1136 Hardwood Dr.
Palrico, FL 33594

Tele & Fax 813-654-3823
Pager 800-535-1135

PREPARED BY: KENNETH V. FIELD
COMPAD INC.

**TAMS COUPLING
ALIGNMENT
SYSTEM**

COMPUTER

OPERATIONS
MANUAL

TAMS
CASE

RIM DIGITAL PANEL METERS

SENSOR ZEROING ADJUSTMENT

FACE DIGITAL PANEL METERS

COUPLING SOFTWARE

TAMS FRONT PANEL

COUPLING SENSOR

COUPLING

STEAM TURBINE SHAFT

DOE PROJECT 353

FAMS PROTOTYPE AND PRODUCTION COUPLING ALIGNMENT UNITS

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

MASTER

COMPAD6418 US HWY 41N. STE 205
APOLLO BEACH, FL 33572

(DOE UNIT)

813-645-3706

Tele: 813-641-2030

After Hours 813-645-3706

Modern Alignment 813-641-2008

TRUELINE AUTOMATED MEASUREMENT SYSTEM**TAMS**

TAMS is a fully automated alignment system. When TAMS is coupled to a micro-computer it becomes the ultimate alignment hardware and software for professionals responsible for turbines, generators, pumps, and motors just like you! COMPAD has been designing and selling alignment tools since 1984 to the power generation industries. TAMS is easy to use and can be in service in a matter of minutes. The TAMS unit is flexible and versatile and can be used on a variety of turbines, pumps and motors. We stand firmly behind any product we build at COMPAD and offer the highest quality product attainable for the power industry. We introduced the first commercially available computer alignment system in the USA to run on a micro-computer for large steam turbines. TAMS transforms your alignment during an overhaul into an UNEQUALED process measured in terms of accuracy, reduced maintenance time and a smoother running machine. TAMS is simple to use, sets up quickly, and dramatically simplifies alignment.

TAMS SYSTEM OUTLINE

Copyright (C) Compad, Inc. 1991

COMPAD MODEL 490 TAMS UNIT AND SENSOR

CASE: Heavy duty shock mounted. Dim 23"X23"X10".

ALIGNS: Turbines, pumps, motors.

ENVIRONMENT: TAMS can operate in temperatures up to 115 (F)

SENSORS: Manufactured by COMPAD and are of the direct measurement type, accurate to within 0.00025"

TIME: Reduces alignment time by 50%.

TAMS comes with a full one year parts and labor WARRANTY.
THIS PRODUCT DESIGNED, MANUFACTURED, AND ASSEMBLED IN THE USA.
COMPAD reserves the right to alter this product at anytime.

COMPUTERIZED ALIGNMENT DEVICES, INC.

(DOE UNIT)

COMPAD TECHNICIAN CHECKS THE
ALIGNMENT OF 130 MEGAWATT
WESTINGHOUSE TURBINE
LP-GENERATOR COUPLING DURING
AN OVERHAUL USING TAMS.
(TRUELINE AUTOMATED MEASUREMENT SYS)

U L PICT: SYSTEM READY TO READ DATA.
U R PICT: TAMS SENSOR
(DIRECT MEASUREMENT TYPE)
MORE ACCURATE THAN LASER.

L L PICT: TECHNICIAN CHECKS LP-GEN
ALIGNMENT.

©

COMPAD

INC

1136 Hardwood Dr.
Valrico, FL 33594

Tele & Fax: 813-654-3823

Pager: 800-535-1135

**TRUELINE AUTOMATED MEASUREMENT SYSTEM
TAMS**

TAMS IS RELIABLE AND IS EASY TO USE. ALIGNMENT CHECKS CAN BE MADE WITHOUT TURNING THE SHAFTS IF DESIRED! IMPROVED ALIGNMENTS CAN BE SEEN AS SOON AS YOU USE TAMS. TAMS IS SUPERIOR IN ALL RESPECTS.

TAMS FRONT PANEL

TAMS SYSTEM 934 comes with carrying case, TAMS unit, coupling probe, & 8 channels for diaphragms and all interconnecting cables and computer cards, software & manuals.

TAMS system is geared toward the OEM industry standards. Standard rim & face alignment convention is easy to understand and implement alignment moves.

CALL US NOW TO TEST DRIVE THE TAMS SYSTEM, YOU WILL BE GLAD YOU DID!

TAMS BACK PANEL & COUPLING PROBE

TAMS comes with a full one year parts and labor WARRANTY.

**THIS PRODUCT DESIGNED, MANUFACTURED, AND ASSEMBLED IN THE USA.
COMPAD reserves the right to alter this product at anytime.**

Copyright (C) Compad, Inc. 1994

**TURBINE INSPECTIONS & COMPUTER ALIGNMENT & OEM TRAINED ENGINEERS & CONTROLS & STARTUPS & BALANCING
GENERATOR TESTING & STEAM PATH EVALUATIONS & PROJECT MANAGEMENT**

FASTER, MORE RELIABLE TURBINE ALIGNMENT EVERY TIME! LET THE "C2" TURBINE GENERATOR ALIGNMENT SYSTEM GIVE YOU THE EDGE.

There is more to turbine alignment than just sharpening your pencil. More and more utilities are discovering this everyday. The "C2" Turbine Generator Alignment System provides rapid, reliable alignment of steam, gas and nuclear turbines. The "C2" System's technology is not just 5% better but 500% better over existing techniques. "C2" provides the means needed for accomplishing design specified alignment right every time.

The "C2" System achieves this higher standard by providing the user with an improved alignment process measured in terms of:

VASTLY IMPROVED ALIGNMENT

- * Design Quality positioning of shafts & casing for longer equipment life and lower stress.
- * Situation analysis before alignment moves.
- * Standard approach to alignment with documentation for quality control.

The "C2" System is designed specifically for use as a tool for calculating coupling and casing alignment for multi-shaft turbines. "C2" provides the user instantaneous, error free alignment moves (shim changes) based on state of the art technology. Using today's microprocessors, equipment alignment problems become clear and solved immediately.

The "C2" System uses proven industry standard calculations, terminology and methods. By concentrating on the best of these and using new technology we provide desired results.

FEATURES

- * Bearing Pad Shim Changes printed or displayed.
- * Coupling report for analyzing rim & face alignment.
- * Oil and Gland Bore shaft positions calculated with variance from design.
- * Graphics - printed or displayed for:
  - Shaft Moves
  - Coupling Alignment
  - Casing Alignment and Moves including Step Shims
- * Up to 7 shafts calculated simultaneously.
- * Projects alignment moves for all shafts.
- * Data validation and error checking.
- * Horizontal and vertical alignment calculated simultaneously.

TRAINING

- * COMPAD offers complete training for all of its software products. Training is done at either our "Hands On" Computerized Turbine Alignment Workshop or at your job site and taught by a qualified instructor.

BENEFITS OF A "C2" ALIGNMENT

- * Improved uptime of the unit.
- * Improved operational performance.
(lower 2X vibration and better steam rate.)
- * Highly improved alignment analysis.
- * Alignment time cut by as much as 50%.
(Save 2-4 days outage time depending on the size of the turbine.)

FOR MORE INFORMATION Call or Write:

1136 Hardwood Dr.
Valrico, FL 33594

Tele & Fax: 813-554-3823
Pager: 800-535-1135

COMPAD

Copyright (c) Computerized Alignment Devices, Inc.

The Alignment Experts Serving the Power Industry

DOE PROJECT 353

TECHNICAL REPORT AND EVALUATION

Trueline Automated Measurement System (TAMS)
Compad, Inc
Apollo Beach, FL

Project Description

TAMS is an electronic measurement system used to determine the alignment of turbine-generator shafts at the coupling interface. The displacement transducer is a strain gage based sensor mounted in a portable probe. The measurement system was experiencing zero input drift and temperature induced drift. This project endeavored to determine the source of these problems and to revise a unit to be returned to a customer, Baltimore Gas and Electric (BGE), within a period of five weeks.

Detailed Report

As designed, the TAMS unit used a single active strain gage as the measurement transducer. The resulting signal was used to drive TAMS unit panel meters and a personal computer based Data Acquisition System (DAS). The excitation voltage for the strain gage bridge was set at 12 vdc so that the output was approximately 1 mvdc per 1 mil of displacement at the transducer. Front panel potentiometers were used in the loading portion of the single active bridge. The loading resistance was nominally about 1K ohm, therefore the resulting voltage across the strain gage was about 6 vdc. Since the nominal resistance of the gage was 1K ohms, the power dissipation of the gage was about 36 mw. While this power dissipation is within the acceptable range, it is outside the recommended range of 0 - 25 mw and resulted in excess self heating of the gage. The temperature response time constant of the transducer seemed to be on the order of 30 min, resulting in very long stabilization times of about 2 hours.

Our first correction experiment consisted of simply reducing the excitation voltage to 5 vdc. While this voltage reduced the extent of the temperature induced drift, it did not change the zero input temperature effect. We next constructed an equivalent strain gage based temperature compensation network to be used in a half active bridge configuration. This compensation was preferred to be installed in the measurement probe and not require additional cabling between the probe and the TAMS unit.

Initial results with a prototype TAMS and probe were promising although the light piece of metal the compensating gages were mounted to was easily deformed and contributed deformation errors occasionally.

A second compensator was constructed for the BGE unit, but on a stiffer extrusion. The stiffer extrusion suffered a much longer temperature time constant than expected. The first activation of the circuit was acceptable, however. A subsequent long term test was not encouraging. One circuit (face) drifted up with temperature while the other (rim) drifted down.

Additionally, the face circuit had a tendency to continually drift in a positive direction over time. Further testing showed that because of vastly different Thermal Coefficients of Resistance (TCR) between the legs of the active bridges, temperature related errors were still a problem.

The strain gages were disconnected so that each gage could be characterized individually. Once the bonded TCRs of each gage were determined, trimming resistors could be calculated and added in order to match the complimentary legs of the bridges. However, this procedure is an iterative process, therefore time was insufficient to determine the ultimate success of this effort. Initial observations indicated reduced susceptibility to temperature changes but there was still an overall bridge value shift with temperature change.

Wire bundles inside the TAMS unit were subject to heat being generated by the panel meter power supply. Moving these wires away from this heat source seemed to help stabilize meter output. The enclosure cooling fan was reversed so that its air flow would impact directly on the power supply.

A 3.8 vdc power supply was constructed to energize the probe bridge circuit. The lower excitation voltage resulted in lower power dissipation on the strain gages and lower overall shifts due to temperature rise.

Amplifiers were constructed to boost the bridge voltage up to a level that provided the 1 mv = 1 mil displacement relationship. The gain of these amps is about 3.0. The reference voltage was generated by attenuating the 6.9 reference voltage supply from the DAS through the panel potentiometers.

BGE unit change summary

1. Rim and face reference voltages developed from Omega 6.9 vdc source through panel pots to 5-7 vdc
2. Strain gage excitation by new 3.8 vdc source, powered by Omega 12 vdc supply
3. Temperature compensation strain gage network installed in probe
4. Strain gage multiplexer circuits not used
5. Rim and face bridge circuits trimmed for equal TCR
6. Amplifier circuit with adjustable gain installed in TAMS unit
7. Amplifier gains set for 1 mv = 1 mil at opposite ends of span
8. Rim panel meters recalibrated to Fluke DVM
9. Amplifier/regulator board installed on top of Mux board
10. TAMS cooling fan reversed direction to help cool off 5 v power supply

Recommendations

1. Provide signal conditioning and temperature compensation as close to the transducer area as possible.
2. Eliminate effects of instrumentation loop resistance such as cabling, multiplexers, and connectors.
3. Revise meter power supply design for more efficient, cooler operation.
4. Provide shielded instrumentation leads inside TAMS unit and arrange away from noise sources (transformers).

Figure Legend

Voltage data recorded automatically with Omega DAS except as indicated; 1 minute intervals.

Temperature data recorded manually, data points shown as diamonds

1. Drift test at Compad, prototype probe and box
12v excitation
Manual data recording
2. Drift test, proto probe and box
unrecorded ambient temperature variation
5v exc., unamplified
3. Drift test, proto probe and box
ambient temp variation (change not recorded)
5v exc, unamplified
4. Drift test, prototype probe w/external compensation
ambient temperature changing during test
5v exc, unamplified
5. Drift test, proto probe w/ext comp
step change temp variation w/light bulb
5v exc, unamplified
6. Drift test, proto probe w/ext comp
relatively stable amb temp
5v exc, unamplified
7. Drift test, BGE probe w/internal comp, proto box
amb temp changing during test
5v exc, unamplified
8. Drift test, BGE probe w/int comp, proto box
Long term drift test
6.9v exc, unamplified
9. Drift test, BGE probe and compensator gages individually
6.9v exc through 1K ohm resistors
10. Excitation voltage test, individual gages
1K ohm dropping resistors for each gage
11. Drift test, BGE box and probe w/trimmed internal compensation
Voltage manually recorded from front panel meters
3.8v exc, amplified
probe in cooler, ambient temp relatively constant (not recorded)
12. Drift test, BGE box with dummy probe load (1K resistors)
3.8v exc, amplified
13. Drift test, BGE box and probe
3.8v exc, amplified
probe in cooler at start of test, moved to warmer location as shown
14. Drift test, BGE box and probe
3.8v exc, amplified
amb temp increasing slowly (not recorded)
first 75 minutes of data recorded at 5 second intervals
15. Drift test, BGE box and probe
3.8v exc, amplified
amb temp relatively constant (not recorded)

Probe Connection
Internal View

Reference Voltage, typical
6.9V (from 7.5V_{DC})

U1 - LM317T
U2 - LM324

TAMS, BGE Unit
Amplifier Brd & Comp
Compd
3-2-75

BGE Probe
 Compad
 2-5-77-25 14012

Figure 1

Figure 2

Drift Test, 2-16-95
Prototype probe and unit, uncompensated

Figure 3

Figure 4

Drift test, 2-21-95
Prototype probe and unit, compensated

Figure 5

Drift test, 2-22-95
Prototype probe and unit, compensated

Figure 6

Drift test, 2-23-95
BGE probe into prototype box

Figure 7

Drift Test, 2-24-95 to 2-27-95
BGE Probe, 5.9 volt bridge excitation

Figure 8

Drift Test, 2-28-95 BGE gages and compensator gages individually

Figure 9

Excitation voltage test, individual gages
voltage set as shown, pots not adjusted

Figure 10

Drift Test, 3-5-95
BGE box and probe

Figure 11

Drift Test, 3-6-95
BGE unit with dummy probe load

Figure 12

Figure 13

Figure 14

Figure 15

BGE TAMS system user notes

1. Handle the measurement assembly only by the rectangular mounting block and the locking sleeve, avoiding hand contact with the measurement tube. The temperature compensating network is in the connector end of the measurement tube while the displacement transducer is located in the measurement end.
2. Allow the measurement tube to stabilize at the test environment temperature for at least 30 minutes if it is different than that of the storage area.
3. Allow a warm up period for the computer and TAMS system of 15 minutes or more.
4. The rim measurement circuit is stable after approximately one hour of warm up, with approximately a 0.04 mils/min rate of change at 15 minutes and 0.02 mils/min change after 30 minutes.
5. Our tests show that the face measurement circuit takes as long as 2.5 hours to stabilize. However, the approximate rate of change at 15 minutes after startup is 0.05 mils/min and 0.03 mils/min after 30 minutes.
6. Connect the coupling sensor to connector port 1 only.
7. The face sensor has developed a hysteresis of approximately 2 mils. Overtravel the sensor approximately 5 mils and return to the final position before setting the zero lock and before recording displacement values.

TAMS system
B&E Unit

8-3-95

COMPAD, Inc.

1136 Hardwood Dr.
Altico, FL 33584

Tele & Fax 813-654-3823
Pager 800-535-1135

DOE PROJECT 353

FINAL SENSOR DRAWINGS

SPRING
 Ø .142 OD
 Ø .012 WIRE
 .250 FREE LENGTH

REF 15" Ø COUPLING O.D.

