) VP_2$$

$$= \left(\frac{T_1 \times P_2}{4 \times CH_4 T \times P_1} \right) \left(\frac{VCH_4 T}{CH_4 T^2} + \frac{VP_2}{P_2^2} \right)$$

$$= \frac{SF^2}{4} \left(\frac{VCH_4 T}{CH_4 T^2} + \frac{VP_2}{P_2^2} \right) \quad 1.8[2. 560]$$

$$= 0.845 (1.68 \times 10^{-6} + 1.26 \times 10^{-5})$$

$$= 1.21 \times 10^{-5}$$

250



BROOKS
INSTRUMENT CANADA LTD.
ONTARIO

CALIBRATION CURVE

TYPE R-10M 25-1

FLOAT: 10-R5-64

SERIAL NO. _____

200

150

MM

100

MILLIMETERS

50

SCFH METHANE SPGR .55 AT 85°F & 14.7 PSIA

200

400

600

800

1000

1200

1400

1600

1800

2000

725-6136-METH-C

Figure 1.1 flowmeter Calibration Chart

SF = sizing factor for correction to standard conditions
 TI = temperature at standard conditions ($^{\circ}$ R)
 PI = pressure at standard conditions (psia)
 P2 = pressure at operating conditions (psia)
 CH4T = temperature at operating conditions ($^{\circ}$ R)

1.2.1(c)

$$\begin{aligned}
 \text{FLWCH4} &= \text{INJRATE} \times \text{SF} & 1.9[2.561] \\
 &= (11.93 \text{ scfm})(1.84) \text{ scfm} \\
 &= 21.93 \text{ scfm}
 \end{aligned}$$

$$\begin{aligned}
 \text{VFLWCH4} &= \text{SF}^2 \times \text{VINJRAT} + \text{INJRATE}^2 \times \text{VSF} & 1.10[2.562] \\
 &= 0.00481 + 0.00172 \text{ scfm}^2 \\
 &= 0.00653 \text{ scfm}^2
 \end{aligned}$$

FLWCH4 = methane injection rate (scfm)

1.2.2 Stack Gas Flow

$$\begin{aligned}
 \text{SFLOW}(t) &= \frac{\text{FLWCH4}}{\text{CH4}} \times 100 & 1.11[2.567] \\
 &= 123,900 \text{ scfm}
 \end{aligned}$$

From 1.3

$$\text{VSFLOW} = \left(\frac{100}{\text{CH}_4} \right)^2 \left[\text{VFLWCH}_4 + \left(\frac{\text{FLWCH}_4}{\text{CH}_4} \right)^2 \text{VCH}_4 \right]$$

$$= 3.19 \times 10^7 (0.00653 + 0.0427) \text{ scfm}^2$$

$$= 1.57 \times 10^6 \text{ scfm}^2$$
1.12 [2. 569]

Relative error of stack gas flow

$$= \frac{3 \times \sqrt{\text{VSFLOW}} \times 100}{\text{SFLOW}(t)}$$

$$= 3.0 \%$$

SFLOW(t) = stack gas flow; dry clean basis (scfm)

CH₄ = stack gas methane analysis (%)

1.3 Carbon Evolution

1.3.1 Rate

$$\text{CEV}(t) = \frac{12 \times (\text{CO}_2 + \text{CO}) \times \text{SFLOW}(t)}{\text{AMOLVOL} \times 100}$$
1.13 [2. 612]

$$= 165 \text{ lb C/min}$$

Combining 1.2 and 1.4 gives

$$\begin{aligned}
 VCEV(t) &= \left(\frac{0.031}{100} \right)^2 [SFLOW(t)^2 (VC02 + VCO) \\
 &\quad + (CO2 + CO)^2 VSFLOW] \\
 &= 9.64 \times 10^{-8} (2.218 \times 10^8 + 2.876 \times 10^7) (lb C/min)^2 \\
 &= 24.1 (lb C/min)^2
 \end{aligned}$$

1.14[2. 613]

Relative error of CEV(t)

$$\begin{aligned}
 &= \frac{3 \times \sqrt{VCEV(t)} \times 100}{CEV(t)} \% \\
 &= 9.0 \%
 \end{aligned}$$

CEV(t) = carbon evolution rate (lb C/min)

AMOLVOL = molar volume at standard conditions (386.8 ft³/mole)

CO₂ = stack gas carbon dioxide analysis (%)

CO = stack gas carbon monoxide analysis (%)

Since the error of individual gas analyses are specified as 1% of full scale analyzer reading (30 %, 20 %, 25 % and 500 ppm for CO₂, CO, O₂ and CH₄ respectively) the relative errors of quantities calculated from these analyses will increase with decreasing component gas contents. Thus for this sample CEV(t) calculation the relative error is approximately double the relative error averaged throughout the heat (~ 5.0 %) because of low CO₂ and CO contents at turndown.

1.3.2 Integration-Trapezoidal Rule

$$\text{CEVD} = \left(\frac{\text{TCYCLE}}{60 \times 2} \right) [\text{CEV}(t_1) + 2 \sum_{i=2}^{n-1} \text{CEV}(t_i) + \text{CEV}(t_n)] \quad 1.15(a)$$

[2. 648]

$= 409 \text{ lb C}$

VCEVD based on 1.1 does not give reasonable results (e.g., integration over a longer time results in a smaller relative error of integration). This suggests that $\text{CEV}(t_i)$ are not independent, a fundamental assumption of 1.1; intuitively the same conclusion will be reached because an individual $\text{CEV}(t_i)$ will always have a value close to that of its nearest neighbours, thus it is not totally independent of them.

One alternative for calculation of VCEVD is to rearrange 1.15(a) by assuming $\text{CEV}(t_1) = \text{CEV}(t_n)$ so that we now have

$$\text{CEVD} = \left(\frac{\text{TCYCLE}}{60 \times 2} \right) [2 \sum_{i=1}^{n-1} \text{CEV}(t_i)] \quad 1.15(b)$$

If $\text{CEV}(t_i)$ were all independent and if $\text{VCEV}(t_i) = \text{VCEV}(t_{i-1})$ for $i = 2, \dots, n$ then the variance of 1.15(b) would be expressed as follows:

$$\text{VCEVD} = \left(\frac{(n-1) \text{TCYCLE}}{60} \right)^2 \text{VCEV}(t_i) \quad 1.16(a)$$

In place of equal variances we can take the latest averaged value of all $\text{VCEV}(t_i)$ and use it to approximate VCEVD, i.e.,

$$\text{VCEVD} = \left(\frac{(n-1) \text{TCYCLE}}{60} \right)^2 \frac{\text{QCEVD}}{n} \quad 1.16(b)$$

[2. 650]

where

$$QCEVD = \sum_{i=1}^n VCEV(t_i) \approx 1897 (\text{lb C/min})^2$$

Thus

$$VCEVD = 323 \text{ lb}^2$$

CEVD = amount of carbon involved at time t_n (lb)

TCYCLE= data collection cycle (seconds)

1.3.3 Bath Carbon Content

- (a) The bath carbon content (lb) at the datum point sampling time is

$$HMCLB = \frac{DATMPT \times TOTWT}{100}$$

1.17 [2. 798]

$$= 782 \text{ lb C}$$

$$VHMCLB = 10^{-4} [DATMPT^2 \times VTOTWT + TOTWT^2 \times VDATMPT]$$

1.18 [2. 799]

$$= 16.7 + 452.4 \text{ lb C}^2$$

$$= 469 \text{ lb C}^2$$

(b) The bath carbon (lb) at time t may be expressed

$$\text{BATHCC} = \text{HMCLB} - \text{CEVD}$$

1.19[2. 835]

$$= 373 \text{ lb C}$$

$$\text{VBATHCC} = \text{VHMCLB} + \text{VCEVD}$$

1.20[2. 836]

$$= 469 + 323 \text{ lb C}^2$$

$$= 792 \text{ lb C}^2$$

(c) % C in bath at time t is

$$\text{BCC}(t) = \frac{100 \times \text{BATHCC}}{\text{TOTWT}}$$

1.21[2. 840]

$$= 0.117 \%$$

1.3 gives

$$\begin{aligned} \text{VBCC}_t &= \left(\frac{100}{\text{TOTWT}} \right)^2 \left[\text{VBATHCC} + \left(\frac{\text{BATHCC}}{\text{TOTWT}} \right)^2 \text{VTOTWT} \right] \\ &= 9.82 \times 10^{-8} (792 + 4) \%^2 \\ &= 7.81 \times 10^{-5} \%^2 \end{aligned}$$

1.22[2. 842]

Thus $\text{BCC}(t) = 0.117 \pm 0.027 \%$

1.4 Oxygen Distribution

1.4.1 Unused Lance Oxygen

At time t we have:

(a) Nitrogen content by difference expressed as

$$\begin{aligned} N_2 &= [100 - (CO_2 + CO + O_2)] N2FACTR \\ &= 76.4 \% \end{aligned}$$

1.23[2. 437]

$$\begin{aligned} VN_2 &= (VCO_2 + VCO + VO_2) N2FACTR^2 \\ &= 0.021 \%^2 \end{aligned}$$

1.24[2. 547]

N₂ = stack gas nitrogen content (%) adjusted for air, argon and CO₂ content

O₂ = stack gas oxygen analysis (%)

N2FACTR = correction factor to account for air argon and CO₂ content

(b) Air flow into stack (scfm) is

$$\text{AIRFLW} = \frac{N_2 \times S\text{FLOW}(t)}{\text{AIRN}_2} \quad 1.25[2. 573]$$

$$= 121,200 \text{ scfm}$$

$$V\text{AIRFLW} = \frac{S\text{FLOW}(t)^2 \times V\text{N}_2 + N_2^2 \times V\text{SFLOW}}{\text{AIRN}_2^2} \quad 1.26[2. 574]$$

$$= 52,600 + 1.50 \times 10^6 \text{ scfm}^2$$

$$= 1.55 \times 10^6 \text{ scfm}^2$$

(c) Stack air oxygen flow (scfm)

$$S\text{AO}_2 = \frac{\text{AIRO}_2 \times \text{AIRFLW}}{100} \quad 1.27[2. 577]$$

$$= 25,400 \text{ scfm}$$

$$V\text{SAO}_2 = \left(\frac{\text{AIRO}_2}{100} \right)^2 V\text{AIRFLW} \quad 1.28[2. 578]$$

$$= 68,200 \text{ scfm}^2$$

(d) UNUSD02(t)

By assuming no $\text{CO} \rightarrow \text{CO}_2$ combustion in the furnace, the amount of air oxygen remaining in the stack

$$= SAO_2 - \frac{0.5 CO_2 \times SFLOW(t)}{100} \quad 1.29$$

$$\text{Stack oxygen} = \text{unused air oxygen} + \text{unused lance oxygen} \quad 1.30$$

Rearranging 1.30 we have

$$UNUSDO_2(t) = \text{stack oxygen} - \text{unused air oxygen}$$

$$= \frac{O_2 \times SFLOW(t) + 0.5 CO_2 \times SFLOW(t)}{100} - SAO_2$$

$$\text{Substituting for } SAO_2 \text{ from 1.27 and defining } O2BYN2 = \frac{AIRO_2}{AIRN_2} \times N2FACTR$$

$$\begin{aligned} UNUSDO_2(t) &= \frac{SFLOW(t)}{100} [CO_2(0.5 + O2BYN2) + CO \times O2BYN2 + O_2(1+O2BYN2) \\ &\quad - 100 \times O2BYN2] \quad 1.31[2.579] \\ &= 10.5 \text{ scfm} \end{aligned}$$

$$\begin{aligned} VUNUSD &= \frac{SFLOW(t)^2 [(0.5 + O2BYN2)^2 VC_02 + O2BYN2^2 \times VC_0 + (1+O2BYN2)^2 VO_2]}{100^2} \\ &\quad + \left(\frac{UNUSDO_2(t)}{SFLOW(t)} \right)^2 VSFLOW \quad 1.32[2.583] \\ &\approx 26,500 + 0 \text{ scfm}^2 \\ &\approx 26,500 \text{ scfm}^2 \end{aligned}$$

1.4.2 PRL02

(a) Lance oxygen recovered as CO₂, CO and unused lance O₂ (scfm) is

$$RL02 = \frac{(0.5 CO_2 + 0.5 CO)SFLOW(t)}{100} + UNUSDO2(t)$$

1.32(a)
[2. 585]

Substituting for UNUSDO2(t)

$$RL02 = \frac{[(1 + 02BYN2)(CO_2 + O_2) + (0.5 + 02BYN2)CO - 100 \times 02BYN2]SFLOW(t)}{100}$$

1.32(b)

$$= 2660 \text{ scfm}$$

$$VRL02 = \frac{SFLOW(t)^2 [(1 + 02BYN2)^2 (VC02 + VO2) + (0.5 + 02BYN2)^2 VCO]}{100} + \left(\frac{RL02}{SFLOW(t)} \right)^2 VSFLOW$$

1.33 [2. 588]

$$= 45,640 + 720 \text{ scfm}^2$$

$$= 46,400 \text{ scfm}^2$$

(b) Percentage recovered lance oxygen is

$$PRL02 = \frac{RL02}{FLW02(t)} \times 100$$

1.34(a)
[2. 592]

$$= \frac{[1.265 (CO_2 + O_2) + 0.765 CO - 26.50]SFLOW(t)}{FLW02(t)} \times 100 \quad 1.34(b)$$

$$= 24.8 \%$$

$$\text{VPRLO2} = \left(\frac{100}{\text{FLW02}(t)} \right)^2 \left[\text{VRL02} + \left(\frac{\text{RL02}}{\text{FLW02}(t)} \right)^2 \text{VFLW02} \right] \quad 1.35 [2. 593]$$

$$= 4.0 + 0.2 \%^2$$

$$= 4.2 \%^2$$

This gives a relative error for PRL02 of $\sim 25\%$. Reasoning identical to that used for the CEV(t) (1.3.1) explains why this error is well above the heat average of $\sim 13\%$.

11.1 Listing

PROGRAM DOFASCO(INPUT=103,OUTPUT=103,TAPE5=INFUT,TAPE6=CUTFLT
1 TAFF10=1004)

PROGRAM LIMITATIONS - 1) MEMORY SIZE OF MACHINE.
2) TIME LIMIT.

ANY NO. OF HEATS CAN BE STUDIED IN SEQUENCE.
DYNAMIC STORAGE ALLOCATION IS USED TO MINIMIZE CORE.

SUBROUTINES CALLED: 1) CONT

```

DIMENSION ENDPLT(3)
COMMON NDUUD1,DUD2,DUD3,XX(33000)
COMMON /JPLOTS/NOPLOTS
COMMON /TITLES/MNYPLT,TITLE1(2),TITLE2(2),TITLE3(2),TITLE4(2)
COMMON /NBLANK/I1,I2,I3,I4,I5,I6,I7,I8,I9,I10,I11,I12,FLOTS
COMMON /READ/KR/WRITE/KW
COMMON /MDL/ MODEL 7(3), MODEL 8(3), MODEL 9(3), MCDEL 10(3), MCDL(3),
1      MCDEL1(3),MCDEL2(3),MCDEL3(3),MCDEL4(3),MCDEL5(3),MCDEL6(3)
COMMON/CPTIME/TIME2,TIME3,TIME4,TIME5,TIME15,TIME7,TIME14,TIME12,
1      AVGTIME,TIME13,TIME16
DATA ENDPLT/10HEND OF'PLO,10HT JOB -- T,10HHANK YOU /

```

```
REAL MISDTIM
CALL SECOND (TIMEA)
KW=6
KR=5
MNYFLIT = 1 IMPLIES THAT A TITLE WILL BE PRINTED BY THE FLOTTER.
MNYPLT=1
TITLES FOR PLOTS (E.G. NAME, DEPT., JOB DESCRIPTION, DATE).
CONTINUE
READ 7, TITLE1,TITLE2,TITLE3,TITLE4
IF (EOF,5) 4,3
CONTINUE
```

MATHEMATICAL MODELS ARE READ IN FOR LATER DESCRIPTIVE PRINTOUT.
READ (KR,6) (MODEL 1(I), I=1,J3)
READ (KR,6) (MODEL 2(I), I=1,J3)
READ (KR,6) (MODEL 3(I), I=1,J3)
READ (KR,6) (MODEL 4(I), I=1,J3)
READ (KR,6) (MODEL 5(I), I=1,J3)
READ (KR,6) (MODEL 6(I), I=1,J3)
READ (KR,6) (MODEL 7(I), I=1,J3)
READ (KR,6) (MODEL 8(I), I=1,J3)
READ (KR,6) (MODEL 9(I), I=1,J3)
READ (KR,6) (MODEL 10(I), I=1,J3)

NDATA IS THE NO. OF STORAGE LOCATIONS PER ARRAY.
READ (KR,8) NDATA

DIMENSIONING IN BLANK COMMON. DYNAMIC STORAGE ALLOCATION.

```

I1=1          1.  58
I2=I1+NDATA  1.  59
I3=I2+NDATA  1.  60
I4=I3+NDATA  1.  61
I5=I4+NDATA  1.  62
I6=I5+NDATA  1.  63
I7=I6+NDATA  1.  64
I8=I7+NDATA  1.  65
I9=I8+NDATA  1.  66
I10=I9+NDATA 1.  67
I11=I10+NDATA 1.  68
I12=I11+NDATA 1.  69
I13=I12+NDATA 1.  70
I14=I13+NDATA 1.  71
I15=I14+NDATA 1.  72
I16=I15+NDATA 1.  73
C
CCC AMOUNT OF BLANK COMMON USED.
NAMT=16*NDATA 1.  74
WRITE (KW,9) NDATA,NAMT 1.  75
CC
CALL CONTRL (XX(I1),XX(I2),XX(I3),XX(I4),XX(I8),XX(I6),XX(I7),XX(I
15),XX(I2),XX(I3),XX(I4),XX(I5),XX(I8),NDATA,XX(I1),XX(I3),XX(I1),
2XX(I12),XX(I14),XX(I14),XX(I15),XX(I1),XX(I3),XX(I7),XX(I16)) 1.  81
CALL SECOND (TIMEB)
TIME1=TIME3-TIMEA
CAL TIME=TIME2+TIME3+TIME4+TIME5+TIME7+TIME12+TIME13+TIME14+TIME15+
1 TIME16
MISOTIM=TIME1-CAL TIME 1.  87
CCC ANALYSIS OF COMPUTATION TIME IS PRINTED OUT HERE.
WRITE (KW,11) TIME2,TIME3,TIME4,TIME5,TIME15,TIME7,TIME14,TIME12,F
1LOT,AVGTIME,TIME13,TIME16,CALTIME,TIME1,MISOTIM 1.  90
WRITE (KW,10) 1.  91
GO TO 2 1.  92
CONTINUE 1.  93
IF (NCFLOTS.EQ.1) GO TO 5 1.  94
HI=0.6 1.  95
CALL PLOT (0.0,0.0,999) 1.  96
CONTINUE 1.  100
STOP 1.  101
FORMAT (5A10) 1.  102
FORMAT (4(2A10)) 1.  104
FORMAT (I3) 1.  105
FORMAT (1H1,54X,*NO. OF OBSERVATIONS = *,I3/4EX,*AMOUNT OF BLANK C
10MMON USED (OCTAL) = *,C6//) 1.  106
FORMAT (1H1) 1.  107
FORMAT (1H1,11(/),38X,*ANALYSIS OF COMPUTATION TIME 1.  108
1.CN T I M E*/38X,58(1H-)//43X,*1. DATA READ-IN, WRITE-CLT * 1.  109
2.*((C 2000) = *,F6.3//43X,*2. SUBROUTINE CALIBR (MV. TO * * 1.  110
3ENG. UNITS) = *,F6.3//43X,*3. SUBROUTINE ARANGE (ALLOW FOR **TIME 1.  111
4LAG) = *,F6.3//43X,*4. CALCULATION OF MIN-MAX TABLES*,10X,*= *,F 1.  112
56.3//43X,*5. PRINTOUT OF CONVERTED DATA (DO 9080) = *,F6.3//43X 1.  113
6,*6. CALC. OF VAR. (IGNITION TO 1ST TURN. */47X,*SAMPLE TIME) AND 1.  114

```

7 PRINTOUT (DO 2003) = *,F6.3//43X,*7. CALC. OF VAR. (DATUM FT. 1. 116
8 SAMPLE TIME TO */47X,*1ST TURNDOWN SAMPLE TIME)*,14X,*=*,F7.3//43 1. 117
9X,*8. TOTAL PLOTTING*,* TIME*,20X,*=*,F7.3//43X,*9. TOTAL NO. OF 1. 118
\$ PLCTS*,21X,*=*,I7//42X,*10. AVERAGE CP TIME PER PLOT*,15X,*=*,F 1. 119
\$6.3//42X,*11. TIME USEC IN CURVE FITTING (DO 2007) =*,F7.3//42X 1. 120
\$,*12. ITERATION TIME*,25X,*=*,F7.3//63X,*CALCULATED TOTAL *,*TIME 1. 121
\$,*=*,F7.3//65X,*MEASURED TOTAL TIME =*,F7.3//64X,*TIME UNACCOUNTE 1. 122
\$D FCR =*,F7.3)
END 1. 123
1. 124-

SUBROUTINE CONTRL (SC02,SCO,SCH4,SO2,SN2,FLW02,HTLANS,TCH4,PRLC2,F
 1XSAIR,CEV,BCC,TME,N,SFLCW,UNUSDO2,EQU,VCEV,BOUNDS,UPPER,LCKER,DLM
 2E1,DUMIE2,DUMIE3,SDR)

CONTRL IS THE MAIN SUBROUTINE. IT IS RESPONSIBLE FOR ALL CALC.

SUBROUTINE CALLED BY: 1) DOFASCO

SUBROUTINES CALLED: 1) READ1
 2) SECOND
 3) CALIBRT
 4) ARANGE
 5) XDIM
 6) NPLOTS
 7) LESQ
 8) EQUATN
 9) NPLCT1
 10) ITER8

ARRAY DEFINITIONS:	SCC2	- STACK GAS CARBON DIOXIDE (MV.).	2.	1
	SCC	- MV. CONVERTED TO PCT. BY CALIBRT.	2.	2
	SCH4	- STACK GAS CARBON MONOXIDE (MV.).	2.	3
	SO2	- MV. CONVERTED TO PCT. BY CALIBRT.	2.	4
	SN2	- STACK GAS METHANE (MV.).	2.	5
	FLW02	- MV. CONVERTED TO PCT. BY CALIBRT.	2.	6
	HTLANS	- STACK GAS OXYGEN (MV.).	2.	7
	TCH4	- STACK GAS NITROGEN (PCT.).	2.	8
	FRLO2	- CALCULATED BY DIFFERENCE.	2.	9
	FXSAIR	- LANCE OXYGEN FLCW RATE (MV.).	2.	10
	CEV	- MV. CONVERTED TO SCFM. BY CALIBRT.	2.	11
	ECC	- LANCE HEIGHT ABOVE STATIC EATH (MV.).	2.	12
	TME	- MV. CONVERTED TO FEET BY CALIBRT.	2.	13
	SFLOW	- TEMP. OF INJECTED METHANE (MINUTES).	2.	14
	UNUSDO2	- MIN. CONVERTED TO DEG. RANKINE BY CALIBRT.	2.	15
	EQU	- PERCENTAGE RECOVERED LANCE OXYGEN. (PERCENTAGE OF LANCE OXYGEN RECOVERED AS CO AND CO ₂ IN THE STACK).	2.	16
	VCEV	- PERCENTAGE EXCESS AIR.	2.	17
	BOUNDS	- AMT. OF AIR DRAWN INTO STACK OVER AMT. NEEDED FOR BURNING CO TO CO ₂ .	2.	18
	UPPER	- CARBON EVOLUTION RATE (LB. C./MIN.).	2.	19
		- EATH CARBON CONTENT (PCT.).	2.	20
		- TIME (MINUTES).	2.	21
		- STACK GAS FLOW RATE (SCFM.) ON A DRY, CLEAN BASIS.	2.	22
		- AMT. OF LANCE O ₂ USED FOR CC TO CC ₂ COMBUSTION IN THE FURNACE.	2.	23
		- CUMMY ARRAY USED BY LESQ.	2.	24
		- USED BY EQUATN TO CALCULATE FITTED DEPENDENT VARIABLE.	2.	25
		- VARIANCE OF DATA IN CEV.	2.	26
		- CONFIDENCE LIMITS OF CALCULATED EATH CARBON CONTENTS (BCC(J)).	2.	27

LOWER	- UPPER CONFIDENCE LIMIT.
	- ECC(J) - BOUNDS(J).
SDR	- LOWER CONFIDENCE LIMITS.
	- SPECIFIC DECARDURIZAZION RATE.
	- (D(BCC(J))/DT)/FLWO2(J)
DUMIE1	- PCT. C/CU. FT. C ²
DUMIE2	- DUMMY STORAGE ARRAY.
DUMIE3	- DUMMY STORAGE ARRAY.

ARRAY STARTING ADDRESSES:

I1	- SCO2, SFLOW, EQU, DUMIE1
I2	- SCO, PRLO2
I3	- SCH4, PXSAIR, DUMIE2
I4	- SO2, CEV
I5	- TCH4, BCC
I6	- FLWO2
I7	- HTLANS, DUMIE3
I8	- SN2, TME
I9	- NONE USED FCR PLOTTING
I10	- NONE USED FCR PLOTTING
I11	- NONE USED FCR PLOTTING
I12	- VCEV USED FCR PLOTTING
I13	- UNUSDO2 USED FCR PLOTTING
I14	- BOUNDS, UPPER USED FCR PLOTTING
I15	- LOWER USED FCR PLOTTING
I16	- SDR USED FCR PLOTTING

THE VARIANCE OF A VARIABLE IS DENOTED BY THE NAME OF THE VARIABLE PREFIXED BY `#V#`, E.G. VARIANCE OF CEV(J) IS VCEV(J). WHERE ORIGINAL NAME WAS `#` CHARACTERS THE NAME OF THAT VARIABLE'S VARIANCE HAS BEEN SHORTENED TO A RECOGNIZABLE FORM OF THE ORIGINAL, E.G. VARIANCE OF UNUSDO2(J) IS VUNUSD. THE SAME HOLDS TRUE FOR THE FOLLOWING PREFIXES:

S - STANDARD DEVIATION OF VARIABLE.
 C - CONFIDENCE LIMIT OF VARIABLE.
 P - FCT RELATIVE ERROR OF VARIABLE.

CERTAIN VARIABLES ARE SUFFIXED BY A 1 OR A 2. THESE VARIABLES ARE USED IN CALCULATING INTEGRATED TOTALS. SUFFIX 1 INDICATES THE PRESENT VALUE OF VARIABLE. SUFFIX 2 INDICATES PREVIOUS VALUE OF VARIABLE (THEREBY AVOIDING EXCESS SUBSCRIPTING).

```

1 DIMENSION SCO2(N), SCO(N), SCH4(N), SO2(N), SN2(N), FLWO2(N), HTLANS(N),
2 TCH4(N), PRLC2(N), PXSAIR(N), CEV(N), BCC(N), TME(N),
3 SFLOW(N), UNUSDO2(N), VCEV(N), SDR(N),
4 BOUNDS(N), UPER(N), LOWER(N), DUMIE1(N), DUMIE2(N),
      DUMIE3(N), EQU(N)
DIMENSION B(10), BI(10), NPROB(5), COEFF(5), COEFFN(5)
DIMENSION DUD1(10), DUC2(10)
COMMON /MDL/ MODEL7(3), MODEL8(3), MODEL9(3), MCDEL10(3), MCDL(3),
1 MODEL1(3), MODEL2(3), MODEL3(3), MCDEL4(3), MCDEL5(3), MCDEL6(3),
COMMON /TITLES/MNYPLT, TITLE1(2), TITLE2(2), TITLE3(2), TITLE4(2)
COMMON /NBLANK/I1, I2, I3, I4, I5, I6, I7, I8, I9, I10, I11, I12, PLCTS
COMMON /JPLCTS/NOPLCFS
COMMON /READ/KR/ WRITE/KW
COMMON NDUD1, DUD4, DUD3, XX(33000)
COMMON /CONVRT/LANSHT(16), HTINFT(16), FLWMV(16), FLWCFT(16),

```

1	CO2PCT(16), CO2MV(16), COPCT(16), COMV(16), TCH4MV(56),	2.	116
2	TCH4T(56), CH4PCT(16), CH4MV(16), CH4PPM(16)	2.	117
	COMMON /CONNO/NHT, NFLW, NC02, NCO, NCH4, NTCH4	2.	118
	COMMON /HTGPD/ HEATNO(2), GRADE1(2)	2.	119
	COMMON /NDODAD/ NDO	2.	120
	COMMON /DOPLTS/MPL OT, NPT	2.	121
	COMMON/LBL/YLVERT(7), YRVERT(7), XLHCR(7), XUHOR(7)	2.	122
	COMMON/EFORMAT/FMTE	2.	123
1	COMMON/CPTIME/TIME2, TIME3, TIME4, TIME5, TIME15, TIME7, TIME14, TIME12,	2.	124
	AVGTIME, TIME13, TIME16	2.	125
	DATA HEATNO(1), GRADE1(1)/10HHEAT NO. ,10HGRADE /	2.	126
	DATA BLANK/4H /	2.	127
	DATA BLANK10/10H /	2.	128
14	DATA CCEFF(1), COEFF(2), COEFF(3), COEFF(4), COEFF(5)/4HALFA, 4HBETA,	2.	129
	14HGAMA, 4HPIE, 4HNETA/	2.	130
	DATA COEFFN(1), COEFFN(2), COEFFN(3)/5HALPHA, 5HBETA, 5HGAMMA/	2.	131
	DATA COEFFN(4), COEFFN(5)/5HPIE, 5HNETA /	2.	132
	INTEGER PCHANGE, PLOTS, ADJFLW	2.	133
	REAL LANSHT, N2MAX, N2MIN, NATGAS, LIME, MNMLWT, MNRMVD, N2FACTR, INJRATE	2.	134
	REAL N2, MAXLAG, MULTFLY, LOWER	2.	135
C	READ (KR,114) HEAT, HMWT, SCRAP, SPECC, HMSI, HMTEMP, ORDING	2.	136
C	READ (KR,114) GRADE, DATMPT, LIME, DOLC, SPAR, TOTWT	2.	137
C	READ (KP,111) HMMN, STALMN, STALSI, STALC	2.	138
C	READ (KR,111) CATMPTIM, TURN1, PTIME1, FTIME2, PRESTR	2.	139
C	READ (KR,112) FCHANGE, ROTFLW1, P3, ROTFLW2, P4, PTIME, ADJUST, CH4FCTR	2.	140
C	READ (KP,113) NSTART, JI1, INHIBIT, NCPLOTS, NOFITS, II2, NW8TS, ACJFLW	2.	141
C	WRITE (KW,116)	2.	142
C	WRITE (KW,115) HEAT, HMWT, SCRAF, SPECC, HMSI, HMTEMP, ORDING	2.	143
C	THE FOLLOWING ARRAYS INTO WHICH DATA IS READ ARE DEFINED ABOVE.	2.	144
	NHT=16	2.	145
	CALL READ1 (LANSHT, HTINFT, NHT)	2.	146
	NFLW=16	2.	147
	CALL READ1 (FLWMV, FLWCFT, NFLW)	2.	148
	NC02=16	2.	149
	CALL READ1 (CO2MV, CC2FCT, NC02)	2.	150
	NCO=16	2.	151
	CALL READ1 (COMV, COFCT, NCO)	2.	152
	NCH4=16	2.	153
	CALL READ1 (CH4MV, CH4FPM, NCH4)	2.	154
	NTCH4=56	2.	155
	CALL READ1 (TCH4MV, TCH4T, NTCH4)	2.	156
	TEMFO=459.69	2.	157
	TEMPI=32.0	2.	158
	TEMPO=70.0	2.	159
CCC	COMPOSITION OF DRY AIR	2.	160
C	AIRN2=7.09	2.	161
C	AIRO2=20.95	2.	162
C	AIRAR=0.93	2.	163
C	AIRC02=0.03	2.	164
C		2.	165
C		2.	166
C		2.	167
C		2.	168
C		2.	169
C		2.	170
C		2.	171
C		2.	172
C		2.	173