REF COUPLING FACE

PROPRIETARY
 1993 (C) COPYRIGHT COMPAD, INC.
 6418 US HWY 41 N.
 APOLLO BEACH, FL 33572
 (813) 645-3706

SECTION A-A

DOE PROJECT 353 COUPLING ALIGNMENT SENSOR

PROPRIETARY
 1993 (C) COPYRIGHT COMPAD, INC.
 6418 US HWY 41 N.
 APOLLO BEACH, FL 33572
 (813) 645-3706

DRILL ϕ .156 THROUGH ---
 CORE ϕ .250 X .187 DEEP
 3 HOLES

ϕ .250051A @.6

MECHANICAL PROPERTIES		PMIC		UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES		CONTRACT NUMBER 931104		Precision Engineering		10850 U.S. 19 N.W. 11L #500 CLEARWATER, FL 34624 (813) 530-1860	
						CONTRACTOR COMPAD		SENSOR BOTTOM RAIL			
TP				TOLERANCES ON 2 PLACE 3 PLACE ± .03 ± .005		DRAWN BY GWS		DATE 11/10/93			
TS						CHECKER		ENGINEER			
CLR						DRAWING APPROVAL					
QA						DESIGN APPROVAL					
WH				MATERIAL CRS MAKE FROM ϕ .500 DRILL ROD							
IN											
		NEXT ASSY		USED ON							
		APPLICATION									
				MATERIAL ENGR							
								SCALE 2/1		SHEET 1 OF 1	

*

					REVISIONS																		
					ZONE	LTR	DESCRIPTION	DATE	APPROVED														
PROPRIETARY ©1993 (C) COPYRIGHT COMPAD, INC. 6418 US HWY 41 N. APOLO BEACH, FL 33572 (813) 645-5706																							
MECHANICAL PROPERTIES		PMIC		UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES		CONTRACT NUMBER 931106				10050 US 9 NORTH, #500 CLEARWATER, FL 34624 (813) 370-3889													
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>YP</td><td></td></tr> <tr><td>IS</td><td></td></tr> <tr><td>EL2</td><td></td></tr> <tr><td>SA</td><td></td></tr> <tr><td>BH</td><td></td></tr> <tr><td>PH</td><td></td></tr> </table>		YP		IS		EL2		SA		BH		PH				TOLERANCES ON 2 PLACE 3 PLACE .03 .005		CONTRACTOR COMPAD		SENSOR TUBE END COVER			
YP																							
IS																							
EL2																							
SA																							
BH																							
PH																							
		MATERIAL: 2024 AL ALLOY		DRAWN BY G.W.S. CHECKER EMERSON		DATE 11/17/93 ENGINEER																	
				DRAWING APPROVAL		DESIGN APPROVAL		SIZE B		CAGE CODE 10-105-01													
		NEXT ASSY USED ON APPLICATION		MATERIAL ENGR				SCALE 2/1		UNIT WT													
										SHEET 1 OF 1													

REVISIONS																																	
ZONE	LTR	DESCRIPTION	DATE	APPROVED																													
1.040 ± 0.002 .25 .896 DRILL AND TAP .625-24 THRU 3 HOLES EQ. SPACED 0.009 BORE .042 X .086 DEEP BREAK OUT PERMISSIBLE																																	
PROPRIETARY © 1993 (C) COMPAD INC. 6415 US HWY 41 N APOLLO BEACH, FL 33772 (813) 645-3706																																	
MECHANICAL PROPERTIES <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>YP</td><td></td></tr> <tr><td>TS</td><td></td></tr> <tr><td>E</td><td></td></tr> <tr><td>RA</td><td></td></tr> <tr><td>BH</td><td></td></tr> <tr><td>HF</td><td></td></tr> </table>		YP		TS		E		RA		BH		HF		PMIC <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td></td><td></td></tr> <tr><td></td><td></td></tr> <tr><td></td><td></td></tr> <tr><td></td><td></td></tr> <tr><td></td><td></td></tr> <tr><td></td><td></td></tr> </table>														UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES TOLERANCES ON 2 PLACE 3 PLACE .03 .005 MATERIAL: 2024 AL ALLOY <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%;">NEXT ASSY</td> <td style="width: 50%;">USED ON</td> </tr> <tr> <td colspan="2" style="text-align: center;">APPLICATION</td> </tr> </table>		NEXT ASSY	USED ON	APPLICATION	
YP																																	
TS																																	
E																																	
RA																																	
BH																																	
HF																																	
NEXT ASSY	USED ON																																
APPLICATION																																	
CONTRACT NUMBER 931104 CONTRACTOR COMPAD <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%;">DRAWN BY G.W.S.</td> <td style="width: 50%;">DATE 11/19/93</td> </tr> <tr> <td style="width: 50%;">CHECKER</td> <td style="width: 50%;">ENGINEER</td> </tr> </table>		DRAWN BY G.W.S.	DATE 11/19/93	CHECKER	ENGINEER	SENSOR TUBE END COVER																											
DRAWN BY G.W.S.	DATE 11/19/93																																
CHECKER	ENGINEER																																
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%;">DRAWING APPROVAL</td> <td style="width: 50%;">DESIGN APPROVAL</td> </tr> </table>		DRAWING APPROVAL	DESIGN APPROVAL	<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 15%;">SIZE B</td> <td style="width: 25%;">CASE CODE</td> <td style="width: 60%;">10-106-01</td> </tr> <tr> <td colspan="2">SCALE 2/1</td> <td>SHEET 1 OF 1</td> </tr> </table>			SIZE B	CASE CODE	10-106-01	SCALE 2/1		SHEET 1 OF 1																					
DRAWING APPROVAL	DESIGN APPROVAL																																
SIZE B	CASE CODE	10-106-01																															
SCALE 2/1		SHEET 1 OF 1																															

REVISIONS			
ZONE	LTG	DESCRIPTION	DATE
			APPROVED

PROPRIETARY
 1993 IC) COPYRIGHT COMPAD, INC
 6416 US HWY 41 N.
 44010 BEACH FL 33572
 (813) 645-3706

MECHANICAL PROPERTIES		PMIC		UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES	CONTRACT NUMBER 931104		18850 U.S. 19 NORTH #500 CLEARWATER FL 34624 (813) 530-3889	WISHBONE SHAFT
					CONTRACTOR COMPAD			
VP				TOLERANCES ON 2 PLACE 3 PLACE ±.03 ±.005	DRAWN BY G.W.S.	DATE 11/23/93	SIZE B	PAGE CODE 10-305-01
VS								
EL2				MATERIAL BRASS	CHECKER	ENGINEER	SCALE 4/1	UNIT WT
RA								
BH				NEXT ASSY APPLICATION	DRAWING APPROVAL DESIGN APPROVAL		SHEET 1 OF 1	
AN								
				MATL SIGN				

REVISIONS		
ZONE	DESCRIPTION	DATE
1		APPROVED

PRECISION ENGINEERING
 1993 US COPYRIGHT COMPAD, INC.
 5418 US HWY 41 N
 APOLEON, FL 32052
 (913) 643-3300

MECHANICAL PROPERTIES	PMSC	UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES	CONTRACT NUMBER 931104	Precision Engineering 3850 US J HIGHWAY #500 CLEARWATER, FL 34624 (813) 330-3889
TP		TOLERANCES UNLESS OTHERWISE SPECIFIED	CONTRACTOR COMPAD	RIM SENSOR SHAFT
TS		2 PLACE 3 PLACE		
EL2		± .03 ± .005	DRAWN BY G.W.S.	SIZE CASE CODE B
RA			DATE 12/2/93	
BH		MATERIAL BRASS	ENGINEER	SCALE 4/1 JUST W/1
RM			DESIGNER	
	NEXT ASSY	USED ON	DESIGN APPROVAL	10-110-01
	APPLICATION			
				SHEET 1 OF 1

REVISIONS		DATE	APPROVED
ZONE	1.25		
DESCRIPTION			

250-25 UNF THREAD

PRECISION
1997 OCT. COPYRIGHT © COMPAD, INC.
6418 US HWY 91 N.
APOLLO BEACH, FL 33612
(813) 643-3756

MECHANICAL PROPERTIES		FIMC		UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES		CONTRACT NUMBER		Precision Engineering		1850 US 9 NORTH, #500 CLEARWATER, FL 34624 (813) 530-3889	
VP						931104					
TS						CONTRACTOR					
US						COMPAD					
RA											
DM											
BT											
				TOLERANCES (IN)		DRAWN BY		DATE		RIM SENSOR SLEEVE	
				2 PLACE 3 PLACE		G.W.S.		12/2/93		SIZE	
				+ .03 ± .005		CHECKER		ENGINEER		CAGE CODE	
				MATERIAL		DRAWING APPROVAL				B	
				BRASS		DESIGN APPROVAL				10-112-01	
				MATERIAL ENGR						SCALE 4/1	
				NEXT ASSY						SHEET 1 OF 1	
				USED ON							
				APPLICATION							

REVISIONS				
ZONE	LTB	DESCRIPTION	DATE	APPROVED

PROPRIETARY
 1993 (C) COMPAGHT COMPAD, INC.
 4418 US HWY 41 N.
 APOLO BEACH, FL 33572
 (813) 445-2706

MECHANICAL PROPERTIES		PMIC		UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES		CONTRACT NUMBER 931104				4450 US 19 NORTH, #500 CLEARWATER, FL 34624 (813) 530-3085	
VP				TOLERANCES ON 2 PLACE 3 PLACE ± .03 ± .005		CONTRACTOR COMPAD		COIL SPRING RETAINER			
IS				MATERIAL BRASS		DRAWN BY G.W.S.					
ELS								CHECKER		ENGINEER	
SA						DRAWER'S APPROVAL				SIZE CAGE CODE	
BH						DESIGN APPROVAL				10-111-01	
MT										B	
		NEXT ASSY USED ON		MAIL ENGR						SCALE 4/1	
		APPLICATION								SHEET 1 OF 1	

ZONE	DATE	DESCRIPTION	DATE	APPROVED

PROPRIETARY
 1940 101 COMPASS CO. COMPASS, INC.
 1425 US HWY. 91 N.
 AVONDALE, ARIZ. 85324
 (602) 643-5704

MECHANICAL PROPERTIES	UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES		CONTRACT NUMBER 931104		1940 US HWY. 91 N. AVONDALE, ARIZ. 85324 (602) 643-5704
	TOLERANCES ON Z PLAC 2 PLAC		CONTRACTOR COMPASS		
	.03 .005		DATE OF DRAWING 11/10/93		
	DRAWING APPROVAL		DESIGN APPROVAL		
NEXT ASSY USED ON		DATE CHG		SCALE 2/1	
APPLICATION				SHEET 1 OF 1	

SENSOR HOLDER

10-100-00

					REVISIONS		
ZONE		LTR		DESCRIPTION		DATE	APPROVED
MECHANICAL PROPERTIES YP _____ TS _____ EL2 _____ RA _____ DH _____ RH _____		PMIC _____ _____ _____ _____ _____		UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES TOLERANCES ON 2 PLACE 3 PLACE .03 .005		CONTRACT NUMBER 931104	
						CONTRACTOR COMPAD	
						DRAWN BY G.W.S.	
						DATE 11/23/93	
						ENGINEER G.W.S.	
MATERIAL CARBON STEEL		DRAWING APPROVAL _____		WISHBONE ROLLER 10-302-01			
		ORIGIN APPROVAL _____					
		SCALE 4/1					
NEXT ASSY _____		USED ON _____		UNIT WT _____		SHEET 1 OF 1	

PROPRIETARY
 1993 (C) COPYRIGHT COMPAD INC
 6418 US HWY 41 N.
 APOLO BEACH, FL 33670
 (813) 641-5725

PROPRIETARY
 1993 (C) COPYRIGHT COMPAD, INC.
 6418 US HWY 41 N.
 APOLLO BEACH, FL 33572
 (813) 645-3706

DRILL ϕ .156 THROUGH
 CBORE ϕ .250 X .187 DEEP
 2 HOLES

$\pm .0005$ A @ B

MECHANICAL PROPERTIES		PMIC		UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES		CONTRACT NUMBER 931104		1850 U.S. 19 NORTH #500 CLEARWATER, FL 34624 (813) 530-3888	
TP				TOLERANCES ON		CONTRACTOR		SENSOR TOP RAIL	
TS				2 PLACE 3 PLACE		COMPAD			
ELC				$\pm .03 \quad \pm .005$		DRAWN BY G.W.S.			
PA				MATERIAL: CRS MAKE FROM ϕ .500 DRILL ROD		DATE 11/10/93			
GH						CHECKER		ENGINEER	
RM						DRAWING APPROVAL			
		NEXT ASSY		USED ON		DESIGN APPROVAL			
		APPLICATION							
				MAIL ENDR				SCALE 2/1 UNIT WT SHEET 1 OF 1	

REVISIONS				
ZONE	LTN	DESCRIPTION	DATE	APPROVED

PROPRIETARY
 1993 CSI COMPAD-ONE COMPAD, INC.
 8418 US HWY 41 N.
 APOLO BEACH, FL 33678
 (813) 845-5706

MECHANICAL PROPERTIES		PMIC		UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES		CONTRACT NUMBER 931104		18850 US 19 NORTH, #500 CLEARWATER, FL 34624 (813) 530-3889	
VP				TOLERANCES UN 2 PLACE 3 PLACE ± .03 ± .005		CONTRACTOR COMPAD			
TS				MATERIAL BRASS		DRAWN BY G.W.S.		DATE 11/23/93	
EL2						CHECKED ENGINEER		ENGINEER	
RA						DRAWING APPROVAL		SENSOR CONTACT SCREW SIZE: B CASE CODE: 10-304-01	
BH						DESIGN APPROVAL			
RI								SCALE: 4/1 UNIT: WT SHEET 1 OF 1	
		NEXT ASSY USED ON		MATERIAL ENGR					
		APPLICATION							

					REVISIONS																														
					ZONE	LINE	DESCRIPTION	DATE	APPROVED																										
					PROPERTY 1993 U.S. COPYRIGHT COMPAD, INC. 6418 US HWY 41 E. APOLO BEACH, FL 33572 (813) 444-3306																														
MECHANICAL PROPERTIES <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td>TP</td> <td></td> </tr> <tr> <td>TS</td> <td></td> </tr> <tr> <td>EL</td> <td></td> </tr> <tr> <td>RA</td> <td></td> </tr> <tr> <td>SH</td> <td></td> </tr> <tr> <td>RI</td> <td></td> </tr> </table>		TP		TS		EL		RA		SH		RI		PMIC <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td></td><td></td></tr> <tr><td></td><td></td></tr> <tr><td></td><td></td></tr> <tr><td></td><td></td></tr> <tr><td></td><td></td></tr> </table>												UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES TOLERANCES ON 2 PLACE 3 PLACE + .03 + .005 MATERIAL: 2024 AL ALLOY		CONTRACT NUMBER 93104 CONTRACTOR COMPAD <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td>DRAWN BY G.W.S.</td> <td>DATE 12/2/93</td> </tr> <tr> <td>CHECKED</td> <td>ENGINEER</td> </tr> </table>		DRAWN BY G.W.S.	DATE 12/2/93	CHECKED	ENGINEER	Precision Engineering 18890 US 19 NORTH #200 CLEARWATER, FL 34624 (813) 530-3823 LOWER ELEMENT TUBE	
		TP																																	
		TS																																	
		EL																																	
		RA																																	
SH																																			
RI																																			
DRAWN BY G.W.S.	DATE 12/2/93																																		
CHECKED	ENGINEER																																		
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td>SIZE B</td> <td>CAGE CODE</td> <td rowspan="4" style="text-align: center; font-size: 1.5em;">10-109-01</td> </tr> <tr> <td colspan="2"></td> </tr> <tr> <td colspan="2"></td> </tr> <tr> <td colspan="2"></td> </tr> </table>		SIZE B	CAGE CODE	10-109-01																															
		SIZE B	CAGE CODE		10-109-01																														
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td>NEXT ASSY</td> <td>USED ON</td> <td rowspan="3">MATE ENGR</td> </tr> <tr> <td colspan="2">APPLICATION</td> </tr> <tr> <td colspan="2"></td> </tr> </table>		NEXT ASSY	USED ON	MATE ENGR	APPLICATION				<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td>DRAWING APPROVAL</td> <td rowspan="3">DESIGN APPROVAL</td> </tr> <tr><td></td></tr> <tr><td></td></tr> </table>		DRAWING APPROVAL	DESIGN APPROVAL																							
		NEXT ASSY	USED ON		MATE ENGR																														
		APPLICATION																																	
DRAWING APPROVAL	DESIGN APPROVAL																																		
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td>SCALE 2/1</td> <td>TUBS W1</td> <td>SHEET 1 OF 1</td> </tr> </table>		SCALE 2/1	TUBS W1	SHEET 1 OF 1																															
		SCALE 2/1	TUBS W1	SHEET 1 OF 1																															

1. APPLY #470 LOCITE PER MANUFACTURER'S INSTRUCTIONS PRIOR TO INSERTING INSULATOR IN PLATE
2. INSULATOR TO BE BONDED TO PLATE SUCH THAT DATUM --A-- IS PERPENDICULAR TO NEAR SIDE AND FAR SIDE OF PLATE WITHIN .0005.

PROPRIETARY
1993 (C) COPYRIGHT COMPANY, INC
6414 175 HWY 41 N.
ANDOLFO BEACH, FL 33512
1813: 645-1444

MECHANICAL PROPERTIES		TENSILE		QUANTITY OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES		CONTRACT NUMBER 931104		ORDER NO. IN HEADLINE CLEARWATER, FL 34624 (813) 532-0091	
1P				TOLERANCES ON		CONTRACTOR		PLATE ASSY.	
1S				2 PLACE 3 PLACE		COMPAD			
1X				.03 .005					
1Y									
2P				MATERIAL		DRAWN BY		DATE	
2S				2024 AL ALLOY		GVS		1/17/2023	
2X						CHECKED		APPROVED	
2Y									
3P						STANDARD, REVISION			
3S									
3X									
3Y									
4P									
4S									
4X									
4Y									
5P									
5S									
5X									
5Y									
6P									
6S									
6X									
6Y									
7P									
7S									
7X									
7Y									
8P									
8S									
8X									
8Y									
9P									
9S									
9X									
9Y									
10P									
10S									
10X									
10Y									
11P									
11S									
11X									
11Y									
12P									
12S									
12X									
12Y									
13P									
13S									
13X									
13Y									
14P									
14S									
14X									
14Y									
15P									
15S									
15X									
15Y									
16P									
16S									
16X									
16Y									
17P									
17S									
17X									
17Y									
18P									
18S									
18X									
18Y									
19P									
19S									
19X									
19Y									
20P									
20S									
20X									
20Y									
21P									
21S									
21X									
21Y									
22P									
22S									
22X									
22Y									
23P									
23S									
23X									
23Y									
24P									
24S									
24X									
24Y									
25P									
25S									
25X									
25Y									
26P									
26S									
26X									
26Y									
27P									
27S									
27X									
27Y									
28P									
28S									
28X									
28Y									
29P									
29S									
29X									
29Y									
30P									
30S									
30X									
30Y									
31P									
31S									
31X									
31Y									
32P									
32S									
32X									
32Y									
33P									
33S									
33X									
33Y									
34P									
34S									
34X									
34Y									
35P									
35S									
35X									
35Y									
36P									
36S									
36X									
36Y									
37P									
37S									
37X									
37Y									
38P									
38S									
38X									
38Y									
39P									
39S									
39X									
39Y									
40P									
40S									
40X									
40Y									
41P									
41S									
41X									
41Y									
42P									
42S									
42X									
42Y									
43P									
43S									
43X									
43Y									
44P									
44S									
44X									
44Y									
45P									
45S									
45X									
45Y									
46P									
46S									
46X									
46Y									
47P									
47S									
47X									
47Y									
48P									
48S									
48X									
48Y									
49P									
49S									
49X									
49Y									
50P									
50S									
50X									
50Y									
51P									
51S									
51X									
51Y									
52P									
52S									
52X									
52Y									
53P									
53S									
53X									
53Y									
54P									
54S									
54X									
54Y									
55P									
55S									
55X									
55Y									
56P									
56S									
56X									
56Y									
57P									
57S									
57X									
57Y									
58P									
58S									
58X									
58Y									
59P									
59S									
59X									
59Y									
60P									
60S									
60X									
60Y									
61P									
61S									
61X									
61Y									
62P									
62S									
62X									
62Y									
63P									
63S									
63X									
63Y									
64P									
64S									
64X									
64Y									
65P									
65S									
65X									
65Y									
66P									
66S									
66X									
66Y									
67P									
67S									
67X									
67Y									
68P									
68S									
68X									
68Y									

NOTES:

1. APPLY #470 LOCTITE PER MANUFACTURER'S INSTRUCTIONS PRIOR TO INSERTING PINS IN MOUNT.

PROPRIETARY

1993 (C) COPYRIGHT COMPAG, INC
6400 IIS HWY 41 N.
APOLLO BEACH, FL 33572
(813) 645-1706

MECHANICAL PROPERTIES		PMIC		UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES		CONTRACT NUMBER 931104		CS Precision Engineering		18850 US 9 NORTH, #500 CLEARWATER, FL 34624 (813) 530 3883	
YP				TOLERANCES ON 2 PLACE 3 PLACE * .03 * .005		CONTRACTOR COMPAG		ELEMENT MOUNT ASSY.			
TS						DRAWN BY G.W.S.					
EL2						CHECKER ENGINEER		SCALE B		CAGE CODE	
RA						DRAWING APPROVAL		10-107-00			
BH						DESIGN APPROVAL		SCALE 2/1		UNIT WT	
RI								SHEET 1 OF 1			
		NEXT ASSY		USED ON							
		APPLICATION									

COMPAD, Inc.

---

136 Hardwood Dr.
Valrico, FL 33594

Tele & Fax 813-654-3823
Pager 800-535-1135

DOE PROJECT 353

PRELIMINARY SENSOR DRAWINGS

PRELIMINARY SENSOR SKETCHES

CUT W/ 0.375" BALL MILL 0.047" DEEP

COPYRIGHT COMPAD, INC 1993

PROPRIETARY SKETCH
SENSOR HOLDER

CONTRA VINC.
R.V. FIEDER
800-221-6557
813-645-3706

END COVE.

PROPRIETARY DWG 1 OF 4

NOT TO SCALE

- NOTE ① QUANTITY REQUIRED - 2 EA, MAKE FROM ALUMINUM
② ALL SURFACES TO BE CHAMFERED, DEBURRED & POLISHED
TO $\sqrt{62}$ FINISH ON OUTSIDE SURFACES.

11-27-90

GENERAL NOTE: ALL SURFACES TO BE CHAMFERED & DEBURRED
SURFACE "A" MUST BE FLAT & PARALLEL
WITHIN $\pm .00025"$, FINISH 32
OR BETTER.

NOTE $\varnothing 1.311"$ HOLE MUST BE BORED PERPENDICULAR & SQUARE
TO SURFACE "A" WITHIN $.0002$ DEGREES).

1/27/10

COMPARED
K.V. FIELDER

PROPRIETARY DWG 4 OF 4

D 1.310 $\frac{+0.01}{-0.01}$

D 1.045 $\frac{+0.00}{-0.01}$

MILL 2 SLOTS $\frac{1}{32}$ " DEEP WITH $\frac{1}{2}$ " BALL END MILL IN PART ②.

DRILL & TAP PART ① FOR 4 SCREWS #6-40, COUNTERSINK @ 2° ANGLE, TAP DRILL #33.
DRILL HOLE WITH #28 IN PART ①. PROVIDE 8 #6-40 SCREWS.

NOTES ① ALL PARTS TO BE POLISHED TO BETTER THAN

62 FINISH, ALL PARTS CHAMFERED $\frac{1}{32}$ " & DEBURRED.

NOT TO SCALE

② PART ① MADE FROM $\frac{1}{2}$ " DRILL ROD. X 4" LONG

③ PART ② MADE FROM 1" SCH 40 STEEL PIPE (SEAMLESS) OR B&E STOCK.

④ QUANTITY REQ'D 1 EA

⑤ FINISH BARE 1.045" DIENSION AFTER WELDING IS COMPLETE

⑥ ALL PARTS TO BE DRAWN/EL. WITHIN ± 0.001 ON 4" LENGTH WITHIN

NOT TO SCALE

Quantity Required 1EA

- NOTES
- ① MAKE PART FROM $\frac{3}{4}$ " SCH 40 STEEL SERVICES PIPE
  - ② POLISH ALL EXTERIOR SURFACES TO $\sqrt{\text{FINISH}}$ OR BETTER
  - ③ ALL EDGES AND CORNERS MUST HAVE $\frac{1}{32}$ " CHAMFER & DEBURRED.
  - ④ ID & OD OF PART MUST BE PARALLEL & CONCENTRIC WITH .001 OVER THE ENTIRE LENGTH OF 12" LENGTH

11-27-90

Ken Ford
644-2080. Office.
645-3706 Home

CHK 1042
P 100.25

DRN BY: KUJ
813-641-2030

$$\sin 10^\circ = \frac{x}{5}$$

$$x = .5(\sin 10^\circ)$$

SIDE VIEW

TITLE: ELEMENT MOUNT & PINS

MTL: 2024 AL ALLOY

FINISH: 32 ✓

NOTE:

1993 (C) COPYRIGHT COMPAD, INC PROPRIETARY SKETCH 11-19-93

TURBINE INSPECTIONS & COMPUTER ALIGNMENT & O&M TRAINED ENGINEERS & CONTROLS & STARTUPS & BALANCING

GENERATOR TESTING & STEAM PATH EVALUATIONS & PROJECT MANAGEMENT

©

COMPAD

INC

6418 US HWY 41N. STE 205

APOLLO BEACH, FL 33572

813-645-3706 TELE

813-645-3706 FAX

DIMENSIONS
ELEMENT TUBE
O.D. = 0.282
I.D. = 0.250"

TURBINE INSPECTIONS • COMPUTER ALIGNMENT • OEM TRAINED ENGINEERS • CONTROLS • STARTUPS • BALANCING

GENERATOR TESTING • STEAM PATH EVALUATIONS • PROJECT MANAGEMENT

COMPAD, Inc.

---

136 Hardwood Dr.
Valrico, FL 33594

Tele & Fax 813-554-3823
Pager 800-535-1135

DOE PROJECT 353

FRONT AND REAR PRODUCTION PANELS FOR TAMS UNIT

0.224" D

COMPAD INC

MODEL 934

0.224" D

TAMS - TRUELINE AUTOMATED MEASUREMENT SYSTEM

METER ON/OFF

SENSOR ADJUST
COARSE SENSOR ADJUST
FINE

RIM ALIGNMENT

FACE ALIGNMENT

METER ON/OFF

7.250"
(7.930")

CARL MAGNUSON

NOTE: CHANGES MADE Added SQUARE PATTERN FAN INTAKE & EXHAUST.
 ALSO Added COMPAD, INC etc.
 ALL OTHER DIMENSION TO PREVIOUS DRAWING SENT. K.U.F.

12-6-93

DOE PROJECT 353

SENSOR CALIBRATION PROCEDURE

COMPAD, INC. (PROPRIETARY)

DATE: 2-9-95

SUBJECT: SENSOR CALIBRATION PROCEDURE

1) TURN COMPUTER & TAMS UNIT ON AND LET THEM WARM UP UNTIL MINIMUM DRIFT IS MAINTAINED

2) EXIT TAMS SOFTWARE TO TAMS DIRECTORY.

SENSOR CALIBRATION PROGRAM IS CALLED CALIB.EXE TO START CALIBRATION OF THE PROBE PROCEED WITH-

(A) C:\TAMS> CALIB.EXE (ENTER)

(B) SELECT CALIBRATION OF RIM OR FACE.

(C) SET SENSOR IN THE Z WAY (AXIS) VISE AS SHOWN IN SKETCH "A" AND MOVE SENSOR TOWARD POST * UNTIL IT JUST TOUCHES THE POST, SET METER OUTPUT TO +49.0 MILS, LOCK FRICTION GRIP ON ADJUSTMENT KNOB.

(PROPRIETARY)

PROBE CALIBRATION CONT. -

PAGE
#2

- d) NOW PRESS ENTER ON THE KEYBOARD - THE FIRST VALUE WILL BE RECORDED IN THE CALIBRATION TABLE.
- e) ADVANCE PROBE TOWARD POST 1 mil @ A TIME ON THE INDICATOR & HIT ENTER UNTIL YOU HAVE DONE ALL POSITIONS.
- f) NEXT PERFORM FACE CALIBRATION IN A SIMILAR MANNER. (SKETCH B)

NOTE: WHEN CALIBRATING (EIM) INDICATOR WILL RUN OUT #1 OF TRAVEL AND MUST BE RESET DURING CALIBRATION PROCEDURE AT ABOUT 56 MILS OF TRAVEL.

G) SAVE CALIBRATION FILE TO DISK.

NOTE: DO NOT OVERTIGHTEN JAWS
#2

(PROPRIETARY)

PROBE CALIBRATION CONT -

Page
#3

NOTE: DO NOT TRAVEL RIM PIN MORE THAN
#3 0.100" (100 MILS).

DO NOT TRAVEL FACE PIN MORE THAN
0.060" (60 MILS).

NOTE: BE SURE INDICATOR IS PERPENDICULAR
#4 AND PARALLEL TO THE PROBE TUBE
BEFORE CALIBRATING.

NOTE: RECORD THE BEFORE AND AFTER
#5 INITIAL STARTING MILS OR METER
READINGS. IF START & END VALUES ARE > 1mi, ^{THEN} RECAL

NOTE: PLOT DIAL INDICATOR VS STORED VALUE
#6 IN CALIBRATION TABLE

③ CAREFULLY PACK SENSOR PROBE IN CARRYING
CASE OR BOX TO PREVENT ANY DAMAGE.

NOTE: THE TAMS DEMO UNIT STARTS
#7 CALIBRATION AT - VALUES
i.e. -49 & -29 ON THE METERS.
THE SOFTWARE CAN INTERPET THE
NEGATIVE VALUES

TAMS SENSOR CALIBRATION CURVE

SENSOR PROTOTYPE 1993

COMPAD, Inc.

---

136 Hardwood Dr.
Valrico, FL 33594

Tel & Fax 813-654-3823
Pager 800-535-1135

DOE PROJECT 353

_RODUCTION CIRCUIT BOARDS AND SENSOR SKETCHE

TOP SIDE TAMS RIM & FACE MULTIPLEXER AND SIGNAL CONDITIONING BOARD

Drill table data			
Tool	Hole	Sym	Note
Code	Size	bol	Count

Board type: F11
 Board thickness: .062"
 Copper weight: 2oz
 All dimensions are in inches.
 Board should have green solder mask over solder with white silk screening on top.
 View is from top side of the board.

Co: Compad Inc

Title: ?

Board: CP2

Drawn: Carl Magnuson

Date: October 12, 1993

Revision: A

Size: A

Sheet 1 of 1

BACKSIDE OF RIM & FACE MULTIPLEXER AND SIGNAL CONDITIONING BOARD

BACK SIDE OF TAMS UNIT POWER SUPPLY

TOP SIDE OF TAMS UNIT POWER SUPPLY

COMPAD

INC.

6418 US HWY 41N STE 205
APOLLO BEACH, FL 33572

8-18-92

813-641-2030 TELE & FAX
813-645-3706 AFTER HOURS

WIRE FOLDED & PRESSED
IN TO ACT AS A STRESS
RELIEF

YELLOW (FACE)
WHITE (FACE)

Gauge Bonded Resistance = 1022 Ω . TEMP 72°F

BLUE (RIM)
Green (RIM)

Gauge Bonded Resistance = 1026 Ω . TEMP 72°F

NOTES ① SENSOR ELEMENT FINISHED 8:45AM (EASTERN) 8-18-92.

CONSTRUCTION NOTES:

SOLDERING TIP: ① WHEN SOLDERING GOLD WIRES OF THE GAUGE TO THE LEADOUT WIRES, FIRST HEAT SOLDER ON THE LEAD WIRE & THEN PUSH THE GOLD LEAD INTO THE SOLDER & IMMEDIATELY REMOVE THE SOLDER IRON.

SWITCHCRAFT PLUG

- ① Green
- ② Black
- ③ White
- ④ Red

DC VOLTAGES 5.0VDC Meters
7.5 Energize Probe

TURBINE INSPECTIONS • COMPUTER ALIGNMENT • OEM TRAINED ENGINEERS • CONTROLS • STARTUPS • BALANCING

GENERATOR TESTING • STEAM PATH EVALUATIONS • PROJECT MANAGEMENT

11/10 11:11 1331

6-3-92

SENSOR

9:21AM EDT
Kenneth V. Field
KENNETH V FIELD

SEMI-CONDUCTOR Strain gage.
ENTRAN DEVICES
1-800-635-0650
10-T WASHINGTON Ave
FAIRFIELD, NJ 07004.
Bob Leary

SMD 803-235-9330 MMP 919-365-3800
HAM SDB-624-4500 TRANSICOM 800-323-7115

Kenneth V. Field
805-522-4676

COMPAD, Inc.

---

136 Hardwood Dr.
Valrico, FL 33594

Tele & Fax 813-654-3823
Pager 800-535-1135

DOE PROJECT 353

COMPUTER SOFTWARE CODE

```

rem
rem This program frees the meters on the TAMS box.
rem Wand #1 is activated and all meters are left on
%include TAMS1.INC

```

```

AN = SetupA2D
if AN <> 0 then stop

call SetWandOn(1)      rem Turn wand #1 on
call SetAllMetersOn    rem Turn all meters on
stop

```

```

REM-----1-----2-----3-----4-----5-----6-----7-----
REM> Copyright (C) COMPAQ Incorporated 1984,5,6
REM> ALL RIGHTS RESERVED
REM>
REM> TITLE: C2-TURBINE GENERATOR ALIGNMENT SYSTEM
REM> VERSION: 86.1
REM> REVISION: #11-15-88
REM> FILE: C2 - C2 BEGINS
REM>
REM>-----1-----2-----3-----4-----5-----6-----7-----
REM> SETUP,

%include COMMON
%include FUNC
      chafe "C2A"

```



```

REM-----2-----3-----4-----5-----6-----7-----
REM> Copyright (C) COMPAD Incorporated 1984,5,6
REM> ALL RIGHTS RESERVED
REM>
REM> TITLE: C2-TURBINE GENERATOR ALIGNMENT SYSTEM
REM> VERSION: 00.1
REM> REVISION: 011-15-88
REM> FILE: C2G - CURRENT ALIGNMENT
REM>
REM>-----1-----2-----3-----4-----5-----6-----7-----
REM> SETUP,
%include COMMON
    def fn.ROG(RNUM,RS) external
    fend
    def fn.CHRM external
    fend
    def fn.EMPTY(ENTS) external
    fend
    def fn.DISPM(NTH,ROW,COL,SS) external
    fend
    def fn.DISPR(NTH,ROW,COL,SS) external
    fend
    def fn.DISPC(NTH,CL,ROW,COL,SS) external
    fend
    def fn.DISUC(NTH,CL,ROW,COL,RS,NUM) external
    fend
    def fn.CLS(NTH,CL) external
    fend
    def fn.FGR(NTH,CL,UNITS,COMPS,RS,SS) external
    fend
    def fn.BUC(NTH,ROW,COL,RS,MAX) external
    fend
    def fn.VNINP(NTH,ROW,COL) external
    fend
    def fn.MESG(NTH,CL,NR,NFILLS,MS) external
    fend
    def fn.INPUT(NTH,CL,ROW,COL,ENTS,EP,MIN,MAX,TYPES,IFS,NR,NFILLS) external
    fend
    def STRIP(AC) external
    string STRIP
    fend
    def on(rnum,lo,hi)
    real rnum
    string lo, hi
    if rnum=0 then go=" " else if rnum<0 then go=lo else go=hi
    return
    fend
    EQUALS=
    "====F1:PrevEntry====F2:NextEntry====Esc:ToMenu====F3:PrevScreen====F4:NextScreen===="
    gobub 01.0
    if ROUTE=2 then goto 1000.0
REM>-----1-----2-----3-----4-----5-----6-----7-----
REM> DISPLAY SCREEN # 405 - CURRENT ALIGNMENT TOLERANCE.
405.0 if ROUTE=2 then gobub 2405.0 also gobub 2405.0
    gobub 00.0
    dim DATAS(50)
    if ROUTE=1 then read 10,50;D1$,D2$,D3$,D4$,D5$,D6$,DATAS(1)\
    also read 10,70;D1$,D2$,D3$,D4$,F1,F2,DATAS(1)
    call fn.DISPC(NTH,CL,24,0,\
    USER$=" ", fill in the information above.      ")
    gobub 00.5
    SRN=1
405.0 DATAS(SRN)=fn.INPUT(NTH,CL,SRN(SRN),BCN(SRN),DATAS(SRN),1,BMIN(SRN),\
    BMAX(SRN),BTYPES(SRN),"",22,EQUALS)
    if mid$(DATAS(SRN),3,2)="W" then KEY$=int$(val(left$(DATAS(SRN),2))) \
    :DATAS(SRN)=mid$(DATAS(SRN),3,BMAX(SRN))
%include TOGGLE.TOL
    if ROUTE=1 then print 10,50;D1$,D2$,D3$,D4$,D5$,D6$,DATAS(1)\
    else print 10,70;D1$,D2$,D3$,D4$,F1,F2,DATAS(1)
    TOL$=DATAS(1)
    gobub 00.00
    if KEY$=5 then ROUTE=2 :chain "C2G"
    if ROUTE=2 then goto 420.0
REM>-----1-----2-----3-----4-----5-----6-----7-----
REM> DISPLAY SCREEN # 410 - CURRENT COUPLING ALIGNMENT.
410.0 gobub 2410.0
    dim DATAS(50), F(0)
    SHN=SRN
410.1 if SRN lt 95 then SHN=95
    gobub 00.0
    call fn.DISPM(NTH,9,6,SHN(SHN))
    call fn.DISPM(NTH,9,20-LEN$(SHN+1),SHN(SHN+1))
    call fn.DISPM(NTH,20,20,\
    "The "SHN(SHN)+"-SRS(SHN)+" (or "+CHR$(84+SHN)+" Coupling")
    read 10,51+(4*(SHN-1));DATAS(3),DATAS(4),DATAS(5),DATAS(6),\
    DATAS(7),DATAS(8),DATAS(9),DATAS(10)
    read 10,52+(4*(SHN-1));DATAS(11),DATAS(12),DATAS(13),DATAS(14),\
    DATAS(15),DATAS(16),DATAS(17),DATAS(18)
    read 10,53+(4*(SHN-1));DATAS(19),DATAS(20),DATAS(21),DATAS(22),\
    DATAS(1),DATAS(2)
    call fn.DISPC(NTH,CL,3,0,TOL$)
410.2 call fn.DISPC(NTH,CL,24,0,\
    USER$=" ", fill in the information above.      ")
    gobub 00.5
    SRN=1
410.3 if SRN le 0 then SRN=1
    DATAS(SRN)=fn.INPUT(NTH,CL,SRN(SRN),BCN(SRN),DATAS(SRN),1,BMIN(SRN),\
    BMAX(SRN),BTYPES(SRN),"",22,EQUALS)
    if mid$(DATAS(SRN),3,2)="W" then KEY$=int$(val(left$(DATAS(SRN),2))) \
    :DATAS(SRN)=mid$(DATAS(SRN),3,BMAX(SRN))
    if SRN NE 1 then goto 410.31
    DATAS(1)=STRIP(DATAS(1))
    call fn.BUC(NTH,SRN(1),BCN(1),DATAS(1),BMAX(1))
    if DATAS(1)=SHN(SHN) or DATAS(1)=SHN(SHN+1) then goto 410.31
    call fn.MESG(NTH,CL,22,EQUALS,"INCORRECT SHAFT NAME.")
    goto 410.3
410.31 if SRN NE 2 then goto 410.32
    DATAS(2)=STRIP(DATAS(2))
    call fn.BUC(NTH,SRN(2),BCN(2),DATAS(2),BMAX(2))
    if DATAS(2)="CW" or DATAS(2)="CCW" then goto 410.32
    call fn.MESG(NTH,CL,22,EQUALS,"INCORRECT ROTATION, USE CW OR CCW.")
    goto 410.3
410.32 if KEY$=1 then SRN=SRN-1 :goto 410.3
    if KEY$ ge 6 and KEY$ le 7 then goto 410.33
    SRN=SRN+1
    if SRN le BNM then goto 410.3