C	CALCULATION OF MOLECULAR VOLUME FROM METRIC UNITS	2.	174
	ALITER=22.414	2.	175
	GPERLE=0.0022046	2.	176
	CUFT=ALITER*0.03532	2.	177
	AMOL VOL=CUFT/GPERLB	2.	178
C	MOLECULAR VOLUME ADJUSTED FOR SCFM. CALCULATIONS.	2.	179
	AMOLVCL=(TEMP0+TEMP2)/(TEMP0+TEMP1)*AMOLVOL	2.	180
C		2.	181
C		2.	182
C	ATOMIC WEIGHTS	2.	183
	CMLWT=12.011	2.	184
	O2MLWT=15.994	2.	185
C		2.	186
C		2.	187
C	GAS ANALYSIS DELAY TIMES (SECONDS)	2.	188
	CO2DLY=18.2+32.7	2.	189
	CO2DLY=19.0+32.7	2.	190
	CH4DLY=21.2+32.7	2.	191
	O2DLY=26.0+32.7	2.	192
	MAXLAG=CH4DLY	2.	193
C		2.	194
C		2.	195
C	ASSUMED VARIANCE VALUES.	2.	196
	ABC=3.0	2.	197
	VHMWT=(5000.0/ABC)**2	2.	198
	VHMTMP=(40.0/ABC)**2	2.	199
	VHMSI=(0.05/ABC)**2	2.	200
	VSCRCAF=(1000.0/ABC)**2	2.	201
	VSCRAPC=(0.05/ABC)**2	2.	202
	VP3=(0.5/ABC)**2	2.	203
	VROTFL=(0.1/ABC)**2	2.	204
	VNATGAS=(0.005/ABC)**2	2.	205
	VCO2=(0.3/ABC)**2	2.	206
	VCO=(0.2/ABC)**2	2.	207
	VO2=(0.25/ABC)**2	2.	208
	VCH4=(0.0005/ABC)**2	2.	209
	VCH4T=(2.0/ABC)**2	2.	210
	VFLW02=(600.0/ABC)**2	2.	211
	VTOTWT=(5000.0/ABC)**2	2.	212
	VDATMPT=(0.02/ABC)**2	2.	213
C		2.	214
C		2.	215
C	VARIABLES USED FOR ANALYSIS OF COMPUTATION TIME ARE	2.	216
	INITIALIZED TO ZERO HERE.	2.	217
	TIME1=0.0	2.	218
	TIME2=0.0	2.	219
	TIME3=0.0	2.	220
	TIME4=0.0	2.	221
	TIME5=0.0	2.	222
	TIME6=0.0	2.	223
	TIME7=0.0	2.	224
	TIME8=0.0	2.	225
	TIME9=0.0	2.	226
	TIME10=0.0	2.	227
	TIME11=0.0	2.	228
	TIME12=0.0	2.	229
	TIME13=0.0	2.	230
	TIME14=0.0	2.	231

TIME15=0.0
TIME16=0.0

CC
ADDEDFE=5000.0
TOTWT=TOTWT+ADDEDFE
B1=CMLWT/(AMOLVCL*100.0)
T1=TEMP0+85.0
P1=14.696
P2=P1+P3
ROTF LW=ROTF LW1
N2FACTR=AIRN2/(AIRN2+AIRAR+AIRCO2)
TCYCLE=3.168
NTCYCLE=TCYCLE
NATGAS=0.93E
TFLW02=8000.0
XTEST=1400.0
FLWTST=350000.0
LINES=52
NTIME=0
HEATNO(2)=HEAT
GRADE1(2)=GRADE

CC
IF (INHIBIT.GE.1) GO TO 1
1 WRITE (KW,117) HEATNO(1),HEATNO(2),GRADE1(1),GRADE1(2)
CONTINUE
JN=0
N1=700

CCC
READ IN DATA
THE FOLLOWING ARRAYS INTO WHICH DATA IS READ ARE DEFINED ABOVE.

CALL SECOND (TIMEA)

DO 10 I=1,N1

IF (I.GT.N) GO TO 2

GO TO 3

2 WRITE (KW,119)

STOP 1

3 CONTINUE

ILINES=J/LINES

ILINES=ILINES*LINES

IF (I.EQ.ILINES.AND.INHIBIT.EQ.0) WRITE (KW,118)

READ (KR,120) SCO2(I),SCO(I),SCH4(I),S02(I),FLW02(I),HTLANS(I)

TIME=FLOAT(I-1)*TCYCLE

TIME=TIME/60.0

TCH4(I)=TIME

TME(I)=TIME

SCH4(I)=SCH4(I)-CH4FCTR

IF (ADJFLW.EQ.1) SCH4(I)=SCH4(I)+ADJUST

IF (INHIBIT.EQ.0) WRITE (KW,121) I,SCO2(I),SCO(I),SCH4(I),S02(I),F

1LW02(I),HTLANS(I),TIME

IF (JN-1) 4,6,8

CC
TIME = DATMTIM IMPLIES THAT THE DATUM POINT
SAMPLING TIME HAS BEEN REACHED.
IF (TIME-DATMTIM) 10,5,5

2. 232
2. 233
2. 234
2. 235
2. 236
2. 237
2. 238
2. 239
2. 240
2. 241
2. 242
2. 243
2. 244
2. 245
2. 246
2. 247
2. 248
2. 249
2. 250
2. 251
2. 252
2. 253
2. 254
2. 255
2. 256
2. 257
2. 258
2. 259
2. 260
2. 261
2. 262
2. 263
2. 264
2. 265
2. 266
2. 267
2. 268
2. 269
2. 270
2. 271
2. 272
2. 273
2. 274
2. 275
2. 276
2. 277
2. 278
2. 279
2. 280
2. 281
2. 282
2. 283
2. 284
2. 285
2. 286
2. 287
2. 288
2. 289

```

5      CONTINUE
N24=I
N7=I
JN=1
GO TO 10
C
C
TIME = TURN1 IMPLIES THAT THE FIRST TURNDOWN
SAMPLING TIME HAS BEEN REACHED.
CONTINUE
IF (TIME-TURN1) 10,7,7
CONTINUE
N3=I
JN=2
GO TO 10
C
C
SC02(I) = 992.0 IMPLIES THAT THE END OF THE DATA DECK HAS BEEN REA
8 IF (SC02(I)-992.0) 10,9,10
9 I=I-1
NN12=I
GO TO 11
10 CONTINUE
11 CONTINUE
CALL SECOND (TIMEB)
TIME2=TIMEB-TIMEA
IF (II2.GT.1) GO TO 12
C
C
ENTRANCE TO 3039 ALLOWS TESTING TO DETERMINE
EFFECT OF SAMPLING EVERY (II2*3) SECONDS.
12 GC TO 15
CONTINUE
II4=0
DO 14 I=1,N3
II3=I/II2
II3=I*II3
IF ((I.EQ.II3).OR.(I.EQ.1)) GC TO 13
13 GC TO 14
CONTINUE
II4=II4+1
SC02(II4)=SC02(I)
SC0(II4)=SC0(I)
SCH4(II4)=SCH4(I)
SO2(II4)=SO2(I)
FLWC2(II4)=FLWC2(I)
HTLANS(II4)=HTLANS(I)
TCH4(II4)=TCH4(I)
14 CONTINUE
N24=N24/II2
N7=N24
N3=N3/II2
NN12=NN12/II2
TCYCLE=TCYCLE*FLOAT(II2)
15 CONTINUE
CALL SECOND (TIMEA)
C

```

```

C THE ARRAYS SCO2,SCO,S02,SCH4,FLW02, AND HTLANS ARE NOW
C CONVERTED FROM MILLIVOLTS READINGS TO ENGINEERING UNITS.
C TCH4 IS CONVERTED FROM TIME (MINUTES) TO TEMPERATURE (DEG. R).
C CALL CLIBRT (SCO2,SCO,SCH4,S02,FLW02,HTLANS,TCH4,NN12)
C CALL SECOND (TIMEB)
C TIME3=TIMEB-TIMEA
C CALL SECOND (TIMEA)

C ARANGE ADJUSTS THE INPUT ARRAYS TO ACCOUNT FOR TIME DELAYS.
C CALL ARANGE (SCO2,SCO,SCH4,S02,CO2DLY,C00LY,CH4DLY,O2DLY,TCYCLE,NN
112)
C CALL SECOND (TIMEB)
C TIME4=TIMEB-TIMEA
DO 16 I=1,N24
J=N24-I+1
IF (FLW02(J).LT.TFLW02) FLW02(J)=0.0
IF (FLW02(J).NE.0.0) GO TO 17
16 CONTINUE
17 CONTINUE
N30=J
DO 18 I=1,N3
J=N3-I+1
IF (FLW02(J).LT.TFLW02) FLW02(J)=0.0
IF (FLW02(J).NE.0.0) GO TO 19
18 CONTINUE
19 N32=J
DO 20 I=N7,N32
IF (FLW02(I).LT.TFLW02) FLW02(I)=0.0
IF (FLW02(I).NE.0.0) GO TO 21
20 CONTINUE
21 NN11=I-N7-1

C IF (NOPLOTS.LE.0) WRITE (KW,124)
PLOTS=1
CALL SECOND (TIMEA)
IF (NOPLOTS.NE.-1) GO TO 22
GO TO 23
22 READ (KR,125)
GO TO 24
23 PLOTS=0

C THESE PLOTS CORRELATE MILLIVOLTS TO ENGINEERING UNITS.
CALL NPLOTS (1,NHT,200,1,1,HTINFT,LANSHT,I9,0,0,0,1)
CALL NPLOTS (1,NFLW,200,1,1,FLWCF,FLWMV,I9,0,0,0,1)
CALL NPLOTS (1,NCH4,200,1,1,CH4PCT,CH4MV,I9,0,0,0,1)
CALL NPLOTS (1,NC02,200,1,1,COFC,COMV,I9,0,0,0,1)
CALL NPLOTS (1,NC02,200,1,1,CO2PCT,CO2MV,I9,0,0,0,1)
PLOTS=PLOTS+5
24 CONTINUE
IF (NPLOTS.EQ.1) GO TO 27
IF (NPLOTS.EQ.0) PLOTS=0

C XCIM PLOTS CO2, CO AND C02 STACK GAS ANALYSES (FCT) VS. TIME.
C THE NEXT THREE PLOT LANCE O2 FLOW, LANCE HEIGHT ABOVE

```

2. 348
 2. 344
 2. 345
 2. 346
 2. 347
 2. 348
 2. 349
 2. 350
 2. 351
 2. 352
 2. 353
 2. 354
 2. 355
 2. 356
 2. 357
 2. 358
 2. 359
 2. 360
 2. 361
 2. 362
 2. 363
 2. 364
 2. 365
 2. 366
 2. 367
 2. 368
 2. 369
 2. 370
 2. 371
 2. 372
 2. 373
 2. 374
 2. 375
 2. 376
 2. 377
 2. 378
 2. 379
 2. 380
 2. 381
 2. 382
 2. 383
 2. 384
 2. 385
 2. 386
 2. 387
 2. 388
 2. 389
 2. 390
 2. 391
 2. 392
 2. 393
 2. 394
 2. 395
 2. 396
 2. 397
 2. 398
 2. 399
 2. 400
 2. 401
 2. 402
 2. 403
 2. 404
 2. 405

```

C      STATIC BATH, AND METHANE TEMPERATURES (DEG F) AS A
C      FUNCTION OF TIME.
C      CALL XEIM (SCO2,SCO,SC2,3,N3,I12,XX(I9),TME)
C      CALL NPLOTS (1,N3,200,0,1,FLWC2,TME,I9,0,0,0,1)
C      CALL NPLOTS (1,N3,200,0,1,HTLANS,TME,I9,0,0,0,1)
C      DO 25 I=1,N3
C          TCH4(I)=TCH4(I)-TEMPO
25      CONTINUE
C          CALL NPLOTS (1,N3,200,0,1,TCH4,TME,I9,0,0,0,1)
C          DO 26 I=1,N3
C              TCH4(I)=TCH4(I)+TEMPO
26      CONTINUE
C          PLCTS=FLOTS+4
27      CONTINUE
C          CALL SECOND (TIMEB)
C          TIMEA=TIMEB-TIMEA

C      CALCULATION OF NITROGEN CONTENT BY DIFFERENCE.
C      CALL SECOND (TIMEI)
C      IF (INHIBIT.EQ.0) WRITE (KW,122) HEATNO(1),HEATNC(2),GRADE1(1),GRA
1 DE1 (2)
C      DO 28 I=1,N3
C          ILINES=I/LINES
C          ILINES=ILINES*LINES
C          IF (I.EQ.ILINES.AND.INHIBIT.EQ.0) WRITE (KW,123)
C          TIME=FLOAT(I-1)*TCYCLE
C          TIME=TIME/60.0
C          FCH4=TCH4(I)-TEMPO
C          NFT=HTLANS(I)+0.001
C          INCHES=(HTLANS(I)-FLOAT(NFT))*12.0+0.51
C          SN2(I)=(100.0-(SCO2(I)+SCO(I)+SO2(I)))*N2FACTR
C          IF (INHIBIT.EQ.0) WRITE (KW,126) I,TIME,SCO2(I),SCO(I),SCH4(I),SO2
1(I),SN2(I),FLW02(I),NFT,INCHES,FCH4
28      CONTINUE
C          CALL SECOND (TIMEH)
C          TIME15=TIMEH-TIMEI

C      CALCULATE MAXIMUM AND MINIMUM GAS CCONTENTS, METHANE TEMPERATURES,
C      OXYGEN FLOW, AND LANCE HEIGHT.
C      CALL SECOND (TIMEA)
C      CO2MAX=SCO2(1)
C      CO2MIN=SCO2(1)
C      COMAX=SCO(1)
C      COMIN=SCO(1)
C      CH4MAX=SCH4(1)
C      CH4MIN=SCH4(1)
C      O2MAX=SO2(1)
C      O2MIN=SO2(1)
C      N2MAX=SN2(1)
C      N2MIN=SN2(1)
C      TCH4MAX=TCH4(1)
C      TCH4MIN=TCH4(1)
C      FLWMAX=FLW02(1)
C      FLWMIN=FLW02(1)
C      HTLMAX=HTLANS(1)
C      HTLMIN=HTLANS(1)

```

```

DO 29 I=1,N3
TIME=FLOAT(I-1)*TCYCLE
CO2MAX=AMAX1(CC2MAX,SCO2(I))
CO2MIN=AMIN1(CO2MIN,SCO2(I))
CCMAX=AMAX1(COMAX,SCO(I))
COMIN=AMIN1(COMIN,SCO(I))
CH4MAX=AMAX1(CH4MAX,SCH4(I))
CH4MIN=AMIN1(CH4MIN,SCH4(I))
O2MAX=AMAX1(O2MAX,SC2(I))
O2MIN=AMIN1(O2MIN,SC2(I))
TCH4MAX=AMAX1(TCH4MAX,TCH4(I))
TCH4MIN=AMIN1(TCH4MIN,TCH4(I))
FLWMAX=AMAX1(FLWMAX,FLWC2(I))
FLWMIN=AMIN1(FLWMIN,FLWC2(I))
HTLMAX=AMAX1(HTLMAX,HTLANS(I))
HTLMIN=AMIN1(HTLMIN,HTLANS(I))
N2MAX=AMAX1(N2MAX,SN2(I))
N2MIN=AMIN1(N2MIN,SN2(I))
29 CONTINUE
CALL SECOND (TIMEB)
TIME5=TIMEB-TIMEA
C
C PRINT CUT TABLE OF MAXIMUMS AND MINIMUMS.
IF (INHIBIT.LE.1) WRITE (KW,127) TIME5
IF (INHIBIT.LE.1) WRITE (KW,128) CO2MAX,COMAX,CH4MAX,O2MAX,N2MAX,C
102MIN,COMIN,CH4MIN,C2MIN,N2MIN
IF (INHIBIT.LE.1) WRITE (KW,129)
C
C THE NEXT 4 LINES CONVERT LANCE HT. IN FEET TO FEET AND INCHES.
MN1=HTLMAX+0.001
MN2=(HTLMAX-FLOAT(MN1))*12.0
MN3=HTLMIN+0.001
MN4=(HTLMIN-FLOAT(MN3))*12.0
TCH4MAX=TCH4MAX-TEMPO
TCH4MIN=TCH4MIN-TEMPO
IF (INHIBIT.LE.1) WRITE (KW,130) FLWMAX,MN1,MN2,TCH4MAX,FLWMIN,MN3
1,MN4,TCH4MIN
C
C CALCULATION OF CARBCN CONTENT OF HOT METAL ASSUMING SATURATION.
HMC=1.34+0.0014*HMTEMP-0.3*HMSI
VHMC=(0.0014**2)*VHTEMP+(0.3**2)*VHMSI
SCRAPC=0.10
SCRFCLB=SCRAPC*SCRAF/100.0
VSCPCLB=(SCRAPC/100.0)**2*VSCRAPC+(SCRAPC/100.0)**2*VSCRAF
HMCLB=HMC*HMWT/100.0
VHMCLB=(HMWT/100.0)**2*VHMC+(HMC/100.0)**2*VHMWT
HMCLP=HMCLB+SCPCLB
VHMCLB=VHMCLB+VSCPCLB
IF (INHIBIT.LE.2) WRITE (KW,131) HEATNO(1),HEATNC(2),GRACE1(1),GRA
10E1 (2)
C
DELTA2=(TCYCLE/60.0)**2
O2BYN2=(AIR02/AIRN2)*N2FACTR
OCH4AMT=0.0

```

```

QUNLSD=0.0
QTOTAIR=0.0
OTRLC2=0.0
QTOTXS=0.0
QCEVN=0.0
CALL SECOND (TIMEA)
IF (ADJFLW.EQ.1) ROTFLW=ROTFLW*ADJUST

DO 38 DOES CALCULATIONS FOR THE WHOLE HEAT. DO 50 (PROGRAM) CALCULATES INFORMATION AFTER THE DATUM POINT UPON THE SAME DATA CALCULATED HERE, E.G. CEV(J), ETC ARE IN DO 108.
DO 38 J=1,N3
DIVIDE=FLOAT(J)
MULTPLY=FLOAT(J-1)**2
DELMULT=DELTA2*MULTPLY
JLINES=J/LINES
JLINES=JLINES*LINES
IF (J.EQ.JLINES.AND.INHIBIT.LE.2) WRITE (KW,132)
CO2=SCO2(J)
CO=SCO(J)
O2=S02(J)
CH4=SC14(J)
N2=SN2(J)
VN2=(VCO2+VCO+VCO2)*N2FACTR**2
CN2=ABC*SQRT(VN2)
PN2=(CN2*100.0)/N2
CH4T=TCH4(J)
TME(J)=FLOAT(J-1)*TCYCLE/60.0
IF (FCHANGE.EQ.1.AND.FTIME.GE.TME(J)) GO TO 30
GC TC 31
ROTFLW=ROTFLW2
P2=F1+P4
CONTINUE
INJRATE=ROTFLW*NATGAS
VINJRAT=(NATGAS**2)*VROTFLW+(ROTFLW**2)*VNATGAS
SF=SORT((T1*P2)/(CH4T*P1))
VSF=(T1*P2)/(4.0*P1*CH4T)*(VCH4T/CH4T**2+VP3/P2**2)
FLWC4=INJRATE*SF
VFLWCH4=(SF**2)*VINJRAT+(INJRATE**2)*VSF
VFLWCH41=VFLWCH4
FLWCH41=FLWCH4
TIME=FLCAT(J-1)*TCYCLE
TIME=TIME/60.0
SFLCW(J)=(FLWCH4/CH4)*100.0
IF (SFLOW(J).GT.FLWTST) SFLOW(J)=FLWTST
VSFLOW=(100.0/CH4)**2*(VFLWCH4+(FLWCH4/CH4)**2*VCH4)
SSFLOW=SQRT(VSFLOW)
CSFLOW=ABC*SSFLOW
PSFLOW=(CSFLOW/SFL0W(J))*100.0
AIRFLW=N2*SFLOW(J)/AIRN2
VAIRFLW=(SFLOW(J)**2*VN2+N2**2*VSFLOW)/(AIRN2**2)
VARFLW1=VAIRFLW
AIRFLW1=AIRFLW
SAO2=AIRO2*AIRFLW/100.0
VSAO2=(AIRO2/100.0)**2*VAIRFLW
UNUSD02(J)=(O2+0.5*CO2)*SFL0W(J)/100.0-SA02

```

```

IF (FLW02(J).LT.TFLW02) UNUSDC2(J)=0.0
DUMMY1=C02*(0.5+O2BYN2)+CO*(O2BYN2)+O2*(1.0+C2BYN2)-100.0*C2BYN2
VDUMMY1=(0.5+O2BYN2)**2*VC02+C2BYN2**2*VC0+(1.0+C2BYN2)**2*V02
VUNUSD=(SFLCW(J)**2*VDUMMY1+DUMMY1**2*VSFLOW)/(100.0**2)
VLSDO21=VUNUSD
RL02=((0.5*C02+0.5*CO)*SFL0W(J)/100.0)+UNUSDO2(J)
DUMMY2=(1.0+O2BYN2)*(C02+O2)+(0.5+O2BYN2)*CO-100.0*C2BYN2
VDUMMY2=(1.0+O2BYN2)**2*(VC02+V02)+(0.5+O2BYN2)**2*VC0
VRLC2=(SFL0W(J)**2*VDUMMY2+DUMMY2**2*VSFLOW)/(100.0**2)
VRLC21=VRL02
RLC21=RL02
IF (FLW02(J).EQ.0.0) GO TO 32
PRLC2(J)=RLC2*100.0/FLWC2(J)
VPRLO2=(100.0/FLW02(J))**2*(VRL02+(RL02/FLW02(J))**2*VFLW02)
SPRL02=SQRT(VPRLO2)
CPRL02=ABC*SPRL02
PFRL02=(CPRL02*100.0)/AES(PRLC2(J))
GO TO 33
32 UNUSDC2(J)=0.0
PRL02(J)=0.0
33 CCNTINUE
TH02=(C02/2.0+C0/2.0)*SFLOW(J)/100.0
VTHC2=(0.005**2)*(C02+CO)**2*VSFL0K+SFLOW(J)**2*(VC02+VC0))
THAIR=(TH02/AIRO2)*100.0
VTHAIR=(100.0/AIRO2)**2*VTH02
XSAIR=(SA02/AIRO2)*100.0-TH02
VXSAIR=(SFLCW(J)**2*(1.005**2*(VC02+VC0)+V02)+(100.0-1.005*(CC2+CC
1)-O2)**2*VSFLOW)/(AIRN2**2)
VXSAIR1=VXSAIR
XSAIR1=XSAIR
PXSAIR(J)=(XSAIR/THAIR)*100.0
IF (FXSAIR(J).GE.XTEST) PXSAIR(J)=XTEST
CEV(J)=(C02+CO)*SFLCW(J)*B1
VCEV(,)=(B1**2)*(SFL0W(J)**2*(VC02+VC0)+(C02+CO)**2*VSFLOW)
SCEV=SQRT(VCEV(J))
CCEV=ABC*SCEV
PCEV=(CCEV*100.0)/CEV(J)
VFLW021=VFLW02
IF (FLW02(J).LT.TFLW02) GO TO 34
GO TO 35
34 VFLW021=0.0
VUSD021=0.0
VARFLW1=0.0
VXSAIR1=0.0
AIRFLW1=0.0
UNUSDO2(J)=0.0
XSAIR1=0.0
VPRLO2=0.0
CPRL02=0.0
PFRL02=0.0
VRLC21=(0.005**2)*(SFL0W(J)**2*(VC02+VC0)+(CC2+CC)**2*VSFL0W)
RL021=0.005*(C02+CO)*SFL0W(J)
CCNTINUE
IF (J.EQ.1) GO TO 36
C
C
C INTEGRATION OF VARIABLES WITH TIME (TRAPEZOIDAL RULE).
CH4AMT=CH4AMT+((FLWCH41+FLWCH42)/2.0)*TCYCLE/60.0

```

```

QCH4AM T=QCH4AMT+VFLCH41
02UNUSD=02UNUSD+((UNUSD02(J)+LNUSD02(J-1))/2.0)*TCYCLE/60.
QUNUSD=QUNUSD+VUSD021
TOTAIR=TOTAIR+((AIRFLW1+AIRFLW2)/2.0)*TCYCLE/60.0
QTOTAIR=QTOTAIR+VARFLW1
TOTRLC2=TOTRL02+((RL021+RL022)/2.0)*TCYCLE/60.0
QTRL02=QTRL02+VRL021
TCTFLW=TCTFLW+((FLWC2(J)+FLW02(J-1))/2.0)*TCYCLE/60.0
TOTXYS=TOTXYS+((XSAIR1+XSAIR2)/2.0)*TCYCLE/60.0
QTOTXYS=QTOTXYS+VXSAIR1
CEVD=C EVD+((CEV(J)+CEV(J-1))/2.0)*TCYCLE/60.0
QCEVD=QCEVD+VCEV(J)
VCEVD=(QCEVD/DIVIDE)*DELMULT
FLWCH42=FLWCH41
AIRFLW2=AIRFLW1
RL022=RL021
XSAIR2=XSAIR1
VFLCH42=VFLCH41
VUSD022=VUSD021
VFLWC22=VFLW021
VARFLW2=VARFLW1
VRLC22=VRL021
VXSAIR2=VXSAIR1

```

CCC

FOLLOWING VALUES ARE INITIALIZED FOR FIRST PASS THRU DO LCCP.

36

CH4 AMT=0.0
O2UNUSD=0.0
TOTAIR=0.0
TCTRLO2=0.0
TOTFLW=0.0
TOTXS=0.0
CEVD=0.0
VCEVD=0.0
VCH4AMT=0.0
VUNUSD=0.0
VTOTAIR=0.0
VTRL02=0.0
VTOTFLW=0.0
VTOTXS=0.0
FLWCH42=FLWCH
AIRFLW2=AIRFL
RLC22=RL021
XSAIP2=XSAIR1
VFLCH42=VFLCH
VUSD022=VUSD0
VARFLW2=VARFL
VRL022=VRL021
VFLW022=VFLW0
VXSAIR2=VXSAI
CONTTNU

37
C C

SCE VD=SQRT (VCE VD)
 CCE VD=ABC * SCE VD
 PCE VD=(CCE VD * 100.0) / CE VD

```

BATHCC=HMCLC-EVDO
VBATHCC=VHMCCLB+VCEVD
ECC(J)=(BATHCC/TOTWT)*100.0
VBCCJ=(100.0/TOTWT)**2*(VBATHCC+(BATHCC/TOTWT)**2*VTOTWT)
SBCCJ=SQRT(VBCCJ)
CBCCJ=ABC*SBCCJ
BOUNDS(J)=CBCCJ
PBCCJ=(CBCCJ*100.0)/BCC(J)
IF (J.GT.2) SDR(J-1)=ECC(J)-ECC(J-2)/(2.0*TCYCLE*FLW02(J-1))
1)
IF (FLW02(J-1).EQ.0.0) SDR(J-1)=0.0
IF (INHIBIT.LE.2) WRITE(KW,133) TIME(J),SFLOW(J),CSFLOW,FRL02(J),C
1PRL02,C2,CEV(J),CCEV,CEVC,CCEVD,ECC(J),CBCCJ,PBCCJ
CONTINUE
SDR(1)=0.0
SDR(N3)=SDR(N3-1)
CALL SECOND(TIMEB)

38
C
TIME7=TIMEB-TIMEA
NSUM=N30+N32-N7-NN11-1
DEL_MUL1=DELTAD2*FLOAT(NSUM)**2/FLOAT(NSUM+1)
DEL_MUL2=DELTAD2*FLOAT(NSUM)**2
VCH4AMT=(QCH4AMT/DIVIDE)*DELMULT
SCH4AMT=SQRT(VCH4AMT)
CCH4AMT=ABC*SCH4AMT
PCH4AMT=(CCH4AMT*100.0)/CH4AMT
VTOTFLW=VFLLW02*DELMUL2
STOTFLW=SQRT(VTOTFLW)
CTOTFLW=ABC*STOTFLW
PTOTFLW=(CTOTFLW*100.0)/TOTFLW
VTRL02=(QTRL02/DIVIDE)*DELMULT
STRLO2=SQRT(VTRL02)
CTRL02=ABC*STRLO2
PTRL02=(CTRL02*100.0)/TCTRL02
VUNUSD=QUNUSD*DELMUL1
SLNUSD=SQRT(VUNUSD)
CUNUSD=ABC*SUNUSD
PLNUUSD=(CUNUSD*100.0)/AES(02UNUSD)
VTOTAIR=QTOTAIR*DELMUL1
STOTAIR=SQRT(VTOTAIR)
CTOTAIR=ABC*STOTAIR
PTOTAIR=(CTOTAIR*100.0)/TOTAIR
VTOTXS=QTOTXS*DELMUL1
STOTXS=SQRT(VTOTXS)
CTOTXS=ABC*STOTXS
PTOTXS=(CTOTXS*100.0)/TOTXS
SCEVD=SORT(VCEVD)
CCEVD=ABC*SCEVD
PCEVD=(CCEVD*100.0)/CEVD
IF (INHIBIT.LE.2) WRITE(KW,134) HEATNO(1),HEATNC(2),GRACE1(1),GRA
1DE1(2)
IF (INHIBIT.LE.2) WRITE(KW,138) TCTFLW,CTOTFLW,FTOTFLW,TCTRL02,CT
1RL02,PTRL02,02UNUSD,CLNUUSD,PUNUSD,TCTAIR,CTOTAIR,PTOTAIR,TCTXS,CTC
2TXS,FTOTXS,CEVD,CCEVD,PCEVD,CH4AMT,CCH4AMT,PCH4AMT
CALL SECOND(TIMEA)
IF (NOLOTS.LE.0) GO TO 39
GO TO 43