```

```

410.33 L% = 2
410.34 J% = 1
      K% = 0
      IF FN.EMPTY%(DATA$(1%+1)) THEN J% = J% + 1 : K% = 1
      IF FN.EMPTY%(DATA$(1%+2)) THEN J% = J% + 1 : K% = 2
      IF FN.EMPTY%(DATA$(1%+3)) THEN J% = J% + 1 : K% = 3
      IF FN.EMPTY%(DATA$(1%+4)) THEN J% = J% + 1 : K% = 4
      ON J% GOTO 410.36, 410.36, 410.37, 410.37, 410.38
410.35 IF ABS(VAL(TOL$)) < ABS(VAL(DATA$(1%+1)) - VAL(DATA$(1%+2))) - \
      VAL(DATA$(1%+3)) + VAL(DATA$(1%+4))) THEN GOTO 410.38
      CALL FN.MESSAGE(MT%, CL%, 22, EQUALS, "FACE READINGS OUT OF TOLERANCE.")
      SR% = 1% + 1
      GOTO 410.3
410.36 IF K% = 1 THEN DATA$(1%+1) = FN.RCS(VAL(DATA$(1%+3)) - VAL(DATA$(1%+4)) \
      - VAL(DATA$(1%+2))), "FF. NNNNN"
      IF K% = 2 THEN DATA$(1%+2) = FN.RCS(VAL(DATA$(1%+3)) + VAL(DATA$(1%+4)) \
      - VAL(DATA$(1%+1))), "RR. NNNNN"
      IF K% = 3 THEN DATA$(1%+3) = FN.RCS(VAL(DATA$(1%+1)) - VAL(DATA$(1%+2)) \
      + VAL(DATA$(1%+4))), "RR. NNNNN"
      IF K% = 4 THEN DATA$(1%+4) = FN.RCS(VAL(DATA$(1%+1)) - VAL(DATA$(1%+2)) \
      - VAL(DATA$(1%+3))), "RR. NNNNN"
      CALL FN.DISPC%(MT%, BRN(1% - K%), BCU(1% - K%), DATA$(1% - K%))
      GOTO 410.33
410.37 CALL FN.MESSAGE(MT%, CL%, 22, EQUALS, "MINIMUM OF 3 FACE READINGS REQUIRED.")
      SR% = 1% + 1
      GOTO 410.3
410.38 L% = L% + 1
410.39 IF L% < 17 THEN L% = L% + 5 : GOTO 410.34

      J% = 1
      K% = 0
      IF FN.EMPTY%(DATA$( 7)) THEN J% = J% + 1 : K% = 1
      IF FN.EMPTY%(DATA$(12)) THEN J% = J% + 1 : K% = 2
      IF FN.EMPTY%(DATA$(17)) THEN J% = J% + 1 : K% = 3
      IF FN.EMPTY%(DATA$(22)) THEN J% = J% + 1 : K% = 4
      ON J% GOTO 410.41, 410.42, 410.43, 410.43, 410.44
410.41 IF ABS(VAL(TOL$)) < ABS(VAL(DATA$(7)) - VAL(DATA$(17))) - \
      VAL(DATA$(12)) + VAL(DATA$(22))) THEN GOTO 410.44
      CALL FN.MESSAGE(MT%, CL%, 22, EQUALS, "RIM READINGS OUT OF TOLERANCE.")
      SR% = 7
      GOTO 410.3
410.42 IF K% = 1 THEN DATA$( 7) = FN.RCS(VAL(DATA$(12)) - VAL(DATA$(22)) \
      - VAL(DATA$(17))), "NN. NNNNN"
      IF K% = 2 THEN DATA$(12) = FN.RCS(VAL(DATA$( 7)) - VAL(DATA$(17)) \
      - VAL(DATA$(22))), "NN. NNNNN"
      IF K% = 3 THEN DATA$(17) = FN.RCS(VAL(DATA$(12)) + VAL(DATA$(22)) \
      - VAL(DATA$( 7))), "RR. NNNNN"
      IF K% = 4 THEN DATA$(22) = FN.RCS(VAL(DATA$( 7)) + VAL(DATA$(17)) \
      - VAL(DATA$(12))), "RR. NNNNN"
      CALL FN.DISPC%(MT%, BRN((K% + 5) + 2), BCU((K% + 5) + 2), DATA$(K% + 5) + 2)
      GOTO 410.44
410.43 CALL FN.MESSAGE(MT%, CL%, 22, EQUALS, "MINIMUM OF 3 RIM READINGS REQUIRED.")
      SR% = 7
      GOTO 410.3
410.44 IF L% = 0 THEN TOP = 0 : BOT = 0 : LEFT = 0 : RIGHT = 0 \
      : VFACE = 0 : HFACE = 0 : GOTO 410.45
      TOP = (VAL(DATA$(3)) - VAL(DATA$(8)) - VAL(DATA$(13)) - VAL(DATA$(18))) / L%
      BOT = (VAL(DATA$(4)) - VAL(DATA$(9)) - VAL(DATA$(14)) - VAL(DATA$(19))) / L%
      LEFT = (VAL(DATA$(5)) - VAL(DATA$(10)) - VAL(DATA$(15)) - VAL(DATA$(20))) / L%
      RIGHT = (VAL(DATA$(6)) - VAL(DATA$(11)) - VAL(DATA$(16)) - VAL(DATA$(21))) / L%
      VFACE = BOT - TOP
      HFACE = RIGHT - LEFT
410.45 IF DATA$(1) = END$(SR%) THEN GOTO 410.47
      VRIM = (VAL(DATA$(17)) - VAL(DATA$(7))) / 2

      IF DATA$(2) = "CCM" THEN GOTO 410.46
      HRIM = (VAL(DATA$(12)) - VAL(DATA$(22))) / 2
      GOTO 410.49
410.46 HRIM = (VAL(DATA$(22)) - VAL(DATA$(12))) / 2
      GOTO 410.49
410.47 VRIM = (VAL(DATA$(7)) - VAL(DATA$(17))) / 2
      IF DATA$(2) = "CCM" THEN GOTO 410.48
      HRIM = (VAL(DATA$(22)) - VAL(DATA$(12))) / 2
      GOTO 410.49
410.48 HRIM = (VAL(DATA$(12)) - VAL(DATA$(22))) / 2
410.49 F(1) = TOP
      CALL FN.DISPC%(MT%, CL%, 16, 37, "FF. NNNNN", F(1))
      F(2) = BOT
      CALL FN.DISPC%(MT%, CL%, 16, 48, "RR. NNNNN", F(2))
      F(3) = LEFT
      CALL FN.DISPC%(MT%, CL%, 16, 53, "RR. NNNNN", F(3))
      F(4) = RIGHT
      CALL FN.DISPC%(MT%, CL%, 16, 61, "RR. NNNNN", F(4))
      F(5) = VRIM
      CALL FN.DISPC%(MT%, CL%, 18, 47, "RR. NNNNN", F(5))
      CALL FN.DISPC%(MT%, CL%, 18, 54, SR(F(5)), "L", "H")
      F(6) = HRIM
      CALL FN.DISPC%(MT%, CL%, 18, 65, "RR. NNNNN", F(6))
      CALL FN.DISPC%(MT%, CL%, 18, 72, SR(F(6)), "R", "L")
      F(7) = VFACE
      CALL FN.DISPC%(MT%, CL%, 18, 47, "NN. NNNNN", F(7))
      CALL FN.DISPC%(MT%, CL%, 18, 54, SR(F(7)), "C", "O")
      F(8) = HFACE
      CALL FN.DISPC%(MT%, CL%, 18, 65, "NN. NNNNN", F(8))
      CALL FN.DISPC%(MT%, CL%, 18, 72, SR(F(8)), "C", "O")

      IF KEY% = 4 AND KEY% < 7 THEN GOTO 410.4
      CALL FN.DISPC%(MT%, CL%, 24, 6, \
      "USER'S * is this data correct (Y/N)?")
      KEY% = FN.INPR%(MT%, 24, 55)
      IF KEY% = 1 THEN GOTO 410.2
      CALL FN.DISPC%(MT%, CL%, 24, 59, "====> THANKS.")

410.4 REM
%INCLUDE TOGGLE1.TOL
PRINT# 10, 50 + (4 * (SR% - 1)); DATA$(3), DATA$(4), DATA$(8), DATA$(9), \
      DATA$(7), DATA$(8), DATA$(9), DATA$(10)

%INCLUDE TOGGLE1.TOL
PRINT# 10, 52 + (4 * (SR% - 1)); DATA$(11), DATA$(12), DATA$(15), DATA$(14), \
      DATA$(15), DATA$(16), DATA$(17), DATA$(18)

%INCLUDE TOGGLE1.TOL
PRINT# 10, 53 + (4 * (SR% - 1)); DATA$(19), DATA$(20), DATA$(21), DATA$(22), \
      DATA$(1), DATA$(2), TOL$

%INCLUDE TOGGLE1.TOL
PRINT# 10, 54 + (4 * (SR% - 1)); F(1), F(2), F(3), F(4), F(5), F(6), F(7), F(8)
IF KEY% = 6 THEN GOTO 410.5
IF KEY% = 5 THEN SR% = SR% + 1 : GOTO 410.1
SR% = SR% + 1
IF SR% < 8 THEN GOTO 410.1
CLOSE #10
SUB 80.98 : POSUB 80.99 : DIM F(8)
SUB 61.99
ROUTE% = 6

```

```

      chdir "C2C"
REM-----2-----3-----4-----5-----6-----7-----
REM> DISPLAY SCREEN # 420 - CURRENT OIL & GLAND SORE ALIGNMENT.
420.0 SHW=SHW
420.1 if SHW lt 3SHW then SHW=3SHW
if left$(SHW(3SHW),3)="DEN" then gosub 2421.0 :goto 420.11
if left$(SHW(3SHW),3)="EXC" then gosub 2422.0 :goto 420.11
gosub 2420.0
420.11 gosub 60.0
call fn.DISPCN(MTW,20,30,\
  "The " & SHW(3SHW) & " (or " & STR$(SHW) & ") Shaft")
dim DATAS(3SHW)
dim F(8)
read# 10,75+(4*(SHW-1)):DATAS(1),DATAS(2),DATAS(3),DATAS(4),F(1),F(2)
read# 10,76+(4*(SHW-1)):DATAS(5),DATAS(6),DATAS(7),DATAS(8),F(3),F(4)
read# 10,77+(4*(SHW-1)):DATAS(9),DATAS(10),DATAS(11),DATAS(12),F(5),F(6)
read# 10,78+(4*(SHW-1)):DATAS(13),DATAS(14),DATAS(15),DATAS(16),F(7),F(8)
read# 10,79:D15,D25,D35,D45,F1,F2,TOLS
call fn.DISPCN(MTW,CLN,3,60,TOLS)
420.2 call fn.DISPCN(MTW,CLN,24,6,USERS=" ", fill in the information above.")
gosub 60.0
SHW=1
420.3 if SHW le 0 then SHW=1
DATAS(SHW)=fn.INPUT$(MTW,CLN,SHW(SHW),60(SHW),DATAS(SHW),1,SHW(SHW),\
  SHW(SHW),STYPES(SHW),"",22,EQUALS)
if mid$(DATAS(SHW),3,2)="" then KEYS=int(val(left$(DATAS(SHW),2))) \
  :DATAS(SHW)=mid$(DATAS(SHW),5,SHW(SHW))
if KEYS=1 then SHW=SHW-1 :goto 420.3
if KEYS ge 5 and KEYS le 7 then goto 420.31
SHW=SHW+1
if SHW le 6SHW then goto 420.3
420.31 I%=0
420.32 J%=1
K%=0
if fn.EMPTY$(DATAS(I%-1)) then J%=J%+1 :K%=1
if fn.EMPTY$(DATAS(I%-2)) then J%=J%+1 :K%=2
if fn.EMPTY$(DATAS(I%-3)) then J%=J%+1 :K%=3
if fn.EMPTY$(DATAS(I%-4)) then J%=J%+1 :K%=4
on J% goto 420.33, 420.34, 420.35, 420.35, 420.36
420.33 if abs(val(TOLS))-abs(val(DATAS(I%-1))+val(DATAS(I%-3)))<\
  [val(DATAS(I%-2))+val(DATAS(I%-4))] then goto 420.36
call fn.MESSAGE(MTW,CLN,22,EQUALS,"BORE READINGS OUT OF TOLERANCE.")
SHW=I%-1
goto 420.3
420.34 if K%=1 then DATAS(I%-1)=fn.RCS(val(DATAS(I%-2))+val(DATAS(I%-4)) \
  -val(DATAS(I%-3)),"###.###")
if K%=2 then DATAS(I%-2)=fn.RCS(val(DATAS(I%-1))+val(DATAS(I%-3)) \
  -val(DATAS(I%-4)),"###.###")
if K%=3 then DATAS(I%-3)=fn.RCS(val(DATAS(I%-2))+val(DATAS(I%-4)) \
  -val(DATAS(I%-1)),"###.###")
if K%=4 then DATAS(I%-4)=fn.RCS(val(DATAS(I%-1))+val(DATAS(I%-3)) \
  -val(DATAS(I%-2)),"###.###")
call fn.DISPCN(MTW,SHW(I%-K%),60(I%-K%),DATAS(I%-K%))
goto 420.36
420.35 call fn.MESSAGE(MTW,CLN,22,EQUALS,"MINIMUM OF 3 READINGS REQUIRED.")
SHW=I%-1
goto 420.3
420.36 if I% lt 12 then I%=I%+4 :goto 420.32
420.37 F(1)=(val(DATAS(3))-val(DATAS(1)))/2
call fn.DISPCN(MTW,CLN,17,13,"###.###",F(1))
call fn.DISPCN(MTW,CLN,17,21,sn(F(1),"L","H"))
F(2)=(val(DATAS(2))-val(DATAS(4)))/2
call fn.DISPCN(MTW,CLN,18,13,"###.###",F(2))
call fn.DISPCN(MTW,CLN,18,21,sn(F(2),"R","L"))
F(2)=(val(DATAS(7))-val(DATAS(5)))/2

call fn.DISPCN(MTW,CLN,17,25,"###.###",F(3))
call fn.DISPCN(MTW,CLN,17,33,sn(F(3),"L","H"))
F(4)=(val(DATAS(8))-val(DATAS(6)))/2
call fn.DISPCN(MTW,CLN,18,25,"###.###",F(4))
call fn.DISPCN(MTW,CLN,18,33,sn(F(4),"R","L"))
F(5)=(val(DATAS(11))-val(DATAS(9)))/2
call fn.DISPCN(MTW,CLN,17,49,"###.###",F(5))
call fn.DISPCN(MTW,CLN,17,57,sn(F(5),"L","H"))
F(6)=(val(DATAS(10))-val(DATAS(12)))/2
call fn.DISPCN(MTW,CLN,18,49,"###.###",F(6))
call fn.DISPCN(MTW,CLN,18,57,sn(F(6),"R","L"))
F(7)=(val(DATAS(15))-val(DATAS(13)))/2
call fn.DISPCN(MTW,CLN,17,81,"###.###",F(7))
call fn.DISPCN(MTW,CLN,17,89,sn(F(7),"L","H"))
F(8)=(val(DATAS(14))-val(DATAS(16)))/2
call fn.DISPCN(MTW,CLN,18,81,"###.###",F(8))
call fn.DISPCN(MTW,CLN,18,89,sn(F(8),"R","L"))

if KEYS ge 5 and KEYS le 7 then goto 420.4
call fn.DISPCN(MTW,CLN,24,6,\
  USERS=" ", is this data correct (Y/N)? " ")
KEYS=fn.YNINP$(MTW,24,65)
if KEYS=1 then goto 420.2
call fn.DISPCN(MTW,CLN,24,58,"#### THANKS.")

420.4 rem
%include TOGGLE1.TOL
print# 10,75+(4*(SHW-1)):DATAS(1),DATAS(2),DATAS(3),DATAS(4),F(1),F(2),TOLS
%include TOGGLE1.TOL
print# 10,76+(4*(SHW-1)):DATAS(5),DATAS(6),DATAS(7),DATAS(8),F(3),F(4)
%include TOGGLE1.TOL
print# 10,77+(4*(SHW-1)):DATAS(9),DATAS(10),DATAS(11),DATAS(12),F(5),F(6)
%include TOGGLE1.TOL
print# 10,78+(4*(SHW-1)):DATAS(13),DATAS(14),DATAS(15),DATAS(16),F(7),F(8)
gosub 60.00 :gosub 60.00 :dim F(0)
if KEYS=5 then goto 420.5
if KEYS=6 then SHW=SHW-1 :goto 420.1
SHW=SHW+1
if SHW le 6SHW then goto 420.1
420.5 close 10
gosub 61.00
ROUTE=6
chdir "C2C"
REM-----1-----2-----3-----4-----5-----6-----7-----
REM> PRINT REPORT.
1000.0 call fn.DISPCN(MTW,10,37,"PRINTING")
call fn.DISPCN(MTW,17,27,"Press <ESC> to Stop Printing")
print
!printer
print PAGES;
gosub 67.0

ALL%=1
if ROUTE<20 then ROUTE=ROUTE+20 :on ROUTE goto 1410.0, 1420.0
if ROUTE=90 then ALL%=2 :goto 1410.0
if ROUTE ge 51 and ROUTE le 87 then ALL%=3 \
  :SHW=ROUTE-90 :PSH=SHW :goto 1420.0
if ROUTE ge 101 and ROUTE le 120 then ALL%=4 \
  :USH=ROUTE-100 :PSH=SHW+1 :goto 1430.0

```

```

if ROUTE% ge 171 and ROUTE% le 177 then \
:SSM%:=ROUTE%-170 :FSM%:=SSM%-1 :goto 1430.0
if ROUTE% ge 181 and ROUTE% le 187 then \
:SSM%:=ROUTE%-180 :FSM%:=SSM% :goto 1420.0
REM-----2-----3-----4-----5-----6-----7-----
REM> PRINT SCREEN # 418 - CURRENT COUPLING ALIGNMENT.
1410.0 gosub 2410.0
dim DATA$(SM%); F(8)
gosub 68.0
for SSM%:=SSM% to FSM%-1
read 10,51+(4*(SM%-1));DATA$(3),DATA$(4),DATA$(5),DATA$(6),\
DATA$(7),DATA$(8),DATA$(9),DATA$(10)
read 10,52+(4*(SM%-1));DATA$(11),DATA$(12),DATA$(13),DATA$(14),\
DATA$(15),DATA$(16),DATA$(17),DATA$(18)
read 10,53+(4*(SM%-1));DATA$(19),DATA$(20),DATA$(21),DATA$(22),\
DATA$(23),DATA$(24),TOL%
read 10,54+(4*(SM%-1));F(1),F(2),F(3),F(4),F(5),F(6),F(7),F(8)
if SSM%-SSM%+3 or SSM%-SSM%+6 then gosub 68.0
gosub 65.0
print using "1MM.###: #S.###: #R.###: #H.###: #V.###: #D.10;\
chr$(34),F(1),F(2),F(3),F(4),chr$(34)
read 10,10; D1$
print using "1 R1= ##.###: R1= ##.###: #10.10;\
chr$(34),F(5),F(6),F(7),F(8),F(9),F(10),F(11),F(12),chr$(34)
read 10,10; D2$
print using "1 F1= ##.###: F1= ##.###: #10.10;\
chr$(34),F(13),F(14),F(15),F(16),F(17),F(18),F(19),F(20),chr$(34)
read 10,10; D3$
print using "1 BLANK"
SCRS(3)=left$(SCRS(3),88)+TOL%+mid$(SCRS(3),89+len(TOL%),60)
SCRS(4)=left$(SCRS(4),5)+ENG$(SM%)+mid$(SCRS(4),6+len(ENG$(SM%)),80)
SCRS(5)=left$(SCRS(5),28+len(ENG$(SM%+1)))+ENG$(SM%+1)+mid$(SCRS(5),29,82)
SCRS(16)=left$(SCRS(16),38)+D1$
SCRS(18)=left$(SCRS(18),40)+D2$
SCRS(19)=left$(SCRS(19),40)+D3$
SCRS(28)=left$(SCRS(28),27)+\
"The "-ENG$(SM%)+"- "+ENG$(SM%+1)+" (or "+chr$(64+SM%)+") Coupling"
gosub 66.0
print A$T$
gosub 66.00
next SSM%
print B$T$
if FSM%-SSM%-1=0 or FSM%-SSM%-1=3 then J%:=60
if FSM%-SSM%-1=1 or FSM%-SSM%-1=4 then J%:=38
if FSM%-SSM%-1=2 or FSM%-SSM%-1=5 then J%:=18
for I%=1 to J% :print next I%
gosub 60.00 :gosub 60.00 :dim F(0)
if ALL$ ne 2 then goto 1499.1
REM-----2-----3-----4-----5-----6-----7-----
REM> PRINT SCREEN # 420 - CURRENT OIL & BLANK BORE ALIGNMENT.
1420.0 gosub 68.0
for SSM%:=SSM% to FSM%
if left$(ENG$(SM%),3)="GEN" then gosub 2421.0 :goto 1430.1
if left$(ENG$(SM%),3)="EXC" then gosub 2422.0 :goto 1430.1
gosub 2420.0
1420.1 dim DATA$(SM%); F(8)
read 10,75+(4*(SM%-1));DATA$(1),DATA$(2),DATA$(3),DATA$(4),F(1),F(2),TOL%
read 10,76+(4*(SM%-1));DATA$(5),DATA$(6),DATA$(7),DATA$(8),F(3),F(4)
read 10,77+(4*(SM%-1));DATA$(9),DATA$(10),DATA$(11),DATA$(12),F(5),F(6)
read 10,78+(4*(SM%-1));DATA$(13),DATA$(14),DATA$(15),DATA$(16),F(7),F(8)
if SSM%-SSM%+3 or SSM%-SSM%+6 then gosub 68.0
gosub 65.0
print using \
"1MM.###: #S.###: #VERT.###: #H.###: #V.###: #D.10;\
#10.10;chr$(34),F(1),F(2),F(3),F(4),F(5),F(6),F(7),F(8),F(9),F(10),F(11),F(12),chr$(34)
F(13),F(14),F(15),F(16),F(17),F(18),F(19),F(20),chr$(34)

read 10,10; D1$
print using \
"1MM.###: #R.###: #HORIZ.###: #H.###: #V.###: #D.10;\
#10.10;chr$(34),F(21),F(22),F(23),F(24),F(25),F(26),F(27),F(28),F(29),F(30),F(31),F(32),chr$(34)
F(33),F(34),F(35),F(36),F(37),F(38),F(39),F(40),chr$(34)

read 10,10; D2$
print using "1 BLANK"
SCRS(3)=left$(SCRS(3),88)+TOL%+mid$(SCRS(3),89+len(TOL%),60)
SCRS(17)=left$(SCRS(17),12)+D1$
SCRS(19)=left$(SCRS(19),12)+D2$
SCRS(28)=left$(SCRS(28),28)+\
"The "-ENG$(SM%)+"- (or "+chr$(64+SM%)+") Shaft"
gosub 66.0
print A$T$
gosub 66.00
gosub 60.00 :gosub 60.00 :dim F(0)
next SSM%
print B$T$
if FSM%-SSM%-1=0 or FSM%-SSM%-1=3 or FSM%-SSM%-1=6 then J%:=60
if FSM%-SSM%-1=1 or FSM%-SSM%-1=4 then J%:=38
if FSM%-SSM%-1=2 or FSM%-SSM%-1=5 then J%:=18
for I%=1 to J% :print next I%
1499.1 close 10
gosub 61.00
ROUTE%:=7
call fn DISP$(MTN,17,27," Returning to Report Menu ")
chain "CSC"
REM-----2-----3-----4-----5-----6-----7-----
REM>-----2-----3-----4-----5-----6-----7-----
%include SUB00
%include SUB01
%include SUB05
%include SUB06
%include SUB07
%include SUB08
%include SUB09
%include SUB2405
%include SUB2410
%include SUB2420

```

```

REM-----1-----2-----3-----4-----5-----6-----7-----
REM> Copyright (C) COMPAD Incorporated 1984,5,6
REM> ALL RIGHTS RESERVED
REM>
REM> TITLE: C2-TURBINE GENERATOR ALIGNMENT SYSTEM
REM> VERSION: 06.1
REM> REVISION: a)1-15-88
REM> FILE: C2H - CALCULATIONS; READ DATA
REM>-----1-----2-----3-----4-----5-----6-----7-----
REM> SETUP.

#include COMMON

def fn.DISPMN(MTN,ROWN,COLN,BS) external
fend
def fn.CLSN(MTN,CLN) external
fend

call fn.CLSN(1,7)
REM>-----1-----2-----3-----4-----5-----6-----7-----
REM> CALCULATION BEGINS.
REM> START COMMENTS.
call fn.DISPMN(1,1,1,UNIT$)
call fn.DISPMN(1,1,10-len(COMPS),COMPS)
REM> READ NUMBER OF SHAFTS
20.00 open DDS rec1 81 as 10
readN 10,8;NPRS
MPPR=ucase$(NPRS)
dim SMS(7)
readN 10,8;SMS(1),SMS(2),SMS(3),SMS(4),SMS(5),SMS(6),SMS(7)
readN 10,8;SENH,FMH,REFN,DDS,AUTOS
call fn.DISPMN(1,10,"CALCULATIONS BEGIN. Positioning to "+DDS+" Alignment")
SEN=REFN-1
ECN=REFN-1
REM>-----1-----2-----3-----4-----5-----6-----7-----
REM> VARIABLE NAMING CONVENTION.
REM>
REM> 1 2 3 4 5
REM> - - - - -
REM> | | | | | C=Coupling, B=Bearing, O=Oil, S=sealing Shw, G=Gland
REM> | | | | | Q=Governor and, C=Collector and
REM> | | | | | V=Vertical, H=Horizontal
REM> | | | | | Q=Design, C=Current, M=Move, Projected
REM> | | | | | S=relative shaft position, C=shaft position in Casing
REM>-----1-----2-----3-----4-----5-----6-----7-----
REM> INITIALIZE & DIMENSION VARIABLES.
IN=SENH
dim RD(1N,8),CO(1N)
dim XCC(1N),XCB(1N),XCO(1N),XCG(1N)
dim XSC(1N),XSB(1N),XSO(1N),XSG(1N),XT(1N)

dim DVD(1N-1),DND(1N-1),PDVG(1N-1),PDNB(1N-1),DVE(1N-1),DHE(1N-1)
dim SDVCC(1N),SDVCB(1N),SDVCO(1N),SDVCG(1N)
dim SDHCC(1N),SDHCB(1N),SDHCO(1N),SDHCG(1N)
dim SDVGC(1N),SDVGB(1N),SDVGO(1N),SDVGG(1N)
dim SDHGC(1N),SDHGB(1N),SDHGO(1N),SDHGG(1N)

dim CVG(1N-1),CND(1N-1),PCVG(1N-1),PCND(1N-1),CVE(1N-1),CHE(1N-1)
dim SGVCC(1N),SGVCB(1N),SGVCO(1N),SGVCG(1N)
dim SGHCC(1N),SGHCB(1N),SGHCO(1N),SGHCG(1N)
dim SGVGC(1N),SGVGB(1N),SGVGO(1N),SGVGG(1N)
dim SGHGC(1N),SGHGB(1N),SGHGO(1N),SGHGG(1N)

dim PVD(1N-1),PNB(1N-1),PVE(1N-1),PNE(1N-1)

dim NND(1N-1),NHB(1N-1),NVE(1N-1),NHE(1N-1)

dim SHVCC(1N),SHVCB(1N),SHVCO(1N),SHVCG(1N)
dim SHHCC(1N),SHHCB(1N),SHHCO(1N),SHHCG(1N)
dim SHVGC(1N),SHVGB(1N),SHVGO(1N),SHVGG(1N)
dim SHHGC(1N),SHHGB(1N),SHHGO(1N),SHHGG(1N)

dim SLV(1N),SNVGF(1N),SNVCF(1N),SLH(1N),SNHGF(1N),SNHCF(1N)

dim CONCB(1N),CONCO(1N)
dim CONDB(1N),CONDO(1N)
dim COVCB(1N),COVCO(1N)
dim CONGB(1N),CONGO(1N)

dim COVCO(1N),COVCB(1N)
dim CONCB(1N),CONCB(1N)
dim COVGB(1N),COVGB(1N)
dim CONGO(1N),CONGO(1N)

dim CPVCO(1N),CPVCB(1N)
dim CPHCO(1N),CPHCB(1N)
dim CPVGB(1N),CPVGB(1N)
dim CPHGO(1N),CPHGO(1N)

dim CMVCO(1N),CMVCB(1N)
dim CMHCO(1N),CMHCB(1N)
dim CMVGB(1N),CMVGB(1N)
dim CMHGO(1N),CMHGO(1N)

dim CMVGS(1N),CMHGS(1N)
dim CMVGS(1N),CMHGS(1N)
REM>-----1-----2-----3-----4-----5-----6-----7-----
REM> READ DATA.
REM> DISPLAY MESSAGE.
call fn.DISPMN(1,9,15,"READING DATA...")
REM> READ ROTOR GEOMETRY (RD=ROTOR DIMENSION).
dim DATAS(16)
for IN=SENH to FMH
readN 10,10+IN;DATAS(1),DATAS(2),DATAS(3),DATAS(4),\
DATAS(5),DATAS(6),DATAS(7),DATAS(8)
RD(1N,1)=VAL(DATAS(1))
RD(1N,2)=VAL(DATAS(2))
RD(1N,3)=VAL(DATAS(3))
RD(1N,4)=VAL(DATAS(4))
RD(1N,5)=VAL(DATAS(5))
RD(1N,6)=VAL(DATAS(6))
RD(1N,7)=VAL(DATAS(7))
RD(1N,8)=VAL(DATAS(8))
next IN
REM> READ DESIGN/DESIRED COUPLINGS.
for IN=SENH to FMH-1
readN 10,10+IN;DATAS(1),DATAS(2),DATAS(3),DATAS(4),\
DATAS(5),DATAS(6),DATAS(7),DATAS(8)
if UPP="UBSTCO" then goto 20.01
DVE(1N)=VAL(DATAS(1))
DVG(1N)=VAL(DATAS(2))
DHE(1N)=VAL(DATAS(3))
DHG(1N)=VAL(DATAS(7))
goto 20.02

```

```

20.01 if DATAS(2)="" then DVE(1%)=VAL(DATAS(1)) else DVE(1%)=VAL(DATAS(2))
if DATAS(4)="" then DMG(1%)=VAL(DATAS(3)) else DMG(1%)=VAL(DATAS(4))
if DATAS(6)="" then DME(1%)=VAL(DATAS(5)) else DME(1%)=VAL(DATAS(6))
if DATAS(8)="" then DMO(1%)=VAL(DATAS(7)) else DMG(1%)=VAL(DATAS(8))
20.02 next I%
REM> READ DESIGN/DESIRED OIL & GLAND BORES.
for I%=55% to F5%
20.03 read 10.37+(2*(I%-1));DATAS(1),DATAS(2),DATAS(3),DATAS(4),\
      DATAS(5),DATAS(6),DATAS(7),DATAS(8)\
      read 10.38+(2*(I%-1));DATAS(9),DATAS(10),DATAS(11),DATAS(12),\
      DATAS(13),DATAS(14),DATAS(15),DATAS(16)
if DCS="Desired" then goto 20.04
CDVGO(1%)=VAL(DATAS(1))
CDHGO(1%)=VAL(DATAS(3))
CDVGO(1%)=VAL(DATAS(5))
CDHGO(1%)=VAL(DATAS(7))
CDVGO(1%)=VAL(DATAS(9))
CDHGO(1%)=VAL(DATAS(11))
CDVGO(1%)=VAL(DATAS(13))
CDHGO(1%)=VAL(DATAS(15))
goto 20.05
20.04 if DATAS(2)="" then CDVGO(1%)=VAL(DATAS(1)) else CDVGO(1%)=VAL(DATAS(2))
if DATAS(4)="" then CDHGO(1%)=VAL(DATAS(3)) else CDHGO(1%)=VAL(DATAS(4))
if DATAS(6)="" then CDVGO(1%)=VAL(DATAS(5)) else CDVGO(1%)=VAL(DATAS(6))
if DATAS(8)="" then CDHGO(1%)=VAL(DATAS(7)) else CDHGO(1%)=VAL(DATAS(8))
if DATAS(10)="" then CDVGO(1%)=VAL(DATAS(9)) else CDVGO(1%)=VAL(DATAS(10))
if DATAS(12)="" then CDHGO(1%)=VAL(DATAS(11)) else CDHGO(1%)=VAL(DATAS(12))
if DATAS(14)="" then CDVGO(1%)=VAL(DATAS(13)) else CDVGO(1%)=VAL(DATAS(14))
if DATAS(16)="" then CDHGO(1%)=VAL(DATAS(15)) else CDHGO(1%)=VAL(DATAS(16))
20.05 next I%
REM> READ CURRENT COMPLINES.
dim DA(8)
for I%=55% to F5%+1
read 10.64+(4*(I%-1));DA(1),DA(2),DA(3),DA(4),\
      CVE(1%),CHE(1%),CVG(1%),CHG(1%)
next I%
REM> READ CURRENT OIL & GLAND BORES.
for I%=55% to F5%
20.06 read 10.73+(4*(I%-1));D1S,D2S,D3S,D4S,CDVGO(1%),CDHGO(1%)
read 10.76+(4*(I%-1));D1S,D2S,D3S,D4S,CDVGO(1%),CDHGO(1%)
read 10.77+(4*(I%-1));D1S,D2S,D3S,D4S,CDVGO(1%),CDHGO(1%)
read 10.78+(4*(I%-1));D1S,D2S,D3S,D4S,CDVGO(1%),CDHGO(1%)
next I%

for I%=1 to 10
  DATAS(I%)=NULL
next I%
dim DATAS(0), DA(0)
CLOSE IO
DMS=(35%+5%)*(35%+5%)-1)*.32
call fn.DISPM(1,0,33,STR$(DMS)*" READ")
REM-----1-----2-----3-----4-----5-----6-----7-----
%include C2H1