```

```

39 CONTINUE
CALL NPLOTS (1,N3,200,0,1,SDR,BCC,I9,0,0.0,1)
CALL NPLOTS (1,N3,200,0,1,SFLCW,TME,I9,0,0.0,1)
CALL NPLOTS (1,N3,200,0,1,PRLC2,TME,I9,0,0.0,1)
CALL NPLOTS (1,N3,200,0,1,UNUSD02,TME,I9,0,0.0,1)
CALL NPLOTS (1,N3,200,0,1,CEV,TME,I9,0,0.0,1)
N88=N3-N30
DO 40 J=1,N88
DUMIE2(J)=(TME(N30+J-1)-TME(N30))*60.0
DUMIE1(J)=CEV(N30+J-1)
CONTINUE
40 CALL NPLOTS (1,N88,200,0,1,DUMIE1,DUMIE2,I9,0,0.0,1)
DO 41 J=1,N3
LCWER(J)=BCC(J)-BOUNDS(J)
IF (LCWER(J).LT.0.0) LOWER(J)=0.0
UPPER(J)=BCC(J)+BOUNDS(J)
DUMIE3(J)=VCEV(J)
41 CONTINUE
CALL XDJM (UPPER,BCC,LOWER,3,N3,I9,XX(I1),TME)
DO 42 J=1,N3
VCEV(J)=DUMIE3(J)
42 CONTINUE
PLCTS=PLOTS+7
43 CONTINUE
CALL SECOND (TIMEB)
TIME8=TIMEB-TIMEA
TOLY=TME(N7+NN11+1)

C
C DO 110 CONTAINS CALCULATIONS FOR DATA COLLECTED AFTER DATUM
C PCINT, PLUS ALL FITTING AND PREDICTIONS.
DO 110 I=1,3
IF (I.LT.3) GO TO 110
IF (I-1) 45,44,45
44 N5=N30
N4=N24
GO TO 51
45 IF (I-2) 47,46,47
46 CONTINUE
N5=N32
N4=N3
GO TO 51
47 N4=N3-N7
N5=N32-N7
HMCLB=DATMPT*TOTWT/100.0
VHMC LB=((1.0/100.0)**2)*((TOTWT**2)*VDATMPT+(DATMPT**2)*VTCTWT)
KJI1=3
IF (INHIBIT.LE.3) WRITE (KW,136) HEATNO(1),HEATNC(2),GRADE1(1),GRA
1DE1(2)
CALL SECOND (TIMEI)
QCEVD=0.0

C DO 50 CALCULATES DATA AFTER DATUM PCINT.
DO 50 J=1,N4
DIVIDE=FLOAT(J)
MULTPLY=FLOAT(J-1)**2
DELMULT=DELTA2*MULTPLY

```

```

N6=N7+J
JLINES=J/LINES
JLINES=JLINES*LINES
IF (J.EQ.JLINES.AND.INHIBIT.LE.3) WRITE (KW,135)
FLWC2(J)=FLW02(N6)
CEV(J)=CEV(N6)
VCEV(J)=VCEV(N6)
SCEV=SQRT(VCEV(J))
CCEV=ABC*SCEV
PCEV=(CCEV*100.0)/CEV(J)
TME(J)=TME(N6)
IF (J.EQ.1) GO TO 48
CEVD=CEVD+((CEV(J)+CEV(J-1))/2.0)*TCYCLE/60.0
QCEVD=QCEVD+VCEV(J)
VCEVD=(QCEVD/DIVIDE)*DELMULT
GO TO 49
48 CEVD=0.0
VCEVD=0.0
CONTINUE
SCEVD=SORT(VCEVD)
CCEVD=ABC*SCEVD
PCEVD=(CCEVD*100.0)/CEVD
IF (J.EQ.1) PCEVD=0.0
BATHCC=HMCLB-CEVD
VBATHCC=VHMCLB+VCEVD
SBATHCC=SQRT(VBATHCC)
CBATHCC=ABC*SBATHCC
PBATHCC=(CBATHCC*100.0)/BATHCC
BCC(J)=(BATHCC/TOTWT)*100.0
BCC(N6)=BCC(J)
VBCCJ=(100.0/TOTWT)**2*(VBATHCC+(BATHCC/TOTWT)**2*VTOTWT)
DUMIE3(J)=1.0/VBCCJ
SECCJ=SQRT(VBCCJ)
CBCCJ=ABC*SBCCJ
IF (J.GT.2) SDR(J-1)=60.0*(BCC(J)-BCC(J-2))/(2.0*TCYCLE*FLW02(J-1))
1) LOWER(J)=BCC(J)-CBCCJ
IF (LOWER(J).LT.0.0) LOWER(J)=0.0
UPPER(J)=BCC(J)+CBCCJ
LOWER(N6)=LOWER(J)
UPPER(N6)=UPPER(J)
SDR(N6)=SDR(J)
PBCCJ=(CBCCJ*100.0)/BCC(J)
IF (INHIBIT.LE.3) WRITE (KW,137) TME(J),CEV(J),CCEV,PCEV,CEVD,CCEV
1D,PCEVD,BCC(J),CBCCJ,PBCCJ
CONTINUE
CALL SECOND (TIMEH)
TIME14=TIMEH-TIMEI
CONTINUE
CALL SECOND (TIMED)
DO 52 J=1,N4
TME(J)=(TME(J)-TME(N7+1))*60.0
CONTINUE
IF (NCPLOTS.LE.0) CALL XDIM (UPPER,BCC,LOWER,3,N4,I9,XX(I1),TME)
DO 53 J=1,N4
TME(J)=TME(J)/60.0+TME(N7+1)
CONTINUE
CALL SECOND (TIMEE)

```

2.	812
22.	813
222.	814
2222.	815
22222.	816
222222.	817
2222222.	818
22222222.	819
222222222.	820
2222222222.	821
22222222222.	822
222222222223.	823
222222222224.	824
222222222225.	825
222222222226.	826
222222222227.	827
222222222228.	828
222222222229.	829
222222222230.	830
222222222231.	831
222222222232.	832
222222222233.	833
222222222234.	834
222222222235.	835
222222222236.	836
222222222237.	837
222222222238.	838
222222222239.	839
222222222240.	840
222222222241.	841
222222222242.	842
222222222243.	843
222222222244.	844
222222222245.	845
222222222246.	846
222222222247.	847
222222222248.	848
222222222249.	849
222222222250.	850
222222222251.	851
222222222252.	852
222222222253.	853
222222222254.	854
222222222255.	855
222222222256.	856
222222222257.	857
222222222258.	858
222222222259.	859
222222222260.	860
222222222261.	861
222222222262.	862
222222222263.	863
222222222264.	864
222222222265.	865
222222222266.	866
222222222267.	867
222222222268.	868
222222222269.	869

```

TIME11=TIME11+TIMEE-TIMED
IF (NCPLOTS.LE.0) PLOTS=PLOTS+2
DO 54 L=1,N5
TIME=FLCAT(L-1)*TCYCLE
CONTINUE
TIME=TIME-FLOAT(START)
ISTART=TIME
IF (I.EQ.3) ISTART=FLCAT(N7+NN11)*TCYCLE
JDO=0
TOTTIME=FLOAT(N5)*TCYCLE
IF ((I.EQ.3).AND.(TOTTIME.GT.(MAXLAG+10.0))) GO TO 55
GO TO 56
55 NSTART=IFIX(MAXLAG+PRESTRT)
ISTART=IFIX(TME(N5)*60.0-FLOAT(START))
JDO=1
CONTINUE
M1=1
M2=N4
M3=1
KJI2=KJI1/NTCYCLE
CALL SECOND (TIMEF)

CCCCC FITTING STARTS HERE AT DO 108. A MAXIMUM OF 10 MODELS CAN BE
      ATTEMPTED. THE PROGRAM IS PRESENTLY ONLY DOING ONE MODEL -- IT
      CAN BE READILY ENLARGED TO HANDLE ANY NO. UP TO 10 AND CHOOSE
      THE BEST ONE.
DO 108 J=M1,M2,M3
TIME=FLOAT(J-1)*TCYCLE
IF (I.EQ.3) TIME=FLCAT(J-0+N7)*TCYCLE
IF (NOFITS.EQ.1) GO TO 109
JSTART=TIME
IF (JSTART.GE.ISTART) GO TO 57
GO TO 108
57 CONTINUE
JJ4=J/KJI2
JJ4=JJ4*KJI2
IF (J.EQ.JJ4) GO TO 58
GO TO 107
58 CONTINUE
NN5=J-IFIX(FLOAT(ISTART+1)/TCYCLE)+1
IF (I.EQ.3.AND.JDO.EQ.0) GO TO 59
IF (I.NE.3) GO TO 60
NN5=IFIX((TME(J)*60.0-FLOAT(ISTART+1))/TCYCLE)+1
GO TO 60
59 NN5=J-NN11+1
IF (NN5.LE.0) GO TO 108
60 CONTINUE
NN9=NN5
IF (NN5.GE.N/2) STOP ?
NN6=IFIX(FLOAT(NTIME)/TCYCLE)
NN7=0
IF (NN6.EQ.0) GO TO 62
IF (NN5.GE.NN6) GO TO 61
GO TO 62
61 NN5=NN6
NN7=NN9-NN6
62 CONTINUE

```

```

NN10=0
IF ((I.EQ.3).AND.((NN6.EQ.0).AND.(NN7.EQ.0))) NN10=10
NN8=NN5
IF ((J+NN10).GT.NN5) GC TO 108
DO E2 JK=1,NN5
J2=JK-1
J3=J2+IFIX(FLOAT(ISTART+1)/TCYCLE)+NN7+NN10
TME(JK)=TME(J3)
BCC(JK)=BCC(J3)
LOWER(JK)=LCWER(J3)
UPPER(JK)=UPPER(J3)
SDR(JK)=SDR(J3)
CONTINUE
IF (J.GT.NN5) GO TO 64
GC TO 65
IF (J.GT.(NN5+NN10)) GC TO 66
GC TC 108
CONTINUE
NOBMAX=NN5+1
IF (NOBMAX.LE.3) GO TC 107
CALL SECOND (TIMEA)
MPLCTD=0
IF (I.GT.1) GO TO 67
CONTINUE
M=1
MCDEL=4
NVARX=2
GO TC 68
M=2
MCDEL=1
NVARX=3
CONTINUE
JJ1=MCDEL
IF (NW8TS.EQ.1) GO TO 71
CALL LESQ (EQU,B,TME,ECC,M,NN5)
DO 69 KIL=1,NVARX
BI(KIL)=0.0
CONTINUE
CALL EQUATN (B,EQU,NN5,TME,JJ1)
SSQ=0.0
DO 70 JKL=1,NN5
RESID=EQU(JKL)-BCC(JKL)
SSG=SSQ+RESID*RESID
CONTINUE
GO TC 74
CONTINUE
IF (M.GE.(NN5-1)) GC TO 107
CALL CRISQ (TME,BCC,DLMIE3,NN5,M,EQU,SSQ,BI)
DC 72 LI=1,NN5
EQU(LI)=EQU(LI)+BCC(LI)
CONTINUE
DC 73 JL=1,NVARX
LJ=NVARX-JL+1
B(JL)=BI(LJ)
BI(LJ)=0.0
CONTINUE
CONTINUE
A30=TME(N5)-(FLOAT(KJI1)*1.001)/E0.0

```

```

CALL SECOND (TIMEB)
TIME13=TIME13+TIMEB-TIMEA
CALL SECOND (TIMED)
IF (TME(J+NN10).GT.A30.AND.NOFLOTS.LE.0) GO TO 75
GO TO 81
IPILOT=200
NF=4
NT=NN5
NPT=1
FMTE=0.0
NDO=1
IPOINT=1
NREAD=1
DO 76 JI=1,7
YLVERT(JI)=BLANK10
YRVERT(JI)=BLANK10
XLHOR(JI)=BLANK10
XUFCR(JI)=BLANK10
CONTINUE
76
J6=I12
I25=6+2*IPILOT
DO 77 JL=1,NT
LJ=LJ-1
XX(I25+LJ)=TME(JL)
XX(I25+LJ+NT)=BCC(JL)
XX(I25+LJ+2*NT)=EQU(JL)
XX(I25+LJ+3*NT)=LOWER(JL)
XX(I25+LJ+4*NT)=UPPER(JL)
CONTINUE
77
WRITE (KW,139)
DO 78 LJ=1,NT
RESID=BCC(LJ)-EQU(LJ)
WRITE (KW,140) TME(LJ),BCC(LJ),EQU(LJ),RESID
CONTINUE
78
DO 79 LJ=1,NN5
JL=LJ-1
XX(I25+JL)=(XX(I25+JL)-TDLY)*60.0
CONTINUE
79
CALL NPLOT1 (NF,NT,IPILOT,IPOINT,NREAD,J6,1)
CALL EQUATN (DUD1,DUMIE1,NT,TME,JJ1)
CALL EQUATN (DUD2,DUMIE2,NT,TME,JJ1)
CALL XDIM (DUMIE1,DUMIE2,BCC,3,NT,I9,XX(I2),XX(I25))
DO 80 LJ=1,NN5
JL=LJ-1
XX(I25+JL)=XX(I25+JL)/60.0+TDLY
CONTINUE
80
CALL NPLOTS (1,NN5,200,0,1,SDR,BCC,I9,0,0.0,1)
PLCTS=PLOTS+3
81
CONTINUE
CALL SECOND (TIMEE)
TIME9=TIME9+TIMEE-TIMED
GO TO (85,86,87,88,89,90,82,83,84,150), MODEL
82
MODEL(1)=MODEL7(1)
MCOL(2)=MODEL7(2)
MODEL(3)=MODEL7(3)
GO TO 91
83
MODEL(1)=MODEL8(1)
MCOL(2)=MODEL8(2)

```

2. 986
2. 987
2. 988
2. 989
2. 990
2. 991
2. 992
2. 993
2. 994
2. 995
2. 996
2. 997
2. 998
2. 999
2. 1000
2. 1001
2. 1002
2. 1003
2. 1004
2. 1005
2. 1006
2. 1007
2. 1008
2. 1009
2. 1010
2. 1011
2. 1012
2. 1013
2. 1014
2. 1015
2. 1016
2. 1017
2. 1018
2. 1019
2. 1020
2. 1021
2. 1022
2. 1023
2. 1024
2. 1025
2. 1026
2. 1027
2. 1028
2. 1029
2. 1030
2. 1031
2. 1032
2. 1033
2. 1034
2. 1035
2. 1036
2. 1037
2. 1038
2. 1039
2. 1040
2. 1041
2. 1042
2. 1043

	MCDL(3)=MODEL8(3)	2.1044
	GO TO 91	2.1045
84	MCDL(1)=MODEL9(1)	2.1046
	MODL(2)=MODEL9(2)	2.1047
	MODL(3)=MODEL9(3)	2.1048
	GO TO 91	2.1049
150	MCDL(1)=MODEL10(1)	2.1050
	MCDL(2)=MODEL10(2)	2.1051
	MODL(3)=MODEL10(3)	2.1052
	GO TO 91	2.1053
85	MODL(1)=MODEL1(1)	2.1054
	MCDL(2)=MODEL1(2)	2.1055
	MCDL(3)=MODEL1(3)	2.1056
	GO TO 91	2.1057
86	MCDL(1)=MODEL2(1)	2.1058
	MCDL(2)=MODEL2(2)	2.1059
	MODL(3)=MODEL2(3)	2.1060
	GO TO 91	2.1061
87	MCDL(1)=MODEL3(1)	2.1062
	MCDL(2)=MODEL3(2)	2.1063
	MCDL(3)=MODEL3(3)	2.1064
	GO TO 91	2.1065
88	MODL(1)=MODEL4(1)	2.1066
	MCDL(2)=MODEL4(2)	2.1067
	MODL(3)=MODEL4(3)	2.1068
	GO TO 91	2.1069
89	MODL(1)=MODEL5(1)	2.1070
	MCDL(2)=MODEL5(2)	2.1071
	MODL(3)=MODFL5(3)	2.1072
	GO TO 91	2.1073
90	MODL(1)=MODEL6(1)	2.1074
	MCDL(2)=MODEL6(2)	2.1075
	MODL(3)=MODEL6(3)	2.1076
91	CONTINUE	2.1077
	GO TO (96,92,93,94,95), NVARX	2.1078
92	WRITE (KW,141) HEATNO(1),HEATNO(2),MODEL,TME(J+NN10),GRADE1(1),GRA 1DE1(2),MODL(1),MODL(2),MODL(3),COEFFN(1),B(1),BI(1),COEFFN(2),E(2) 2,BI(2)	2.1079
	GO TO 96	2.1080
93	WRITE (KW,141) HEATNO(1),HEATNO(2),MODEL,TME(J+NN10),GRADE1(1),GRA 1DE1(2),MODL(1),MODL(2),MODL(3),COEFFN(1),B(1),BI(1),COEFFN(2),E(2) 2,BI(2),COEFFN(3),B(3),BI(3)	2.1081
	GO TO 96	2.1082
94	WRITE (KW,141) HEATNO(1),HEATNO(2),MODEL,TME(J+NN10),GRADE1(1),GRA 1DE1(2),MODL(1),MODL(2),MODL(3),COEFFN(1),B(1),BI(1),COEFFN(2),E(2) 2,BI(2),COEFFN(3),B(3),BI(3),CCEFFN(4),B(4),BI(4)	2.1083
	GO TO 96	2.1084
95	WRITE (KW,141) HEATNO(1),HEATNO(2),MODEL,TME(J+NN10),GRADE1(1),GRA 1DE1(2),MODL(1),MODL(2),MODL(3),COEFFN(1),B(1),BI(1),COEFFN(2),E(2) 2,BI(2),COEFFN(3),B(3),BI(3),CCEFFN(4),B(4),BI(4),COEFFN(5),B(5),BI 3,(5)	2.1085
	CONTINUE	2.1086
	TIMEC=TIMEB-TIMEA	2.1087
	WRITE (KW,142) SSQ, TIMEC, NN8	2.1088
	CALL SECOND (TIMEA)	2.1089
	WRITE (KW,143)	2.1090
	GO TO (97,98,98), I	2.1100
97	CALL ITER8 (B,DATMPT,EY,NVARX,MODEL,TME(J),NCITER,TPREC,0)	2.1101

```

TLEFT1=(TPRED-TME(J))*60.0          2.1102
TLEFT2=(TME(N5)-TME(J))*60.0          2.1103
CALL ITER8 (B,DATMPT,BY,NVARX,MODEL,DUMMY,NONE,TME(N5),1) 2.1104
WRITE (KW,144) DATMPT,TLEFT1,TLEFT2,BCC(N5),CATMFT,BY,BCC(N5) 2.1105
GC TO 107                            2.1106
98  CONTINUE                           2.1107
DEL TAC=BCC(N5)-BCC(N4)             2.1108
O2SPCC=DEL TAC+SPECC              2.1109
CALL ITER8 (B,O2SPCC,EY,NVARX,MODEL,TME(J+NN10),NOITER,TPRED1,0) 2.1110
O2OFFC=STALC+DEL TAC              2.1111
CALL ITER8 (B,O2OFFC,EY,NVARX,MODEL,TME(J+NN10),NOITER,TFRED2,0) 2.1112
TLEFT1=(TPRED1-TME(J+NN10))*60.0-MAXLAG 2.1113
TLEFT2=(TME(N5)-TME(J+NN10))*60.0-MAXLAG 2.1114
TLEFT3=(TPRED2-TME(J+NN10))*60.0-MAXLAG 2.1115
DUDTST=0.5*TCYCLE                  2.1116
IF (ABS(TLEFT2-PTIME1).GE.DUDTST) GC TO 100 2.1117
TIMLFT1=TLEFT2                      2.1118
DO 99 LI=1,NVARX                     2.1119
DUD1(LI)=B(LI)                      2.1120
99  CONTINUE                           2.1121
100 CONTINUE                           2.1122
IF (ABS(TLEFT2-PTIME2).GE.DUDTST) GC TO 102 2.1123
TIMLFT2=TLEFT2                      2.1124
DO 101 LI=1,NVARX                     2.1125
DUD2(LI)=B(LI)                      2.1126
101 CONTINUE                           2.1127
102 CONTINUE                           2.1128
CALL ITER8 (B,SPECC,BY,NVARX,MODEL,DUMMY,NONE,TME(N5),1) 2.1129
GO TO (103,104,105), I               2.1130
103 STOP 3                            2.1131
104 WRITE (KW,145) SPECC,STALC        2.1132
105 WRITE (KW,146) SPECC,STALC        2.1133
106 CONTINUE                           2.1134
107 WRITE (KW,147) TLEFT1,TLEFT3,TLEFT2,BCC(N5),STALC,BY,BCC(N5) 2.1135
108 CALL SECOND (TIMEB)                2.1136
TIME16=TIME16+TIMEB-TIMEA            2.1137
107 CONTINUE                           2.1138
108 CONTINUE                           2.1139
C                                     2.1140
C                                     2.1141
CALL SECOND (TIMEG)                 2.1142
TIME10=TIME10+TIMEG-TIMEF            2.1143
109 CONTINUE                           2.1144
110 CONTINUE                           2.1145
C                                     2.1146
C                                     2.1147
TIME12=TIME6+TIME8+TIME9+TIME11    2.1148
IF (TIME12.LE.0.5) TIME12=0.0        2.1149
AVGTIME=TIME12/FLOAT(FLCTS)          2.1150
IF (FLOTS.LE.3) PLOTS=0              2.1151
WRITE (KW,148) PTIME1,TIMLFT1,PTIME2,TIMLFT2 2.1152
IF (NCPLOTS.EQ.1) READ (KR,149)      2.1153
RETURN                               2.1154
C                                     2.1155
C                                     2.1156
INDEX OF FORMAT STATEMENTS.        2.1157
C                                     2.1158
C                                     2.1159

```

```

111  FORMAT (8F10.4) 2.1160
112  FORMAT (I4,6X,7F10.4) 2.1161
113  FORMAT (20I4) 2.1162
114  FORMAT (A10,6F10.4,I10) 2.1163
115  FORMAT (1H ,60X,*HEAT NC. *,A10//58X,*HOT METAL WEIGHT = *,F6.0/ 2.1164
    1/58X,*SCRAP WEIGHT = *,F6.0//58X,*END PCINT CARBON = *,F6.2/ 2.1165
    2/58X,*HOT METAL SILICCN = *,F6.2//58X,*HOT METAL TEMP. = *,F6.0/ 2.1166
    3/58X,*ORDERED ING. WT. = *,F6.0) 2.1167
116  FORMAT (1H1,22(/)) 2.1168
117  FORMAT (1H1,2A10,29X,*S T A C K   G A S   A N A L Y S I S */50X, 2.1169
    1*-----*/1H ,2A10,29X,*M I L L V C 2.1170
    2 L T*,* R E A D I N G S*/50X,*-----* 2.1171
    3---+//33X,*NO.*6X,*C02*,7X,*CO*,7X,*CH4*,8X,*O2*,6X,*O2FLCW*,2X,* 2.1172
    4LANCE HT.*4X,*TIME*) 2.1173
118  FORMAT (1H1,32X,*NO.*6X,*C02*,7X,*CO*,7X,*CH4*,8X,*O2*,6X,*O2FLOW 2.1174
    1*,2X,*LANCE HT.*4X,*TIME*) 2.1175
119  FORMAT (1H1,*ERROR IN DATA INPUT -- NDATA (DIMENSIONING NO.) IS LE 2.1176
    1SS THAN TO THE NO. OF POINTS TO BE READ (N3)*) 2.1177
120  FORMAT (8F10.4) 2.1178
121  FORMAT (30X,I6,7F10.4) 2.1179
122  FORMAT (1H1,2A10,29X,*S T A C K   G A S   A N A L Y S I S */50X,* 2.1180
    1*-----*/1H ,2A10,26X,*P C T . C 2.1181
    2 O M *,*P O S I T I O N*/47X,*-----* 2.1182
    3/21X,*NO.*6X,*TIME*,7X,*C02*,7X,*CO*,7X,*CH4*,8X,*O2*,8X,*N2*,4X, 2.1183
    4*C02 FLCW*,4X,*LANCE HT.,2X,*CH4 TEMP.*}) 2.1184
123  FORMAT (1H1,20X,*NO.*6X,*TIME*,7X,*C02*,7X,*CO*,7X,*CH4*,8X,*C2*, 2.1185
    18X,*N2*,4X,*O2 FLOW*,4X,*LANCE HT.,2X,*CH4 TEMP.*}) 2.1186
124  FORMAT (1H1) 2.1187
125  FORMAT (14(/)) 2.1188
126  FORMAT (1H ,20X,I3,F10.2,F10.1,F10.2,F10.5,2F10.1,F9.0,6X,I2,2X,I2 2.1189
    1,F10.1) 2.1190
127  FORMAT (1H1,25(/),36X,*T A B L E S   O F   M A X I M U M S   A 2.1191
    1N D   M I N I M U M S*/36X,65(1H-),/51X,*CALCULATION TIME = *,F7. 2.1192
    23,* SECONDS*//49X,*TABLE NO. 1 -- GAS LEVELS (PERCENTAGES*)/) 2.1193
128  FORMAT (1H ,32X,*MAXIMUMS CO2 = *,F4.1,* CO = *,F4.1,* CH4 = *, 2.1194
    1F6.5,* O2 = *,F4.1,* N2 = *,F4.1/33X,*MINIMUMS CO2 = *,F4.1,* 2.1195
    2CO = *,F4.1,* CH4 = *,F6.5,* O2 = *,F4.1,* N2 = *,F4.1//) 2.1196
129  FORMAT (1H ,45X,*TAELE NO. 2 -- O2 FLOW,LANCE HT.,CH4 TEMP.*/) 2.1197
130  FORMAT (1H ,37X,*MAXIMUMS O2 FLOW = *,F5.0,* LANCE HT. = *,I2,I3 2.1198
    1,* CH4 TEMP. = *,F4.1/38X,*MINIMUMS O2 FLOW = *,F5.0,* LANCE HT 2.1199
    2. = *,I2,I3,* CH4 TEMP. = *,F4.1) 2.1200
131  FORMAT (1H1,2A10,31X,*RESULTS -- FRCM START CF BLOW*/52X,29(1H-)// 2.1201
    11X,2A10//11X,*TIME*,4X,*STACK GAS*,4X,*ERROR*,4X,*PRLO2*,5X,*ERROR 2.1202
    2*,5X,*CEV*,6X,*ERROR*,6X,*CEVD*,5X,*ERROR*,4X,*BATH C*,4X,*ERROR* 2.1203
    35X,*ERROR*/10X,*MINUTES FLOW(SCFM)*,3X,*ABS)*,4X,*PCT)*,5X,*(AB 2.1204
    4S)*,3X,*LB/MIN*,5X,*(ABS)*,7X,*LB*,6X,*(AES)*,4X,*PCT)*,5X,*(AB 2.1205
    5*,5X,*PCT)*/) 2.1206
132  FORMAT (1H1,10X,*TIME*,4X,*STACK GAS*,4X,*ERRCR*,4X,*PRLO2*,5X,*ER 2.1207
    1ROR*,5X,*CEV*,6X,*ERRCR*,6X,*CEVD*,5X,*ERROR*,4X,*BATH C*,4X,*ERR 2.1208
    2R*,5X,*DROP*/10X,*MINUTES FLOW(SCFM)*,3X,*ABS)*,4X,*PCT)*,5X,* 2.1209
    3(ABS)*,3X,*LB/MIN*,5X,*(ABS)*,7X,*LB*,6X,*AES)*,4X,*PCT)*,5X,* 2.1210
    4BS)*,5X,*PCT)*/) 2.1211
133  FORMAT (11X,F5.2,5X,F6.0,5X,F4.0,5X,F5.3,5X,F5.4,5X,F5.2) 2.1212
134  FORMAT (1H1,19(/),60X,2A10//59X,2A10//) 2.1213
135  FORMAT (1H1,20X,*TIME*,5X,*CEV*,2(5X,*ERROR*),6X,*CEVD*,6X,*ERROR 2.1214
    1*,5X,*ERROR*,5X,*BATH C*,5X,*ERROR*,5X,*ERROR*/19X,*MINUTES*,3X,*L 2.1215
    2B/MIN *(ABS)*,5X,*REL)*,7X,*LB*,7X,*ABS) (REL)*,6X,*PCT)* 2.1216

```

136 3,5X,*ABS)*,5X,*REL*)*)
 FORMAT (1H1,2A10,34X,*RESULTS -- FROM DATUM POINT*/55X,27(1H-)//1
 1X,2A10//21X,*TIME*,5X,*CEV*,2(5X,*ERROR*),6X,*CEVD*,6X,*ERRCR*,5X
 2,*ERROR*,5X,*BATH C*,5X,*ERROR*,5X,*ERROR*/19X,*MINUTES*,3X,*LB/MI
 3N *(ABS)*,5X,*REL)*,7X,*LB*,7X,*ABS) (REL)*,6X,*PCT)*,5X,
 4*(AES)*,5X,*REL*)*) 2.1218
 2.1219
 2.1220
 2.1221
 2.1222
 2.1223
 2.1224
 2.1225
 2.1226
 2.1227
 2.1228
 2.1229
 2.1230
 2.1231
 2.1232
 2.1233
 2.1234
 2.1235
 2.1236
 2.1237
 2.1238
 2.1239
 2.1240
 2.1241
 2.1242
 2.1243
 2.1244
 2.1245
 2.1246
 2.1247
 2.1248
 2.1249
 2.1250
 2.1251
 2.1252
 2.1253
 2.1254
 2.1255
 2.1256
 2.1257
 2.1258
 2.1259
 2.1260
 2.1261
 2.1262
 2.1263
 2.1264
 2.1265
 2.1266
 2.1267
 2.1268
 2.1269
 2.1270
 2.1271
 2.1272