```

```

REM-----1-----2-----3-----4-----5-----6-----7-----
REM> Copyright (C) COMPAD Incorporated 1984, S.S
REM> ALL RIGHTS RESERVED
REM>
REM> TITLE: C2-TURBINE GENERATOR ALIGNMENT SYSTEM
REM> VERSION: 99.1
REM> REVISION: 812-21-89
REM> FILE: C2H1 - CALCULATIONS: CALCULATE
REM>
REM-----1-----2-----3-----4-----5-----6-----7-----
REM> REFERENCE SHAFT CALCULATIONS.
20.10 I%=REF%
REM> CHECK AUTO POSITIONING.
if AUTOS="01" then goto 20.11
if AUTOS="01and" then goto 20.12
goto 20.20
REM> AUTO POSITION USING OIL BORES.
REM> DISPLAY MESSAGE.
20.11 call fn.DISPM(1,11,16,"REFERENCE SHAFT CALCULATIONS....")
call fn.DISPM(1,12,16,"AUTO POSITION USING OIL BORES")
REM> CALCULATE MOVES AT THE OIL BORES.
SMHGO(1%)=CDHGO(1%)-CDHGO(1%)
SMVGO(1%)=CDVGO(1%)-CDVGO(1%)
SMHGO(1%)=CDHGO(1%)-CDHGO(1%)
SMVGO(1%)=CDVGO(1%)-CDVGO(1%)
REM> CALCULATE X POSITIONS RELATIVE TO GOVERNOR END OIL BORE (GO).
XGO(1%)=0
if left$(BMS(1%),3)="DEM" or left$(BMS(1%),3)="EXC" then goto 20.111
XGO(1%)=-RO(1%,2)-RO(1%,2)
XGO(1%)=-RO(1%,5)
XGO(1%)=-RO(1%,5)+RO(1%,7)
XGO(1%)=-RO(1%,5)+RO(1%,3)+RO(1%,8)
XGO(1%)=-RO(1%,5)+RO(1%,3)+RO(1%,8)
XGO(1%)=-RO(1%,5)+RO(1%,3)
XGO(1%)=-RO(1%,5)+RO(1%,3)+RO(1%,4)
goto 20.112
20.111 XGO(1%)=-RO(1%,5)-RO(1%,2)
XGO(1%)=-RO(1%,5)
XGO(1%)=-RO(1%,5)+RO(1%,7)
XGO(1%)=-RO(1%,5)+RO(1%,3)+RO(1%,8)
XGO(1%)=-RO(1%,5)+RO(1%,3)+RO(1%,8)
XGO(1%)=-RO(1%,5)+RO(1%,3)
XGO(1%)=-RO(1%,5)+RO(1%,3)+RO(1%,4)
20.112 REM> CALCULATE HORIZONTAL MOVES.
if XGO(1%)=0 then SLH(1%)=0 else SLH(1%)=(SMHGO(1%)-SMHGO(1%))/XGO(1%)
SMHCC(1%)=(SLH(1%)*XCC(1%)+SMHGO(1%))
SMHCB(1%)=(SLH(1%)*XCB(1%)+SMHGO(1%))
SMHCC(1%)=(SLH(1%)*XCC(1%)+SMHGO(1%))
SMHCB(1%)=(SLH(1%)*XCB(1%)+SMHGO(1%))
SMHCC(1%)=(SLH(1%)*XCC(1%)+SMHGO(1%))
SMHCB(1%)=(SLH(1%)*XCB(1%)+SMHGO(1%))
SMHCC(1%)=(SLH(1%)*XCC(1%)+SMHGO(1%))
SMHCB(1%)=(SLH(1%)*XCB(1%)+SMHGO(1%))
REM> CALCULATE VERTICAL MOVES.
if XGO(1%)=0 then SLV(1%)=0 else SLV(1%)=(SMVGO(1%)-SMVGO(1%))/XGO(1%)
SMVCC(1%)=(SLV(1%)*XCC(1%)+SMVGO(1%))
SMVCB(1%)=(SLV(1%)*XCB(1%)+SMVGO(1%))
SMVCC(1%)=(SLV(1%)*XCC(1%)+SMVGO(1%))
SMVCB(1%)=(SLV(1%)*XCB(1%)+SMVGO(1%))
SMVCC(1%)=(SLV(1%)*XCC(1%)+SMVGO(1%))
SMVCB(1%)=(SLV(1%)*XCB(1%)+SMVGO(1%))
SMVCC(1%)=(SLV(1%)*XCC(1%)+SMVGO(1%))
SMVCB(1%)=(SLV(1%)*XCB(1%)+SMVGO(1%))

```

```

call fn.DISPM(1,11,49," 26 COMPLETED")
CNTN=26
goto 20.28
REM> AUTO POSITION USING GLAND BORE.
REM> DISPLAY MESSAGE.
20.12 call fn.DISPM(1,11,16,"REFERENCE SHAFT CALCULATIONS....")
call fn.DISPM(1,12,16,"AUTO POSITION USING GLAND BORE")
REM> CALCULATE MOVES AT THE GLAND BORES.
SMHGG(1%)+CDHGG(1%)-CCHGG(1%)
SMVGG(1%)+CDVGG(1%)-CCHGG(1%)
SMHCG(1%)+CDHCG(1%)-CCHCG(1%)
SMVCG(1%)+CDVCG(1%)-CCHCG(1%)
REM> CALCULATE X POSITIONS RELATIVE TO GOVERNOR END GLAND BORE (00).
XCG(1%)=-RD(1%,7)-RD(1%,2)
XGB(1%)=-RD(1%,7)
XGG(1%)=0
XCG(1%)=-RD(1%,7)+RD(1%,3)-RD(1%,8)
XCB(1%)=-RD(1%,7)+RD(1%,3)
XCG(1%)=-RD(1%,7)+RD(1%,3)+RD(1%,4)
if left(SNS(1%),3)="GEN" or left(SNS(1%),3)="EXC" \
then XCG(1%)=-RD(1%,7)-RD(1%,2) : XCB(1%)=-RD(1%,7)-RD(1%,3)+RD(1%,8) \
else XCG(1%)=-RD(1%,7)+RD(1%,3) : XCB(1%)=-RD(1%,7)-RD(1%,3)-RD(1%,8)
REM> CALCULATE HORIZONTAL MOVES.
if XCG(1%)=0 then SLH(1%)=0 else SLH(1%)=(SMHCG(1%)-SMHGG(1%))/XCG(1%)
SMHCC(1%)=(SLH(1%)*XCG(1%))+SMHGG(1%)
SMHCB(1%)=(SLH(1%)*XCB(1%))+SMHGG(1%)
SMHCD(1%)=(SLH(1%)*XCG(1%))+SMHGG(1%)
SMHGD(1%)=(SLH(1%)*XCG(1%))+SMHGG(1%)
SMHGB(1%)=(SLH(1%)*XGB(1%))+SMHGG(1%)
SMHGC(1%)=(SLH(1%)*XGC(1%))+SMHGG(1%)
REM> CALCULATE VERTICAL MOVES.
if XCG(1%)=0 then SLV(1%)=0 else SLV(1%)=(SMVCG(1%)-SMVGG(1%))/XCG(1%)
SMVCC(1%)=(SLV(1%)*XCG(1%))+SMVGG(1%)
SMVCB(1%)=(SLV(1%)*XCB(1%))+SMVGG(1%)
SMVCD(1%)=(SLV(1%)*XCG(1%))+SMVGG(1%)
SMVGD(1%)=(SLV(1%)*XCG(1%))+SMVGG(1%)
SMVGB(1%)=(SLV(1%)*XGB(1%))+SMVGG(1%)
SMVGC(1%)=(SLV(1%)*XGC(1%))+SMVGG(1%)
call fn.DISPM(1,11,49," 26 COMPLETED")
CNTN=26
REM>-----1-----2-----3-----4-----5-----6-----7-----
REM> GOVERNOR SIDE CALCULATIONS.
REM> CHECK IF REFERENCE SHAFT IS FIRST SHAFT.
20.20 if REP% LE SSN then goto 20.30
REM> DISPLAY MESSAGE.
call fn.DISPM(1,14,16,"GOVERNOR SIDE CALCULATIONS.....")
REM> CALCULATE X DIMENSIONS & COUPLING DIAMETER.
for 1%=50% to 56% STEP -1
CD(1%)=RD(1%,1)
XCB(1%)=RD(1%,4)
if left(SNS(1%),3)="GEN" or left(SNS(1%),3)="EXC" \
then XCG(1%)=RD(1%,4)-RD(1%,8) \
else XCG(1%)=RD(1%,4)-RD(1%,8)
XCG(1%)=RD(1%,4)-RD(1%,8)
XGG(1%)=RD(1%,4)-RD(1%,3)-RD(1%,7)
if left(SNS(1%),3)="GEN" or left(SNS(1%),3)="EXC" \
then XCG(1%)=RD(1%,4)-RD(1%,3)-RD(1%,5) \
else XCG(1%)=RD(1%,4)-RD(1%,3)-RD(1%,5)
XGB(1%)=RD(1%,4)-RD(1%,3)
XT(1%)=RD(1%,4)-RD(1%,3)-RD(1%,2)
next 1%
REM> CALCULATE DESIGN VERTICAL & HORIZONTAL POSITIONS.
for 1%=50% to 56% STEP -1
if 1% LE 50% then goto 20.21
SDVCC(1%)=DVE(1%) : SDHCC(1%)=DHE(1%)
if CD(1%)=0 then PDVG(1%)=0 : PDHG(1%)=0 \
else PDVG(1%)=DVG(1%)/CD(1%) : PDHG(1%)=DHG(1%)/CD(1%)
20.21 SDVCC(1%)=SDVCC(1%+1)+DVE(1%) : SDHCC(1%)=SDHCC(1%+1)+DHE(1%)
if CD(1%)=0 then PDVG(1%)=PDVG(1%+1) : PDHG(1%)=PDHG(1%+1) \
else PDVG(1%)=DVG(1%)/CD(1%) : PDHG(1%)=DHG(1%)/CD(1%)
20.22 SDVCB(1%)=PDVG(1%)*XCB(1%)+SDVCC(1%)
SDVCD(1%)=PDVG(1%)*XCG(1%)+SDVCC(1%)
SDVGD(1%)=PDVG(1%)*XCG(1%)+SDVCC(1%)
SDVGB(1%)=PDVG(1%)*XGB(1%)+SDVCC(1%)
SDVGC(1%)=PDVG(1%)*XGC(1%)+SDVCC(1%)
SDVCC(1%)=PDVG(1%)*XT(1%)+SDVCC(1%)
SDHCB(1%)=PDHG(1%)*XCB(1%)+SDHCC(1%)
SDHCD(1%)=PDHG(1%)*XCG(1%)+SDHCC(1%)
SDHGD(1%)=PDHG(1%)*XCG(1%)+SDHCC(1%)
SDHGB(1%)=PDHG(1%)*XGB(1%)+SDHCC(1%)
SDHGC(1%)=PDHG(1%)*XGC(1%)+SDHCC(1%)
SDHCC(1%)=PDHG(1%)*XT(1%)+SDHCC(1%)
next 1%
REM> CALCULATE CURRENT VERTICAL & HORIZONTAL POSITIONS.
for 1%=50% to 56% STEP -1
if 1% LE 50% then goto 20.23
SCVCC(1%)=CVE(1%)-SMVCC(1%+1) : SCHCC(1%)=CHE(1%)-SMHCC(1%+1)
if CD(1%)=0 then PCVG(1%)=SLV(1%+1) : PCHG(1%)=SLH(1%+1) \
else PCVG(1%)=CVG(1%)/CD(1%) : PCHG(1%)=CHG(1%)/CD(1%)
20.23 SCVCC(1%)=SCVCC(1%+1)+CVE(1%) : SCHCC(1%)=SCHCC(1%+1)+CHE(1%)
if CD(1%)=0 then PCVG(1%)=PCVG(1%+1) : PCHG(1%)=PCHG(1%+1) \
else PCVG(1%)=CVG(1%)/CD(1%) : PCHG(1%)=CHG(1%)/CD(1%)
20.24 SCVCB(1%)=PCVG(1%)*XCB(1%)+SCVCC(1%)
SCVCD(1%)=PCVG(1%)*XCG(1%)+SCVCC(1%)
SCVGD(1%)=PCVG(1%)*XCG(1%)+SCVCC(1%)
SCVGB(1%)=PCVG(1%)*XGB(1%)+SCVCC(1%)
SCVGC(1%)=PCVG(1%)*XGC(1%)+SCVCC(1%)
SCVCC(1%)=PCVG(1%)*XT(1%)+SCVCC(1%)
SCHCB(1%)=PCHG(1%)*XCB(1%)+SCHCC(1%)
SCHCD(1%)=PCHG(1%)*XCG(1%)+SCHCC(1%)
SCHGD(1%)=PCHG(1%)*XCG(1%)+SCHCC(1%)
SCHGB(1%)=PCHG(1%)*XGB(1%)+SCHCC(1%)
SCHGC(1%)=PCHG(1%)*XGC(1%)+SCHCC(1%)
SCHCC(1%)=PCHG(1%)*XT(1%)+SCHCC(1%)
next 1%
CHN=30*(SSN-SSN+1)
if CHN=100 then call fn.DISPM(1,14,49," ->3rd(CN)= COMPLETED" ) \
else call fn.DISPM(1,14,49," ->2nd(CN)= COMPLETED" )
CNTN=CNTN+CHN
CHN=0
REM>-----1-----2-----3-----4-----5-----6-----7-----
REM> COLLECTOR SIDE CALCULATIONS.
REM> CHECK IF REFERENCE SHAFT IS THE LAST SHAFT.
20.30 if REP% GE FSN then goto 20.35
REM> DISPLAY MESSAGE.
call fn.DISPM(1,12,16,"COLLECTOR SIDE CALCULATIONS.....")

```

```

REM> CALCULATE X DIMENSIONS & COMPLETING DIAMETER.
CD(BCN-1)=RD(SCN-1,3)
for I%=SCN to FSHM
CD(I%)+RD(I%,2)
XSD(I%)+RD(I%,2)
if left$(SNS(I%),3)="QEM" or left$(SNS(I%),3)="EXC" \
then XCO(I%)=RD(I%,2)-RD(I%,6) \
else XCO(I%)=RD(I%,2)+RD(I%,6)
XSD(I%)+RD(I%,2)+RD(I%,6)
XCO(I%)+RD(I%,2)+RD(I%,6)-RD(I%,6)
if left$(SNS(I%),3)="QEM" or left$(SNS(I%),3)="EXC" \
then XCO(I%)=RD(I%,2)+RD(I%,6)+RD(I%,6) \
else XCO(I%)=RD(I%,2)+RD(I%,6)-RD(I%,6)
XCS(I%)+RD(I%,2)+RD(I%,6)
XT(I%)+RD(I%,2)+RD(I%,6)+RD(I%,4)
next I%

REM> CALCULATE DESIGN VERTICAL & HORIZONTAL POSITIONS.
for I%=SCN to FSHM
if I% gt SCN then 20.31
SOVGC(I%)=-DVE(I%-1)/SOMGC(I%)-DHE(I%-1)
if CD(I%-1)=0 then PDVG(I%-1)=0 :PDHG(I%-1)=0 \
else PDVG(I%-1)=(DVE(I%-1)/CD(I%-1)) :PDHG(I%-1)=(DHG(I%-1)/CD(I%-1))
goto 20.32
20.31 SOVGC(I%)=SOVGC(I%-1)-DVE(I%-1) :SDHGC(I%)=SDHGC(I%-1)-DHE(I%-1)
if CD(I%-1)=0 then PDVG(I%-1)=PDVG(I%-2) :PDHG(I%-1)=PDHG(I%-2) \
else PDVG(I%-1)=(DVE(I%-1)/CD(I%-1))+PDVG(I%-2) \
:PDHG(I%-1)=(DHG(I%-1)/CD(I%-1))+PDHG(I%-2)
20.32 SOVDB(I%)=PDVG(I%-1)*XGB(I%)+SOVGC(I%)
SOVGD(I%)=PDVG(I%-1)*XGD(I%)+SOVGC(I%)
SOVSB(I%)=PDVG(I%-1)*XGB(I%)+SOVGC(I%)
SOVSD(I%)=PDVG(I%-1)*XGD(I%)+SOVGC(I%)
SOVCB(I%)=PDVG(I%-1)*XCB(I%)+SOVGC(I%)
SOVCC(I%)=PDVG(I%-1)*XT(I%)+SOVGC(I%)
SDHDB(I%)=PDHG(I%-1)*XGB(I%)+SDHGC(I%)
SDHDG(I%)=PDHG(I%-1)*XGD(I%)+SDHGC(I%)
SDHGB(I%)=PDHG(I%-1)*XGB(I%)+SDHGC(I%)
SDHGD(I%)=PDHG(I%-1)*XGD(I%)+SDHGC(I%)
SDHCB(I%)=PDHG(I%-1)*XCB(I%)+SDHGC(I%)
SDHCC(I%)=PDHG(I%-1)*XT(I%)+SDHGC(I%)
next I%

REM> CALCULATE CURRENT VERTICAL & HORIZONTAL POSITIONS.
for I%=SCN to FSHM
if I% gt SCN then goto 20.33
SCVCE(I%)=-CVE(I%-1)-SMVCC(I%-1)
SCHCE(I%)=-CNE(I%-1)-SMHCC(I%-1)
if CD(I%-1)=0 then PCVG(I%-1)=SLV(I%-1) :PCNG(I%-1)=SLH(I%-1) \
else PCVG(I%-1)=(CVE(I%-1)/CD(I%-1))-SLV(I%-1) \
:PCNG(I%-1)=(CNG(I%-1)/CD(I%-1))-SLH(I%-1)
goto 20.34
20.33 SCVCE(I%)=SCVCE(I%-1)-CVE(I%-1) :SCHCE(I%)=SCHCE(I%-1)-CNE(I%-1)
if CD(I%-1)=0 then PCVG(I%-1)=PCVG(I%-2) :PCNG(I%-1)=PCNG(I%-2) \
else PCVG(I%-1)=(CVE(I%-1)/CD(I%-1))+PCVG(I%-2) \
:PCNG(I%-1)=(CNG(I%-1)/CD(I%-1))+PCNG(I%-2)
20.34 SCVDB(I%)=PCVG(I%-1)*XGB(I%)+SCVCE(I%)
SCVDG(I%)=PCVG(I%-1)*XGD(I%)+SCVCE(I%)
SCVSB(I%)=PCVG(I%-1)*XGB(I%)+SCVCE(I%)
SCVSD(I%)=PCVG(I%-1)*XGD(I%)+SCVCE(I%)
SCVCB(I%)=PCVG(I%-1)*XCB(I%)+SCVCE(I%)
SCVCC(I%)=PCVG(I%-1)*XT(I%)+SCVCE(I%)
SCHDB(I%)=PCNG(I%-1)*XGB(I%)+SCHCE(I%)
SCHDG(I%)=PCNG(I%-1)*XGD(I%)+SCHCE(I%)
SCHGB(I%)=PCNG(I%-1)*XGB(I%)+SCHCE(I%)
SCHGD(I%)=PCNG(I%-1)*XGD(I%)+SCHCE(I%)
SCHCB(I%)=PCNG(I%-1)*XCB(I%)+SCHCE(I%)
SCHCC(I%)=PCNG(I%-1)*XT(I%)+SCHCE(I%)
next I%
CNH=DE*(FSHM-SCN+1)
if CNH=0 then call fn.DISPM(1,18,49,"*-str$(CNH)-" COMPLETED") \
else call fn.DISPM(1,18,49,str$(CNH)+" COMPLETED")
CNT=CNT-CN#

REM-----1-----2-----3-----4-----5-----6-----7-----8-----9-----
REM> CALCULATE VERTICAL & HORIZONTAL MOVES.
20.35 call fn.DISPM(1,18,16,"COMPLETING CALCULATIONS.....")
cmt=0
for I%=SCN to FSHM
if I%=REFN then goto 20.361
SMVCC(I%)=SDVCC(I%) :SCVCC(I%)
SMVDS(I%)=SDVDS(I%) :SCVDS(I%)
SMVDC(I%)=SDVDC(I%) :SCVDC(I%)
SMVGS(I%)=SDVGS(I%) :SCVGS(I%)
SMVCS(I%)=SDVCS(I%) :SCVCS(I%)
SMVCC(I%)=SDVCC(I%) :SCVCC(I%)
SMHCC(I%)=DMHCC(I%) :SMHCC(I%)
SMHGS(I%)=DMHGS(I%) :SMHGS(I%)
SMHDS(I%)=DMHDS(I%) :SMHDS(I%)
SMHCS(I%)=DMHCS(I%) :SMHCS(I%)
SMHCC(I%)=DMHCC(I%) :SMHCC(I%)
if RD(I%,2)+RD(I%,3)+RD(I%,4)=0 then SLV(I%)=0 :SLH(I%)=0 \
else SLV(I%)=(SMVCC(I%)-SMVCC(I%))/(RD(I%,2)+RD(I%,3)+RD(I%,4)) \
:SLH(I%)=(SMHCC(I%)-SMHCC(I%))/(RD(I%,2)+RD(I%,3)+RD(I%,4))
20.351 SMVCF(I%)=SLV(I%)+CD(I%)
SMHCF(I%)=SLH(I%)+CD(I%)
DN=CNT-28 :call fn.DISPM(1,18,49,str$(DN))
if I%=FSHM then goto 20.382
SMVGF(I%)=SLV(I%)+CD(I%-1)
SMHGF(I%)=SLH(I%)+CD(I%-1)
PVE(I%-1)=SMVCC(I%-1)-SMVCC(I%)+CVE(I%-1)
NVE(I%-1)=PVE(I%-1)-DVE(I%-1)
PME(I%-1)=SMHCC(I%-1)-SMHCC(I%)+CNE(I%-1)
NME(I%-1)=PME(I%-1)-DHE(I%-1)
PVG(I%-1)=SMVGF(I%) -SMVGF(I%-1)+CVG(I%-1)
AVG(I%-1)=PVG(I%-1)-DVG(I%-1)
PHG(I%-1)=SMHGF(I%) -SMHGF(I%-1)+CHG(I%-1)
AHG(I%-1)=PHG(I%-1)-DHG(I%-1)
DN=CNT-10 :call fn.DISPM(1,18,49,str$(DN))

20.352 if left$(SNS(I%),3)="QEM" then goto 20.382
if left$(SNS(I%),3)="EXC" then goto 20.382

REM> CALCULATE OIL & GLAND BORE VALUES, HP,IP,LP.
CPHCQ(I%)=CEHCQ(I%) :SMHCQ(I%)
CPVCG(I%)=CEVCG(I%) :SMVCG(I%)
CPHCG(I%)=CEPHCG(I%) :CMHCG(I%)
SMVCO(I%)=CEVCO(I%) :LMVCO(I%)
CPHCB(I%)=CEPHCB(I%) :SMHCB(I%)
CPVCO(I%)=CEVCO(I%) :LMVCO(I%)