137 FORMAT (20X,F5.2,4X,F4.0,6X,F5.1,5X,F5.1,5X,F6.1,5X,F5.1,5X,F5.1,5X,
 1X,F5.3,5X,F6.4,5X,F5.2)
 138 FORMAT (58X,*OXYGEN DISTRIBUTION///42X,*INPUT: LANCE OXYGEN = *,
 1F6.0,*CU.FT.[*,F5.0,*](*,F5.1,*)//42X,*OUTPUT: RECOVERED AS*/
 250X,*CC AND CO2 = *,F6.0,*CU.FT.[*,F5.0,*](*,F5.1,*)/50X,*U
 3NUSED O2 = *,F6.0,*CU.FT.[*,F5.0,*](*,F5.1,*)//60X,*MISCE
 4LLANEUS DATA*/42X,*TOTAL AMT. OF AIR*/42X,*DRAWN INTO STACK*,5X,
 5* = *,F7.0,*CU.FT.[*,F6.0,*](*,F3.1,*)//42X,*AMT. OF EXCESS AIR
 6 = *,F7.0,*CU.FT.[*,F6.0,*](*,F3.1,*)//42X,*TOTAL EVCLVED CA
 7RBON = *,F7.0,*LB.[*,F3.0,*](*,F3.1,*)//42X,*TOTAL CH4 USED*,7X
 8* = *,F7.0,*CU.FT.*,F3.1,*)
 139 FORMAT (1H1,48X,*TIME*,4X,*BATH C*,4X,*BATH C*,7X,*RESIDUALS*/47X,
 1*MINUTES (DATA)*,5X,*FIT*)*)
 140 FORMAT (43X,F10.2,2F10.5,E17.4)
 141 FORMAT (1H1,7(/),1H,2A10,23X,*MODEL NO. *,I2,* HAS BEEN CHOSEN AT
 1*,F5.2,* MINUTE MARK//1H,2A10,26X,*MODEL IS -- *,3A10//45X,*CO
 2EFFICIENT*,7X,*VALUE*,5X,*CONFIDENCE INTERVAL*,5(/48X,A5,7X,E11.4,
 36X,E10.4))
 142 FORMAT (1H ,/48X,*THE RESIDUAL SUM OF SQUARES = *,E10.4/51X,*CALCU
 1LATION TIME = *,F7.3,* SECONDS*/55X,*NO. OF FCINTS FITTED *,*= *,I
 23//)
 143 FORMAT (1H ,5(/),43X,*CARBON LEVEL PREDICT *,
 1*I C N S *1/43X,49(1H-)//)
 144 FORMAT (1H ,52X,*CARBON LEVEL AT DATUM POINT = *,F5.3//37X,*1) PR
 1EDICTED TIME LEFT UNTIL DATUM POINT IS REACHED = *,F4.0,* SECONDS*
 2/41X,*ACTUAL TIME LEFT*,33X,*= *,F4.0,* SECONDS//37X,*2) CARBON
 3 LEVEL, DATUM PT. TIME (GAS *,*ANALYSIS) = *,F5.3,* PCT.*//41X,*CAR
 4BON LEVEL, DATUM PT. *,*TIME (ACTUAL)*,7X,*= *,F5.3,* PCT.*//37X,*
 5*3) TEST OF MODEL *,*PREDICTION WITH MEASURED LEVEL (GAS ANALYSIS
 6)*//41X,*AT DATUM *,*PCINT.*//51X,*A) MCDEL PREDICTION = *,F5.3,* F
 7CT.*//51X,*B) MEASURED LEVEL = *,F5.3* PCT.*)
 145 FORMAT (1H ,43X,*CARBON MASS BALANCE STARTED AT BEGINNING CF BLCW*
 1/49X,*END POINT CARBON (SPECIFIED) = *,F4.3,* PCT.*//49X,*END PCINT
 2 CARBON (ANALYSED) = *,F4.3,* PCT.*//)
 146 FORMAT (1H ,43X,*CARBON MASS EALANCE STARTED AT CARBON CATLM *,*PC
 1INT*//49X,*END POINT CARBON (SPECIFIED) = *,F4.3,* PCT.*//49X,*ENC P
 2POINT CARBON (ANALYSED) = *,F4.3,* PCT.*//)
 147 FORMAT (1H ,35X,*1) PREDICTED TIME UNTIL SPEC PCINT IS REACHED =
 1*,F5.0,* SECONDS*//40X,*PREDICTED TIME UNTIL END POINT IS *,*REACHE
 2D = *,F5.0,* SECONDS*//40X,*ACTUAL TIME LEFT*,27X,*= *,F5.0,* SECC
 3NDS*//36X,*2) CARBON LEVEL, END POINT TIME (GAS *,*ANALYSIS) = *
 4,F5.3,* PCT.*//40X,*CARBCN LEVEL, END POINT TIME *,*(ACTUAL)*,7X,*= *
 5*,F5.3,* PCT.*//36X,*3) TEST OF MODEL *,*PREDICTION WITH MEASUR
 6ED LEVEL (GAS ANALYSIS)*//40X,*AT END POINT.*//51X,*A) MCDEL PREDIC
 7TION = *,F5.3,* PCT.*//51X,*B) MEASURED LEVEL = *,F5.3,* PCT.*)
 148 FORMAT (1H ,//1H,*FTIME1 = *,F4.0,2X,*TMLFT1 = *,F4.0//1H,*FTIM
 1E2 = *,F4.0,2X,*TMLFT2 = *,F4.0)
 149 FORMAT (54(/))
 END

SUBROUTINE ARANGE (SCC2,SCO,SCH4,S02,C02ELY,CCDLY,CH4DLY,O2DLY,TCY 3
1CLE,N)

ORANGE SHIFTS DATA IN THE ARRAYS TO COMPENSATE FOR TIME DELAYS IN THE GAS SAMPLING AND ANALYSIS SYSTEM.

SUBROUTINE CALLED BY: 1)CONTRL

SUBROUTINES CALLED : NCNE

DIMENSION SC02(N), SC0(N), SCH4(N), SO2(N)

N1=CO2DL Y/TCYCLE

N2=CCDL Y/TCYCLE

N3=CH4DLY/TCYCLE
N1=628LY/TCYCLE

N4=C20L Y/TCYCLE
R0 1 T+1 N

DO 1 I=1,N
S003(I)=S01

$$S_{C02}(T) = S_{C02}(T+N)$$

$SCH_4(T) \equiv SCH_4(T+N)$

$$SCH_4(1) \equiv SCH_4(1+N_3)$$

S02(1)-S02(1+N4)
CONTINUE

CONTINUE
RETURN

END

END

1

תְּהִלָּה תְּהִלָּה תְּהִלָּה תְּהִלָּה תְּהִלָּה

```
SUBROUTINE XDIM (SCO2,SCO,S02,NF,NT,J6,X,TME)
DIMENSION SCO2(NT),SCO(NT),S02(NT),X(NT,NF),TME(NT)
```

DO 1 I=1,NT

$\times(1, 1) = \text{SCO}2$

$$X(I,2) = \text{SCO}(I)$$

$x(1,3) = s02(1)$

CONTINUE

CALL NPL
RETURN

RETURN
ENR

END

1
2
3
4
5
6
7
8
9
0

SUBROUTINE READ1 (MV,UNITS,N)

READ1 READS IN DATA (MV. AND CORRESPONDING ENGINEERING UNITS).

SUBROUTINE CALLED BY: 1) CONTRL

SUBROUTINES CALLED: NCNE

DIMENSION MV(N),UNITS(N)

COMMON /READ/KR/WRITE/KW

REAL MV

READ (KR,3) (MV(I),I=1,N)

READ (KR,3) (UNITS(I),I=1,N)

DO 1 I=1,N

IF (MV(I).EQ.0.0.AND.I.GT.1) GO TO 2

CONTINUE

N=I-1

RETURN

FORMAT (8F10.4)

END

6. 1234567890
6. 10
6. 11
6. 12
6. 13
6. 14
6. 15
6. 16
6. 17
6. 18
6. 19
6. 20
6. 21
6. 22
6. 23-

SUBROUTINE MODEL1 (NPROB,B,FY,NOB,NC,X,NVARX,NCBMAX)

DIMENSION B(NVARX),FY(NCBMAX),X(NVARX,NOBMAX)

DO 1 J=1,NOB

FY(J)=B(1)+B(2)*X(1,J)+B(3)*X(1,J)**2

CONTINUE

RETURN

END

7. 1234567890
7. 7.
7. 7.
7. 7.
7. 7.
7. 7.
7. 7.
7. 7.
7. 7.-

SUBROUTINE MODEL2 (NPROB,B,FY,NOB,NC,X,NVARX,NCBMAX)

DIMENSION B(NVARX),FY(NCBMAX),X(NVARX,NOBMAX)

REAL N

N=16.0

DO 1 J=1,NOB

FY(J)=B(1)**X(1,J)+B(2)+B(3)

CONTINUE

RETURN

END

8. 1234567890
8. 8.
8. 8.
8. 8.
8. 8.
8. 8.
8. 8.
8. 8.
8. 8.-

SUBROUTINE MODEL3 (NPROB,B,FY,NOB,NC,X,NVARX,NCBMAX)
DIMENSION B(NVARX),FY(NCBMAX),X(NVARX,NOBMAX)
DO 1 J=1,NOB
CONTINUE
RETURN
END

9.
9.
9.
9.
9.
9.
9.
1
2
3
4
5
6
-

1

SUBROUTINE MODEL4 (NPROB,B,FY,NOB,NC,X,NVARX,NOBMAX)
DIMENSION B(NVARX),FY(NCBMAX),X(NVARX,NOBMAX)
DO 1 J=1,NOB
FY(J)=B(1)+B(2)*X(1,J)
1 CONTINUE
RETURN
END

10.
10.
10.
10.
10.
10.
10.
10.
10.
10.
10.
10.
10.
10.
10.
1
2
3
4
5
6
7
-

1

SUBROUTINE MODEL5 (NPROB,B,FY,NOB,NC,X,NVARX,NCBMAX)
DIMENSION B(NVARX),FY(NCBMAX),X(NVARX,NOBMAX)
DO 1 J=1,NOB
FY(J)=X(1,J)/(-B(1)+B(2)*X(1,J))
1 CONTINUE
RETURN
END

11.
11.
11.
11.
11.
11.
11.
11.
11.
11.
11.
11.
11.
11.
1
2
3
4
5
6
7
-

1

SUBROUTINE MODEL6 (NPROB,B,FY,NOB,NC,X,NVARX,NCBMAX)
DIMENSION B(NVARX),FY(NCBMAX),X(NVARX,NOBMAX)
DO 1 J=1,NOB
FY(J)=B(1)*B(2)**X(1,J)
1 CONTINUE
RETURN
END

12.
12.
12.
12.
12.
12.
12.
12.
12.
12.
12.
12.
12.
12.
1
2
3
4
5
6
7
-

1

```

SUBROUTINE MODEL7 (NPROB,B,FY,NOB,NC,X,NVARX,NCBMAX) 13.
DIMENSION B(NVARX),FY(NCBMAX),X(NVARX,NOBMAX) 13.
DO 1 J=1,NOB 13.
1 FY(J)=1.0/(B(1)+B(2)*X(1,J)) 13.
CONTINUE 13.
RETURN 13.
END 13.

```

1
2
3
4
5
6
7-

```

SUBROUTINE MODEL8 (NPROB,B,FY,NOB,NC,X,NVARX,NCBMAX) 14.
DIMENSION B(NVARX),FY(NCBMAX),X(NVARX,NOBMAX) 14.
DO 1 J=1,NOB 14.
1 FY(J)=B(1)+B(2)*EXP(-B(3)*X(1,J)) 14.
CONTINUE 14.
RETURN 14.
END 14.

```

1
2
3
4
5
6
7-

```

SUBROUTINE MODEL9 (NPROB,B,FY,NOB,NC,X,NVARX,NCBMAX) 15.
DIMENSION B(NVARX),FY(NOBMAX),X(NVARX,NOBMAX) 15.
DO 1 J=1,NOB 15.
1 FY(J)=B(1)+B(2)*X(1,J)**(-B(3)) 15.
CONTINUE 15.
RETURN 15.
END 15.

```

1
2
3
4
5
6
7-

```

SUBROUTINE MODEL10 (NPROB,B,FY,NOB,NC,X,NVARX,NOBMAX) 16.
DIMENSION B(NVARX),FY(NCBMAX),X(NVARX,NOBMAX) 16.
DO 1 J=1,NOB 16.
1 FY(J)=B(1)+B(2)*EXP(-B(3)*X(1,J))+B(4)*EXP(-B(5)*X(1,J)) 16.
CONTINUE 16.
RETURN 16.
END 16.

```

1
2
3
4
5
6
7-

SUBROUTINE ITER8 (B,FY,BY,NVARX,MODEL,TME,K,TIME,NCHOICE)	17.	1
ITER8 ALLOWS CHOOSING AMONG 10 MODELS IN ORDER TO PERFORM	17.	2
ITERATION ON CORRECT ONE.	17.	3
SUBROUTINE CALLED BY: 1) CONTROL	17.	4
SUBROUTINES CALLED: 1) ITER81	17.	5
DIMENSION B(NVARX)	17.	6
EXTERNAL MODEL1,MODEL2,MODEL3,MODEL4,MODEL5,	17.	7
MODEL6,MODEL7,MODEL8,MODEL9,MODEL10	17.	8
GO TO (1,2,3,4,5,6,7,8,9,10), MODEL	17.	9
1 CALL ITER81 (B,FY,BY,NVARX,MODEL1,TME,K,TIME,NCHOICE)	17.	10
2 RETURN	17.	11
2 CALL ITER81 (B,FY,BY,NVARX,MODEL2,TME,K,TIME,NCHOICE)	17.	12
3 RETURN	17.	13
3 CALL ITER81 (B,FY,BY,NVARX,MODEL3,TME,K,TIME,NCHOICE)	17.	14
4 RETURN	17.	15
4 CALL ITER81 (B,FY,BY,NVARX,MODEL4,TME,K,TIME,NCHOICE)	17.	16
5 RETURN	17.	17
5 CALL ITER81 (B,FY,BY,NVARX,MODEL5,TME,K,TIME,NCHOICE)	17.	18
6 RETURN	17.	19
6 CALL ITER81 (B,FY,BY,NVARX,MODEL6,TME,K,TIME,NCHOICE)	17.	20
7 RETURN	17.	21
7 CALL ITER81 (B,FY,BY,NVARX,MODEL7,TME,K,TIME,NCHOICE)	17.	22
8 RETURN	17.	23
8 CALL ITER81 (B,FY,BY,NVARX,MODEL8,TME,K,TIME,NCHOICE)	17.	24
9 RETURN	17.	25
9 CALL ITER81 (B,FY,BY,NVARX,MODEL9,TME,K,TIME,NCHOICE)	17.	26
10 RETURN	17.	27
END	17.	28
	17.	29
	17.	30
	17.	31
	17.	32
	17.	33
	17.	34
	17.	35
	17.	36-

	SUBROUTINE ITER81 (B,FY,BY,NVARX,MODEL,TME,K,TIME,NCHOICE)	18.	1
C		18.	2
C		18.	3
C		18.	4
C		18.	5
C		18.	6
C		18.	7
C		18.	8
C		18.	0
C		18.	10
C	DIMENSION B (NVARX)	18.	11
1	IF (NCHOICE.EQ.1) GO TO 1	18.	12
	GO TO 2	18.	13
1	CALL MODEL (NPROB,B,BY,1,1,TIME,1)	18.	14
	RETURN	18.	15
2	CONTINUE	18.	16
	N1=1010	18.	17
	N2=6	18.	18
	A1=FY	18.	19
	A2=1.0/10.0**N2	18.	20
	T=1.0/5.0	18.	21
	TIME=TME	18.	22
	K=0	18.	23
	COUNT=T=0.0	18.	24
	T1=1.0	18.	25
	DO 4 I=1,N1	18.	26
	K=K+1	18.	27
	CALL MODEL (NPROB,B,FY,1,1,TIME,1)	18.	28
	TEST=FY/A1-1.0	18.	29
	IF (ABS(TEST).LT.A2) GO TO 5	18.	30
	IF (FY.LT.A1) GO TO 3	18.	31
	IF (T1.LT.0.0) T=T/2.0	18.	32
	T1=T	18.	33
	COUNT=0.0	18.	34
3	GO TO 4	18.	35
	COUNT=COUNT+1.0	18.	36
	T1=-T	18.	37
4	TIME=TIME+T1	18.	38
	GO TO 6	18.	39
5	K=K-1	18.	40
	CONTINUE	18.	41
	BY=FY	18.	42
	FY=A1	18.	43
	RETURN	18.	44
6	END	18.	45
		18.	46-

```

SUBROUTINE EQUATN (B,EQU,NN5,TIM,JJ1)          19.
C
C   EQUATN CHOOSES CORRECT MODEL FOR CALCULATION    19.
C   OF FITTED DATA POINTS.                           19.
C
C   SUBROUTINE CALLED BY: 1) CONTRL                19.
C
C   SUBROUTINES CALLED: MODELS 1-10                 19.
C
C
DIMENSION B(1),EQU(1),TIM(1)                   19.
GC TC (1,2,3,4,5,6,7,8,9,10), JJ1            19.
1 CALL MODEL1 (NPROB,E,EQU,NN5,0,TIM,1,1)      19.
RETURN                                         19.
2 CALL MODEL2 (NPROB,B,EQU,NN5,0,TIM,1,1)      19.
RETURN                                         19.
3 CALL MODEL3 (NPROB,B,EQU,NN5,0,TIM,1,1)      19.
RETURN                                         19.
4 CALL MCDEL4 (NPROB,B,EQU,NN5,0,TIM,1,1)      19.
RETURN                                         19.
5 CALL MODEL5 (NPROB,B,EQL,NN5,0,TIM,1,1)      19.
RETURN                                         19.
6 CALL MODEL6 (NPROB,B,EQU,NN5,0,TIM,1,1)      19.
RETURN                                         19.
7 CALL MODEL7 (NPROB,B,EQL,NN5,0,TIM,1,1)      19.
RETURN                                         19.
8 CALL MCDEL8 (NPROB,B,EQU,NN5,0,TIM,1,1)      19.
RETURN                                         19.
9 CALL MCDEL9 (NPROB,B,EQU,NN5,0,TIM,1,1)      19.
RETURN                                         19.
10 CALL MODEL10 (NPROB,B,EQU,NN5,0,TIM,1,1)     19.
RETURN                                         19.
END

```

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

```

C SUBROUTINE TAINT (XTAB, FXTAB, XBAR, FXBAR, NOPTS, NORD, NERR, MON)      20.    1
C AKINT - TABLE LOOK-UP AND INTERPOLATION SUBROUTINE.                      20.    2
C MCN IS 0 IF TABLE IS TO BE CHECKED FOR MONOTONICITY, +1 IF               20.    3
C ARGUMENT TABLE IS INCREASING, -1 IF DECREASING.                           20.    4
C NERR IS 1 IF TABLE IS IN ORDER, 2 IF THERE ARE TOO FEW ENTRIES FOR        20.    5
C THE ORDER OF INTERPOLATION DESIRED, OR IF NORD > 9, 3 IF THERE             20.    6
C ARE TWO EQUAL ENTRIES, 4 IF THE ARGUMENT TABLE IS NOT MONOTONIC.          20.    7
C DIMENSION XTAB(1),FXTAB(1),X(10),FX(10)                                20.    8
C CHECK NUMBER OF ENTRIES AND ORDER OF INTERPOLATION                         20.    9
C FXBAR=0.                                                               20.   10
C NERR=2.                                                               20.   11
C IF (NORD.GT.9) RETURN.                                              20.   12
C IF (NOPTS.LT.NORD+1) RETURN.                                         20.   13
C AMON=MON.                                                               20.   14
C IF (IABS(MON).EQ.1) GO TO 5.                                           20.   15
C CHECK FOR MONOTONIC ARGUMENT TABLE.                                       20.   16
C NERR=3.                                                               20.   17
C IF (XTAB(2).EQ.XTAB(1)) RETURN.                                         20.   18
C AMON=(XTAB(2)-XTAB(1))/ABS(XTAB(2)-XTAB(1))                          20.   19
C DO 1 JJ=2,NOPTS.                                                       20.   20
C 1 IF ((XTAB(JJ)-XTAB(JJ-1))*AMON) 2,3,1.                               20.   21
C CONTINUE.                                                               20.   22
C GO TO 4.                                                               20.   23
C NOT MCNOTONIC.                                                       20.   24
C NERR=4.                                                               20.   25
C EQUAL ENTRIES.                                                       20.   26
C RETURN.                                                               20.   27
C MCN=AMON*1.1.                                                       20.   28
C NUM WILL BE ADDRESS OF NEAREST ENTRY.                                     20.   29
C NUM IS 1 FOR EXTRAPOLATION BELOW THE TABLE.                            20.   30
C NUM=1.                                                               20.   31
C NERR=1.                                                               20.   32
C IF (((MON.EQ.1).AND.(XBAR.LT.XTAB(1))).OR.((MCN.EQ.(-1)).AND.(XBAR
1.GT.XTAB(1)))) GO TO 12.                                              20.   33
C 1 NUM=NOPTS.                                                       20.   34
C IF (((MON.EQ.1).AND.(XBAR.GT.XTAB(NOPTS))).OR.((MON.EQ.(-1)).AND.(XBAR
1.LT.XTAB(NOPTS)))) GO TO 12.                                         20.   35
C NOW FIND NUM FOR AN INTERIOR POINT BY BINARY SEARCH.                  20.   36
C MIN=1.                                                               20.   37
C MAX=NOPTS.                                                       20.   38
C IF (NOPTS.EQ.NORD+1) GO TO 14.                                         20.   39
C 6 IF ((MAX-MIN).LT.5) GO TO 9.                                         20.   40
C NUM=(MAX+MIN)/2.                                                       20.   41
C 7 IF (AMON*(XTAB(NUM)-XBAR)) 7,18,8.                                 20.   42
C MIN=NUM.                                                               20.   43
C GO TO 6.                                                               20.   44
C 8 MAX=NUM.                                                               20.   45
C GO TO 6.                                                               20.   46
C DETAILED SEARCH FOR NEAREST ENTRY ON LOW SIDE OF XBAR AMONG FIVE       20.   47
C NUM=MAX.                                                               20.   48
C 10 IF (AMCN*(XTAB(NUM)-XBAR)) 12,18,11.                                20.   49
C 11 NUM=NUM-1.                                                               20.   50
C GO TO 10.                                                               20.   51
C C FOUND IT.                                                               20.   52
C 12 IF (NORD.EQ.0) GO TO 17.                                         20.   53
C MIN=NUM-NORD/2.                                                       20.   54
C MAX=MIN+NORD.                                                       20.   55

```

```

C      FIX NUM TO KEEP WITHIN ENDS OF TABLE
IF (MIN.GE.1) GO TO 13
MAX=MAX-MIN+1
MIN=1
GO TO 14
IF (MAX.LE.NOPTS) GO TO 14
MIN=MIN+NOPTS-MAX
MAX=NOPTS
C      SET UP THE X AND FX ARRAYS
14     KKK=1
DO 15 NUM=MIN,MAX
X(KKK)=XTAB(NUM)
FX(KKK)=FXTAB(NUM)
KKK=KKK+1
15     CONTINUE
KKK=KKK-1
C      AIT FEN INTERPOLATION
DO 16 II=2,KKK
DO 16 JJ=II,KKK
FX(JJ)=(FX(II-1)*(X(JJ)-XBAR)-FX(JJ)*(X(II-1)-XBAR))/(X(JJ)-X(II-1))
16     CONTINUE
FXBAR=FX(KKK)
RETURN
C      FICK CUT CLOSEST ENTRY WHEN USED AS LOOK UP ROUTINE
17     IF (ABS(XTAB(NUM+1)-XBAR).LE.ABS(XTAB(NUM)-XBAR)) NUM=NUM+1
FXBAR=FXTAB(NUM)
RETURN
END

```

RESULTS -- FROM DATUM POINT

TIME MINUTES	CEV LB/MIN	ERROR (ABS)	ERROR (REL)	CEV LB	ERROR (ABS)	ERROR (REL)	BATH C (PCT)	ERROR (ABS)	ERROR (REL)
22.33	15	16.9	110.1	0.0	0.0	0.0	245	0.207	8.46
22.39	16	16.8	108.4	0.8	0.7	0.0	245	0.207	8.47
22.44	14	17.0	123.3	7.7	6.7	0.0	244	0.207	8.48
22.49	15	16.8	108.4	6.6	5.6	0.0	244	0.207	8.49
22.55	16	17.1	110.1	8.7	7.7	0.0	244	0.207	8.50
22.60	15	16.6	109.2	7.6	6.6	0.0	244	0.207	8.50
22.65	12	16.5	136.0	8.7	7.7	0.0	244	0.208	8.50
22.70	10	16.6	145.2	7.7	6.7	0.0	244	0.208	8.50
22.76	11	16.7	162.7	8.7	7.7	0.0	244	0.208	8.50
22.81	10	16.7	127.5	8.7	7.7	0.0	244	0.210	8.57
22.86	11	17.0	150.0	8.7	7.7	0.0	244	0.210	8.67
22.91	11	16.9	180.0	8.7	7.7	0.0	244	0.210	8.70
22.97	11	16.9	178.0	8.7	7.7	0.0	244	0.211	8.75
23.02	11	16.9	201.0	8.7	7.7	0.0	244	0.211	8.81
23.07	11	16.9	176.0	8.7	7.7	0.0	244	0.211	8.85
23.12	11	16.9	183.0	8.7	7.7	0.0	244	0.211	8.88
23.17	10	16.9	201.0	8.7	7.7	0.0	244	0.211	8.93
23.22	10	16.9	178.0	8.7	7.7	0.0	244	0.211	8.97
23.27	11	16.9	178.0	8.7	7.7	0.0	244	0.211	9.07
23.32	11	16.9	201.0	8.7	7.7	0.0	244	0.211	9.12
23.37	11	16.9	178.0	8.7	7.7	0.0	244	0.211	9.13
23.42	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.15
23.47	10	16.9	201.0	8.7	7.7	0.0	244	0.211	9.16
23.52	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.17
23.57	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.18
23.62	10	16.9	201.0	8.7	7.7	0.0	244	0.211	9.19
23.67	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.20
23.72	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.21
23.77	10	16.9	201.0	8.7	7.7	0.0	244	0.211	9.22
23.82	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.23
23.87	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.24
23.92	10	16.9	201.0	8.7	7.7	0.0	244	0.211	9.25
23.97	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.26
24.02	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.27
24.07	10	16.9	201.0	8.7	7.7	0.0	244	0.211	9.28
24.12	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.29
24.17	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.30
24.22	10	16.9	201.0	8.7	7.7	0.0	244	0.211	9.31
24.27	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.32
24.32	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.33
24.37	10	16.9	201.0	8.7	7.7	0.0	244	0.211	9.34
24.42	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.35
24.47	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.36
24.52	10	16.9	201.0	8.7	7.7	0.0	244	0.211	9.37
24.57	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.38
24.62	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.39
24.67	10	16.9	201.0	8.7	7.7	0.0	244	0.211	9.40
24.72	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.41
24.77	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.42
24.82	10	16.9	201.0	8.7	7.7	0.0	244	0.211	9.43
24.87	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.44
24.92	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.45
24.97	10	16.9	201.0	8.7	7.7	0.0	244	0.211	9.46
25.02	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.47
25.07	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.48
25.12	10	16.9	201.0	8.7	7.7	0.0	244	0.211	9.49
25.17	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.50
25.22	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.51
25.27	10	16.9	201.0	8.7	7.7	0.0	244	0.211	9.52
25.32	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.53
25.37	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.54
25.42	10	16.9	201.0	8.7	7.7	0.0	244	0.211	9.55
25.47	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.56
25.52	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.57
25.57	10	16.9	201.0	8.7	7.7	0.0	244	0.211	9.58
25.62	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.59
25.67	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.60
25.72	10	16.9	201.0	8.7	7.7	0.0	244	0.211	9.61
25.77	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.62
25.82	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.63
25.87	10	16.9	201.0	8.7	7.7	0.0	244	0.211	9.64
25.92	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.65
25.97	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.66
26.02	10	16.9	201.0	8.7	7.7	0.0	244	0.211	9.67
26.07	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.68
26.12	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.69
26.17	10	16.9	201.0	8.7	7.7	0.0	244	0.211	9.70
26.22	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.71
26.27	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.72
26.32	10	16.9	201.0	8.7	7.7	0.0	244	0.211	9.73
26.37	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.74
26.42	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.75
26.47	10	16.9	201.0	8.7	7.7	0.0	244	0.211	9.76
26.52	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.77
26.57	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.78
26.62	10	16.9	201.0	8.7	7.7	0.0	244	0.211	9.79
26.67	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.80
26.72	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.81
26.77	10	16.9	201.0	8.7	7.7	0.0	244	0.211	9.82
26.82	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.83
26.87	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.84
26.92	10	16.9	201.0	8.7	7.7	0.0	244	0.211	9.85
26.97	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.86
27.02	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.87
27.07	10	16.9	201.0	8.7	7.7	0.0	244	0.211	9.88
27.12	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.89
27.17	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.90
27.22	10	16.9	201.0	8.7	7.7	0.0	244	0.211	9.91
27.27	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.92
27.32	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.93
27.37	10	16.9	201.0	8.7	7.7	0.0	244	0.211	9.94
27.42	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.95
27.47	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.96
27.52	10	16.9	201.0	8.7	7.7	0.0	244	0.211	9.97
27.57	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.98
27.62	10	16.9	178.0	8.7	7.7	0.0	244	0.211	9.99
27.67	10	16.9	201.0	8.7	7.7	0.0	244	0.211	10.00

o₂

TIME MINUTES	CEV LB/MIN	ERROR (ABS)	ERROR (REL)	CEVD LB	ERROR (ABS)	ERROR (REL)	BATH C (PCT)	ERROR (ABS)	ERROR (REL)
25.03	319	15.5		260	17.3		163	0.249	15.25
25.08	310	15.5		277	15.8		158	0.251	15.84
25.13	298	15.4		293	15.3		153	0.252	16.45
25.19	286	15.3		307	14.8		148	0.255	16.08
25.24	271	15.0		314	14.4		144	0.255	17.40
25.29	262	15.2		317	13.3		133	0.225	18.90
25.34	255	15.1		317	12.3		123	0.225	19.50
25.40	247	15.1		317	11.7		117	0.226	20.36
25.45	234	15.0		317	11.4		114	0.226	21.22
25.50	217	15.1		317	11.0		110	0.226	22.34
25.56	193	15.0		317	10.9		109	0.227	22.77
25.61	166	14.8		317	10.7		107	0.227	23.57
25.66	141	14.7		317	10.5		105	0.227	24.80
25.71	121	15.4		317	10.3		103	0.227	25.91
25.77	103	15.6		317	10.1		101	0.227	26.12
25.82	87	15.6		317	9.9		99	0.227	26.58
25.87	74	16.0		317	9.7		97	0.227	27.04
25.92	67	16.6		317	9.5		95	0.227	27.90
25.98	60	16.6		317	9.3		93	0.227	28.67
26.03	54	16.9		317	9.1		91	0.227	29.40
26.08	44	16.9		317	8.9		89	0.227	29.97
26.14	41	16.9		317	8.7		87	0.227	30.24
26.19	39	16.9		317	8.5		85	0.227	30.60
26.24	33	16.9		317	8.3		83	0.227	30.90
26.29	29	16.9		317	8.1		81	0.227	31.16
26.35	26	16.9		317	7.9		79	0.227	31.46
26.40	23	16.9		317	7.7		77	0.227	31.76
26.45	21	16.9		317	7.5		75	0.227	32.07
26.51	19	16.9		317	7.3		73	0.227	32.37
26.56	17	16.9		317	7.1		71	0.227	32.67
26.61	18	16.9		317	6.9		69	0.227	32.97
26.66	18	16.9		317	6.7		67	0.227	33.27
26.72	14	16.9		317	6.5		65	0.227	33.58
26.77	14	16.9		317	6.3		63	0.227	33.88
26.82	15	16.9		317	6.1		61	0.227	34.18
26.88	15	16.9		317	5.9		59	0.227	34.48
26.93	16	16.9		317	5.7		57	0.227	34.78
27.03	15	16.9		317	5.5		55	0.227	35.08
27.09	15	16.9		317	5.3		53	0.227	35.38
27.14	13	16.9		317	5.1		51	0.227	35.68
27.19	13	16.9		317	4.9		49	0.227	36.08
27.24	14	16.9		317	4.7		47	0.227	36.38

O₂ off→