```



```
dim PVG(1%+1), PHO(1%+1), PVE(1%+1), PWE(1%+1)
dim SNG(1%+1), SHH(1%+1), NNE(1%+1), NWE(1%+1)
```

```

dim SHVCG(1%), SHVCG(1%), SHVCG(1%), SHVCG(1%)
dim SHVCG(1%), SHVCG(1%), SHVCG(1%), SHVCG(1%)
dim SHVCG(1%), SHVCG(1%), SHVCG(1%), SHVCG(1%)
dim SHVCG(1%), SHVCG(1%), SHVCG(1%), SHVCG(1%)

dim SLV(1%), SHVCP(1%), SHVCP(1%), SLH(1%), SHVCP(1%), SHVCP(1%)

dim CDHCG(1%), CDHCG(1%)
dim CDHCG(1%), CDHCG(1%)
dim CDHCG(1%), CDHCG(1%)
dim CDHCG(1%), CDHCG(1%)

dim CCHCG(1%), CCHCG(1%)
dim CCHCG(1%), CCHCG(1%)
dim CCHCG(1%), CCHCG(1%)
dim CCHCG(1%), CCHCG(1%)

dim CPVCG(1%), CPVCG(1%)
dim CPVCG(1%), CPVCG(1%)
dim CPVCG(1%), CPVCG(1%)
dim CPVCG(1%), CPVCG(1%)

dim CMHCG(1%), CMHCG(1%)
dim CMHCG(1%), CMHCG(1%)
dim CMHCG(1%), CMHCG(1%)
dim CMHCG(1%), CMHCG(1%)

REM-----1-----2-----3-----4-----5-----6-----7-----
REM> READ DATA.
dim DATAS(10)
REM> READ ROTOR GEOMETRY.
for I%=50% to 95%
read# 10, 10+I%; DATAS(1), DATAS(2), DATAS(3), DATAS(4), \
DATAS(5), DATAS(6), DATAS(7), DATAS(8)
RD(1%,1)=VAL(DATAS(1))
RD(1%,2)=VAL(DATAS(2))
RD(1%,3)=VAL(DATAS(3))
RD(1%,4)=VAL(DATAS(4))
RD(1%,5)=VAL(DATAS(5))
RD(1%,6)=VAL(DATAS(6))
RD(1%,7)=VAL(DATAS(7))
RD(1%,8)=VAL(DATAS(8))
next I%
REM> READ DESIGN COUPLINGS.
for I%=50% to 95%
read# 10, 30+I%; DATAS(1), DATAS(2), DATAS(3), DATAS(4), \
DATAS(5), DATAS(6), DATAS(7), DATAS(8)
DVE(1%)=VAL(DATAS(1))
DVG(1%)=VAL(DATAS(2))
DHE(1%)=VAL(DATAS(3))
DHG(1%)=VAL(DATAS(4))
next I%
REM> READ DESIGN OIL & GLAND BORES.
for I%=50% to 95%
read# 10, 37+(2*(I%-1)); DATAS(1), DATAS(2), DATAS(3), DATAS(4), \
DATAS(5), DATAS(6), DATAS(7), DATAS(8)
read# 10, 38+(2*(I%-1)); DATAS(9), DATAS(10), DATAS(11), DATAS(12), \
DATAS(13), DATAS(14), DATAS(15), DATAS(16)
COVCG(1%)=VAL(DATAS(1))
CDHCG(1%)=VAL(DATAS(2))
CCHCG(1%)=VAL(DATAS(3))
CMHCG(1%)=VAL(DATAS(4))
CCHCG(1%)=VAL(DATAS(5))
CDHCG(1%)=VAL(DATAS(6))
CCHCG(1%)=VAL(DATAS(7))
CDHCG(1%)=VAL(DATAS(8))
CDHCG(1%)=VAL(DATAS(9))
CDHCG(1%)=VAL(DATAS(10))
CDHCG(1%)=VAL(DATAS(11))
CDHCG(1%)=VAL(DATAS(12))
CDHCG(1%)=VAL(DATAS(13))
CDHCG(1%)=VAL(DATAS(14))
CDHCG(1%)=VAL(DATAS(15))
CDHCG(1%)=VAL(DATAS(16))
next I%

REM> READ CURRENT COUPLINGS.
dim DA(8)
for I%=50% to 95%
read# 10, 54+(4*(I%-1)); DA(1), DA(2), DA(3), DA(4), \
CVE(1%), CHE(1%), CVO(1%), CHB(1%)
next I%
REM> READ CURRENT OIL & GLAND BORES.
for I%=50% to 95%
read# 10, 78+(4*(I%-1)); DATAS(1), DATAS(2), DATAS(3), DATAS(4), COVCG(1%), CCHCG(1%)
read# 10, 79+(4*(I%-1)); DATAS(1), DATAS(2), DATAS(3), DATAS(4), COVCG(1%), CCHCG(1%)
read# 10, 77+(4*(I%-1)); DATAS(1), DATAS(2), DATAS(3), DATAS(4), COVCG(1%), CCHCG(1%)
read# 10, 78+(4*(I%-1)); DATAS(1), DATAS(2), DATAS(3), DATAS(4), COVCG(1%), CCHCG(1%)
next I%

for I%=1 to 76
DATAS(1%)=NULL
next I%
dim DATAS(8), DA(8)
REM>-----1-----2-----3-----4-----5-----6-----7-----
REM> CALCULATE X POSITIONS RELATIVE TO OVERHUNG END BEARING (OS).
for SH%=50% to 95%
CD(SH%)=RD(SH%,1)
XGO(SH%)=RD(SH%,5)
XGO(SH%)=RD(SH%,7)
XGO(SH%)=RD(SH%,3)-RD(SH%,8)
XGO(SH%)=RD(SH%,3)-RD(SH%,8)
XGO(SH%)=RD(SH%,3)
XGO(SH%)=RD(SH%,3)
XGO(SH%)=RD(SH%,3)+RD(SH%,4)
next SH%
REM>-----1-----2-----3-----4-----5-----6-----7-----
REM> SET PROJECTED VALUES EQUAL TO CURRENT VALUES.
REM> VERTICAL BORES
CPVCG(SH%)=COVCG(SH%)
CPVCG(SH%)=COVCG(SH%)
CPVCG(SH%)=CCHCG(SH%)
CPVCG(SH%)=CCHCG(SH%)
REM> HORIZONTAL BORES
CPHCG(SH%)=CDHCG(SH%)
CPHCG(SH%)=CDHCG(SH%)
CPHCG(SH%)=CCHCG(SH%)
CPHCG(SH%)=CCHCG(SH%)
next SH%
for SH%=50% to 95%
REM> VERTICAL RIM & FACE
PVE(SH%)=CVE(SH%)
PVG(SH%)=CVO(SH%)
REM> HORIZONTAL RIM & FACE
PHE(SH%)=CHE(SH%)
PHO(SH%)=CHO(SH%)
next SH%
dim RD(8)
REM>-----1-----2-----3-----4-----5-----6-----7-----
REM> DISPLAY SCREEN # 610 - SHAFT POSITIONER.
goend 2010.0
dim DATAS(8%)
REM> INITIALIZE RESULTS.
for SH%=50% to 95%
read# 10, 102-SH%; DATAS(1), DATAS(2), DATAS(3), DATAS(4), BORES

```

```

if BOREX="" then BOREX="G"
if left$(SNG(SHM),3)="DEN" or left$(SNG(SHM),3)="EXC" then BOREX=""
SMVGB(SHM)=val(DATAS(1))
SMVGB(SHM)=val(DATAS(2))
SMHGB(SHM)=val(DATAS(3))
SMHGB(SHM)=val(DATAS(4))
NGN=1
gosub 810.82
gosub 810.85
gosub 810.81
gosub 810.84
next SHM

NGN=0
SHM=SSHM
810.1 readn 10,102-SHM:DATAS(1),DATAS(2),DATAS(3),DATAS(4),BORES
if BOREX="" then BOREX="G"
gosub 810.80
if SHM no SHM \
then call fn.DISPMN(MTN,12,13-(len(SNG(SHM-1))-1)/2,SNG(SHM-1))
call fn.DISPMN(MTN,12,41-(len(SNG(SHM))-1)/2,SNG(SHM))
if SHM no SHM \
then call fn.DISPMN(MTN,12,89-(len(SNG(SHM-1))-1)/2,SNG(SHM-1))

if left$(SNG(SHM),3)="DEN" or left$(SNG(SHM),3)="EXC" \
then goto 810.2
call fn.DISPCM(MTN,CLN,4,17,"C=Coupling, X=Bearing, G=Gland Bore, D=Dil Bore")
BRCG="X" : FBRCG=" " : str$(2+SNG-1)+" " : BBRCG=" " : str$(2+SNG)+" "
goto 810.2
810.2 call fn.DISPCM(MTN,CLN,4,50,"C=Coupling, X=Bearing")
BRCG="X" : FBRCG=" " : BBRCG=" " : BOREX=""

810.21 call fn.DISPCM(MTN,CLN,12,31,BRCG)
call fn.DISPCM(MTN,CLN,14,50,FBRCG)
call fn.DISPCM(MTN,CLN,13,50,BBRCG)
call fn.DISPCM(MTN,CLN,14,49,BBRCG)
call fn.DISPCM(MTN,CLN,24,6,USER$=" enter moves at Bearings. Press Return to Calculate.")
gosub 810.86

gosub 80.3
SRN=1
SMVGB(SRN)=val(DATAS(1))
SMVGB(SRN)=val(DATAS(2))
SMHGB(SRN)=val(DATAS(3))
SMHGB(SRN)=val(DATAS(4))
call fn.DISPCM(MTN,CLN,BRN(1),BCN(1)+BMAX(1)+1,sn(SMVGB(SRN),"D","U"))
call fn.DISPCM(MTN,CLN,BRN(2),BCN(2)+BMAX(2)+1,sn(SMVGB(SRN),"D","U"))
call fn.DISPCM(MTN,CLN,BRN(3),BCN(3)+BMAX(3)+1,sn(SMHGB(SRN),"R","L"))
call fn.DISPCM(MTN,CLN,BRN(4),BCN(4)+BMAX(4)+1,sn(SMHGB(SRN),"R","L"))
gosub 810.82
gosub 810.85
gosub 810.81
gosub 810.84

810.3 if SRN > 0 then SRN=1
NEWDATAS=fn.INPUT3(MTN,CLN,BRN(SRN),BCN(SRN),DATAS(SRN),1,BMIN(SRN),\
BMAX(SRN),STYPES(SRN),IFS,22,EQUALS)
if mid$(NEWDATAS,3,2)="" \
then KEY=int(val(left$(NEWDATAS,2)) \
:NEWDATAS=mid$(NEWDATAS,5,BMAX(SRN))

if KEY=8 then BV=val(NEWDATAS)+0.0005 :goto 810.31
if KEY=9 then BV=val(NEWDATAS)-0.0005 :goto 810.31
if NEWDATAS=DATAS(SRN) then goto 810.36
BV=val(NEWDATAS)

810.21 DATAS(SRN)=fn.RC8(BV,"###.#####")
datn=fn.BCWN(MTN,BRN(SRN),BCN(SRN),DATAS(SRN),BMAX(SRN))
SMVGB(SRN)=val(DATAS(1))
SMVGB(SRN)=val(DATAS(2))
SMHGB(SRN)=val(DATAS(3))
SMHGB(SRN)=val(DATAS(4))
call fn.DISPCM(MTN,CLN,BRN(1),BCN(1)+BMAX(1)+1,sn(SMVGB(SRN),"D","U"))
call fn.DISPCM(MTN,CLN,BRN(2),BCN(2)+BMAX(2)+1,sn(SMVGB(SRN),"D","U"))
call fn.DISPCM(MTN,CLN,BRN(3),BCN(3)+BMAX(3)+1,sn(SMHGB(SRN),"R","L"))
call fn.DISPCM(MTN,CLN,BRN(4),BCN(4)+BMAX(4)+1,sn(SMHGB(SRN),"R","L"))

if SRN=1 or SRN=2 \
then gosub 810.82 :gosub 810.85 \
else gosub 810.91 :gosub 810.94

810.23 if KEY=1 then SRN=SRN-1 :goto 810.3
if KEY=2 then SRN=SRN+1 :goto 810.30
if KEY=3 then goto 810.4
if KEY=4 then goto 810.4
if KEY=5 then goto 810.4
if KEY=6 \
then if BOREX="G" \
then BOREX="D" :gosub 810.80 :gosub 810.85 :gosub 810.94 :goto 810.3 \
else if BOREX="D" \
then BOREX="G" :gosub 810.80 :gosub 810.85 :gosub 810.94 :goto 810.3 \
else goto 810.3

if KEY=7 then goto 810.4
810.39 if SRN < 888 then SRN=SSHM
goto 810.3

810.4 rem
printn 10,102-SHM:DATAS(1),DATAS(2),DATAS(3),DATAS(4),BORES
if KEY=3 or KEY=7 then goto 810.5
if KEY=4 \
then if SHM=SSHM \
then goto 810.3 \
else SHM=SSHM-1 :goto 810.1
if SHM=SSHM then goto 810.3 else SHM=SSHM+1 :goto 810.1
810.5 close io
if KEY=7 then goto 810.99
ROUTE=6
chain "CIC"

REM-----1-----2-----3-----4-----5-----6-----7-----
REM-----1-----2-----3-----4-----5-----6-----7-----
REM> DISPLAY TEMPLATE = BORES
810.80 if BOREX="" then goto 810.81
if BOREX="G" \
then OBS=" " : GBS="G" : FBORBS="Gland" : BBORBS="Gland" \
else OBS="D" : GBS=" " : FBORBS=" " : BBORBS="Dil "
call fn.DISPCM(MTN,CLN,8,8,FBORBS)
call fn.DISPCM(MTN,CLN,8,71,BBORBS)
call fn.DISPCM(MTN,CLN,20,8,FBORBS)
call fn.DISPCM(MTN,CLN,20,71,BBORBS)
call fn.DISPCM(MTN,CLN,13,33,GBS)
call fn.DISPCM(MTN,CLN,13,48,GBS)
call fn.DISPCM(MTN,CLN,13,46,GBS)
call fn.DISPCM(MTN,CLN,13,46,GBS)
810.81 return
REM> VERTICAL DIM & FACE

```

```

810.82 if XCB(SHM)=0 then SLV(SHM)=0 else SLV(SHM)=(SMVCC(SHM)-SMVGB(SHM))/XCB(SHM)
if SHM=SHM then goto 810.83
SMVCC(SHM)=(SLV(SHM)*XCB(SHM))+SMVGB(SHM)
PVE(SHM-1)=SMVCC(SHM-1)-SMVCC(SHM)+CVE(SHM-1)
MVE(SHM-1)=PVE(SHM-1)-DVE(SHM-1)
SMVCF(SHM)=SLV(SHM)*CD(SHM-1)
PVG(SHM-1)=SMVCF(SHM)-SMVCF(SHM-1)+CVG(SHM-1)
MVG(SHM-1)=PVG(SHM-1)-DVG(SHM-1)
if NGM=1 then goto 810.83
call fn.DISPCN(MTS,CLS,6,12," ")
print using "MM.MMM! MM.MMM! MM.MMM!": \
DVE(SHM-1),an(DVE(SHM-1),"L","H"), \
PVE(SHM-1),an(PVE(SHM-1),"L","H"),MVE(SHM-1);
call fn.DISPCN(MTS,CLS,7,12," ")
print using "MM.MMM! MM.MMM! MM.MMM!": \
DVG(SHM-1),an(DVG(SHM-1),"C","O"), \
PVG(SHM-1),an(PVG(SHM-1),"C","O"),MVG(SHM-1);
810.83 if SHM=FSH then goto 810.84
SMVCC(SHM)=(SLV(SHM)*XCB(SHM))+SMVGB(SHM)
PVE(SHM)=SMVCC(SHM)-SMVCC(SHM-1)+CVE(SHM)
MVE(SHM)=PVE(SHM)-DVE(SHM)
SMVCF(SHM)=SLV(SHM)*CD(SHM)
PVG(SHM)=SMVCF(SHM)-SMVCF(SHM-1)+CVG(SHM)
MVG(SHM)=PVG(SHM)-DVG(SHM)
if NGM=1 then goto 810.84
call fn.DISPCN(MTS,CLS,8,45," ")
print using "MM.MMM MM.MMM! MM.MMM!": \
MVE(SHM),PVE(SHM),an(PVE(SHM),"L","H"), \
DVE(SHM),an(DVE(SHM),"L","H");
call fn.DISPCN(MTS,CLS,7,45," ")
print using "MM.MMM MM.MMM! MM.MMM!": \
MVG(SHM),PVG(SHM),an(PVG(SHM),"C","O"), \
DVG(SHM),an(DVG(SHM),"C","O");
810.84 return
REND VERTICAL BORES
810.85 if BORE=" " then goto 810.86
SMVCC(SHM)=(SLV(SHM)*XCB(SHM))+SMVGB(SHM)
SMVCC(SHM)=(SLV(SHM)*XCB(SHM))+SMVGB(SHM)
SMVCC(SHM)=(SLV(SHM)*XCB(SHM))+SMVGB(SHM)
SMVCC(SHM)=(SLV(SHM)*XCB(SHM))+SMVGB(SHM)
CPVCC(SHM)=CMVCC(SHM)-SMVCC(SHM): CMVCC(SHM)=CPVCC(SHM)-CMVCC(SHM)
CPVCC(SHM)=CMVCC(SHM)-SMVCC(SHM): CMVCC(SHM)=CPVCC(SHM)-CMVCC(SHM)
CPVCC(SHM)=CMVCC(SHM)-SMVCC(SHM): CMVCC(SHM)=CPVCC(SHM)-CMVCC(SHM)
CPVCC(SHM)=CMVCC(SHM)-SMVCC(SHM): CMVCC(SHM)=CPVCC(SHM)-CMVCC(SHM)
CPVCC(SHM)=CMVCC(SHM)-SMVCC(SHM): CMVCC(SHM)=CPVCC(SHM)-CMVCC(SHM)
if NGM=1 then goto 810.86
call fn.DISPCN(MTS,CLS,8,12," ")
if BORE="O" \
then print using "MM.MMM! MM.MMM! MM.MMM!": \
CMVCC(SHM),an(CMVCC(SHM),"L","H"), \
CPVCC(SHM),an(CPVCC(SHM),"L","H"),CMVCC(SHM); \
else print using "MM.MMM! MM.MMM! MM.MMM!": \
CMVCC(SHM),an(CMVCC(SHM),"L","H"), \
CPVCC(SHM),an(CPVCC(SHM),"L","H"),CMVCC(SHM);
call fn.DISPCN(MTS,CLS,8,45," ")
if BORE="O" \
then print using "MM.MMM MM.MMM! MM.MMM!": \
CMVCC(SHM),CPVCC(SHM),an(CPVCC(SHM),"L","H"), \
CMVCC(SHM),an(CMVCC(SHM),"L","H"); \
else print using "MM.MMM MM.MMM! MM.MMM!": \
CMVCC(SHM),CPVCC(SHM),an(CPVCC(SHM),"L","H"), \
CMVCC(SHM),an(CMVCC(SHM),"L","H");
810.86 return
REND HORIZONTAL RIM & FACE
810.87 if XCB(SHM)=0 then SLH(SHM)=0 else SLH(SHM)=(SMHCB(SHM)-SMHGB(SHM))/XCB(SHM)
if SHM=FSH then goto 810.82
SMHCC(SHM)=(SLH(SHM)*XCB(SHM))+SMHGB(SHM)
PHE(SHM-1)=SMHCC(SHM-1)-SMHCC(SHM)+CHE(SHM-1)
MHE(SHM-1)=PHE(SHM-1)-DHE(SHM-1)
SMHCF(SHM)=SLH(SHM)*CD(SHM-1)
PHG(SHM-1)=SMHCF(SHM)-SMHCF(SHM-1)+CHG(SHM-1)
MHG(SHM-1)=PHG(SHM-1)-DHG(SHM-1)
if NGM=1 then goto 810.92
call fn.DISPCN(MTS,CLS,10,12," ")
print using "MM.MMM! MM.MMM! MM.MMM!": \
DHE(SHM-1),an(DHE(SHM-1),"R","L"), \
PHE(SHM-1),an(PHE(SHM-1),"R","L"),MHE(SHM-1);
call fn.DISPCN(MTS,CLS,10,12," ")
print using "MM.MMM! MM.MMM! MM.MMM!": \
DHG(SHM-1),an(DHG(SHM-1),"C","O"), \
PHG(SHM-1),an(PHG(SHM-1),"C","O"),MHG(SHM-1);
810.92 if SHM=FSH then goto 810.94
SMHCC(SHM)=(SLH(SHM)*XCB(SHM))+SMHGB(SHM)
PHE(SHM)=SMHCC(SHM)-SMHCC(SHM-1)+CHE(SHM)
MHE(SHM)=PHE(SHM)-DHE(SHM)
SMHCF(SHM)=SLH(SHM)*CD(SHM)
PHG(SHM)=SMHCF(SHM)-SMHCF(SHM-1)+CHG(SHM)
MHG(SHM)=PHG(SHM)-DHG(SHM)
if NGM=1 then goto 810.93
call fn.DISPCN(MTS,CLS,11,45," ")
print using "MM.MMM MM.MMM! MM.MMM!": \
MHE(SHM),PHE(SHM),an(PHE(SHM),"R","L"), \
DHE(SHM),an(DHE(SHM),"R","L");
call fn.DISPCN(MTS,CLS,10,45," ")
print using "MM.MMM MM.MMM! MM.MMM!": \
MHG(SHM),PHG(SHM),an(PHG(SHM),"C","O"), \
DHG(SHM),an(DHG(SHM),"C","O");
810.93 return
REND HORIZONTAL BORES
810.94 if BORE=" " then goto 810.95
SMHCC(SHM)=(SLH(SHM)*XCB(SHM))+SMHGB(SHM)
SMHCC(SHM)=(SLH(SHM)*XCB(SHM))+SMHGB(SHM)
SMHCC(SHM)=(SLH(SHM)*XCB(SHM))+SMHGB(SHM)
SMHCC(SHM)=(SLH(SHM)*XCB(SHM))+SMHGB(SHM)
CPHCC(SHM)=CMHCC(SHM)-SMHCC(SHM): CMHCC(SHM)=CPHCC(SHM)-CMHCC(SHM)
CPHCC(SHM)=CMHCC(SHM)-SMHCC(SHM): CMHCC(SHM)=CPHCC(SHM)-CMHCC(SHM)
CPHCC(SHM)=CMHCC(SHM)-SMHCC(SHM): CMHCC(SHM)=CPHCC(SHM)-CMHCC(SHM)
CPHCC(SHM)=CMHCC(SHM)-SMHCC(SHM): CMHCC(SHM)=CPHCC(SHM)-CMHCC(SHM)
CPHCC(SHM)=CMHCC(SHM)-SMHCC(SHM): CMHCC(SHM)=CPHCC(SHM)-CMHCC(SHM)
if NGM=1 then goto 810.95
call fn.DISPCN(MTS,CLS,20,12," ")
if BORE="O" \
then print using "MM.MMM! MM.MMM! MM.MMM!": \
CMHCC(SHM),an(CMHCC(SHM),"R","L"), \
CPHCC(SHM),an(CPHCC(SHM),"R","L"),CMHCC(SHM); \
else print using "MM.MMM! MM.MMM! MM.MMM!": \
CMHCC(SHM),an(CMHCC(SHM),"R","L"), \
CPHCC(SHM),an(CPHCC(SHM),"R","L"),CMHCC(SHM);
call fn.DISPCN(MTS,CLS,20,45," ")
if BORE="O" \
then print using "MM.MMM MM.MMM! MM.MMM!": \
CMHCC(SHM),CPHCC(SHM),an(CPHCC(SHM),"R","L"), \
CMHCC(SHM),an(CMHCC(SHM),"R","L"); \
else print using "MM.MMM MM.MMM! MM.MMM!": \
CMHCC(SHM),CPHCC(SHM),an(CPHCC(SHM),"R","L"), \
CMHCC(SHM),an(CMHCC(SHM),"R","L");
810.95 return

```



```

        DATA(5),DATA(6),DATA(7),DATA(8)
        RD(1,1)=VAL(DATA(1))
        RD(1,2)=VAL(DATA(2))
        RD(1,3)=VAL(DATA(3))
        RD(1,4)=VAL(DATA(4))
        RD(1,5)=VAL(DATA(5))
        RD(1,6)=VAL(DATA(6))
        RD(1,7)=VAL(DATA(7))
        RD(1,8)=VAL(DATA(8))
        next I%
REM> READ BEARING GEOMETRY (PAGE=PAD ANGLE COV. SIDE BEARING, PAGE=COL. SIDE).
for I%=55% to 95%
    read# 10,17,18,DATA(1),DATA(2),DATA(3),DATA(4),DATA(5),\
        DATA(6),DATA(7),DATA(8),DATA(9),DATA(10)

    if DATA(1)="" then PAGE(I%,1)=99.1234 else PAGE(I%,1)=VAL(DATA(1))
    if DATA(2)="" then PAGE(I%,2)=99.1234 else PAGE(I%,2)=VAL(DATA(2))
    if DATA(3)="" then PAGE(I%,3)=99.1234 else PAGE(I%,3)=VAL(DATA(3))
    if DATA(4)="" then PAGE(I%,4)=99.1234 else PAGE(I%,4)=VAL(DATA(4))
    if DATA(5)="" then PAGE(I%,5)=99.1234 else PAGE(I%,5)=VAL(DATA(5))

    if DATA(6)="" then PAGE(I%,1)=99.1234 else PAGE(I%,1)=VAL(DATA(6))
    if DATA(7)="" then PAGE(I%,2)=99.1234 else PAGE(I%,2)=VAL(DATA(7))
    if DATA(8)="" then PAGE(I%,3)=99.1234 else PAGE(I%,3)=VAL(DATA(8))
    if DATA(9)="" then PAGE(I%,4)=99.1234 else PAGE(I%,4)=VAL(DATA(9))
    if DATA(10)="" then PAGE(I%,5)=99.1234 else PAGE(I%,5)=VAL(DATA(10))
next I%
REM> READ CASINO GEOMETRY.
read# 10,25,DATA(1),DATA(2),DATA(3),DATA(4),\
    DATA(5),DATA(6),DATA(7),DATA(8)
read# 10,26,DATA(9),DATA(10),DATA(11),DATA(12),\
    DATA(13),DATA(14)
for I%=55% to 95%
    if DATA(I%*2-1)="" then RDC(I%,1)=99.1234 else RDC(I%,1)=VAL(DATA(I%*2-1))
    if DATA(I%*2)="" then RDC(I%,2)=99.1234 else RDC(I%,2)=VAL(DATA(I%*2))
next I%
read# 10,27,DATA(1),DATA(2),DATA(3),DATA(4),\
    DATA(5),DATA(6),DATA(7),DATA(8)
read# 10,28,DATA(9),DATA(10),DATA(11),DATA(12),\
    DATA(13),DATA(14),DATA(15),DATA(16)
for I%=1 to 16
    if DATA(I%)="" then RDC(I%)=99.1234 else RDC(I%)=VAL(DATA(I%))
next I%
read# 10,29,DATA(1),DATA(2),DATA(3),DATA(4),\
    DATA(5),DATA(6),DATA(7),DATA(8)
read# 10,30,DATA(9),DATA(10),DATA(11),DATA(12),\
    DATA(13),DATA(14),DATA(15),DATA(16)
for I%=1 to 16
    if DATA(I%)="" then RDE(I%)=99.1234 else RDE(I%)=VAL(DATA(I%))
next I%

for I%=1 to 16
    DATA(I%)=NULL
next I%
dim DATA(0), OA(0)
close 10
REM>-----2-----3-----4-----5-----6-----7-----
REM> READ WORK FILE.
open "WORK.DDD" rac# 80 as 5
read# 5,1,CNTX,CNTX
J%=1
for I%=55% to 95%
    read# 5,1+J%:SNVGB(I%),SNVGB(I%),SNVGB(I%),SNVGB(I%)
    read# 5,2+J%:CMVGO(I%),CMVGO(I%),CMVGO(I%),CMVGO(I%),CMVGO(I%),CMVGO(I%),CMVGO(I%),CMVGO(I%)
    J%=J%+4
next I%

close 5
REM>-----2-----3-----4-----5-----6-----7-----
REM> CALCULATE VERTICAL & HORIZONTAL MOVES.
20.35 call fn.DISPMV(1,10,18,"COMPLETING CALCULATIONS.....")
for I%=55% to 95%
    if left$(SWS(I%),3)="GEN" then goto 20.36
    if left$(SWS(I%),3)="ENC" then goto 20.38
REM> CALCULATE BEARING SHIR MOVES.
for I%=1 to 5
    if PAGE(I%,1)=99.1234 then SHCB(I%,1)=0 :SNVGB(I%,1)=0 :goto 20.351
    SHCB(I%,1)=SNVGB(I%)*SIN(PAGE(I%,1)*3.1415926/180)
    SNVGB(I%,1)=SNVGB(I%)*COS(PAGE(I%,1)*3.1415926/180)
20.351 STCB(I%,1)=SHCB(I%,1)+SNVGB(I%,1)
    if PAGE(I%,2)=99.1234 then SHCB(I%,2)=0 :SNVGB(I%,2)=0 :goto 20.352
    SHCB(I%,2)=SNVGB(I%)*SIN(PAGE(I%,2)*3.1415926/180)
    SNVGB(I%,2)=SNVGB(I%)*COS(PAGE(I%,2)*3.1415926/180)
20.352 STCB(I%,2)=SHCB(I%,2)+SNVGB(I%,2)
next I%
CMX=CMX+30 :call fn.DISPMV(1,18,40,STR$(CMX))

if left$(SWS(I%),3)="ENC" then goto 20.37
if left$(SWS(I%),2)="TH" or left$(SWS(I%),2)="IP" then goto 20.353
goto 20.38
REM> HP OR IP, CALCULATE CASINO SHIR MOVES.
20.353 if -RD(1,7)+RD(1,3)-RD(1,6)=0 then SLVC(I%)=0 :SLHC(I%)=0 \
    else SLVC(I%)=(CMVGB(I%)-CMVGO(I%))/(-RD(1,7)+RD(1,3)-RD(1,6)) \
    :SLHC(I%)=(CMHGO(I%)-CMHGB(I%))/(-RD(1,7)+RD(1,3)-RD(1,6))

    if RDC(1,1)=99.1234 then CMVGB(I%)=99.1234 \
    else XGB(I%)=-RD(1,7)+RDC(1,1) \
    :CMVGB(I%)=SLVC(I%)*COS(I%)+CMVGO(I%)

    if RDC(1,2)=99.1234 then CMHGB(I%)=99.1234 \
    else XGB(I%)=-RD(1,7)+RDC(1,2)-RDC(1,1) \
    :CMHGB(I%)=SLVC(I%)*SIN(I%)-CMVGO(I%)
    CMX=CMX+8 :call fn.DISPMV(1,16,40,STR$(CMX))
    goto 20.38
REM> GENERATORS & EXCITORS.
REM> CALCULATE FRAME SHIR MOVES.
20.36 if RD(1,3)=0 then SLVC(I%)=0 :SLHC(I%)=0 \
    else SLVC(I%)=(SNVGB(I%)-SNVGB(I%))/RD(1,3) \
    :SLHC(I%)=(SNHCB(I%)-SNHCB(I%))/RD(1,3)
    CMX=CMX+2 :call fn.DISPMV(1,18,40,STR$(CMX))

    if left$(SWS(I%),3)="ENC" then goto 20.37
    if match("WEST",MFRS,1) < 0 then goto 20.365
REM> DE - GENERATOR.
    if RDC(1)=99.1234 then CMHGB(I%)=99.1234 \
    else XGB(I%)=-RDC(1) \
    :CMHGB(I%)=SLHC(I%)*XGB(I%)+SNHGB(I%)

    if RDC(2)=99.1234 or RDC(1)=99.1234 then RDC(1)=99.1234 \

```

```

else XG(1)=XGS(1%)*RDG(2)/2 \
      (MG(1)+SLVC(1%)*XG(1)+SMVGB(1%))

for J% = 2 to 7
  if RDG(J%+1)=99.1234 or RDG(1)=99.1234 then MG(J%)=99.1234 :goto 20.361
  XG(J%)=XGS(1%)-RDG(J%+1)/2
  for K% = 2 to J%
    if RDG(K%) ne 99.1234 then XG(J%)=XG(J%)+XGS(K%)
  next K%
  MG(J%)=SLVC(1%)*XG(J%)+SMVGB(1%)
20.361 next J%

if RDG(8)=99.1234 then CMHCS(1%)=99.1234 \
  else XCS(1%)=RD(1%,3)+RDG(8) \
      CMHCS(1%)=SLMC(1%)*XCS(1%)+SMHGB(1%)

if RDG(14)=99.1234 or RDG(9)=99.1234 then MG(14)=99.1234 \
  else XG(14)=XCS(1%)-RDG(14)/2 \
      (MG(14)+SLVC(1%)*XG(14)+SMVGB(1%))

for J% = 13 to 8 step -1
  if RDG(J%+2)=99.1234 or RDG(9)=99.1234 then MG(J%)=99.1234 :goto 20.362
  XG(J%)=XCS(1%)-RDG(J%+2)/2
  for K% = 18 to J%+3 step -1
    if RDG(K%) ne 99.1234 then XG(J%)=XG(J%)-RDG(K%)
  next K%
  MG(J%)=SLVC(1%)*XG(J%)+SMVGB(1%)
20.362 next J%
CM% = CM%+20 :call fn.DISPM%(1,18,49,str$(CM%))
goto 20.28

REM> NEXT - GENERATOR.
20.285
if RDG(1)=99.1234 \
  then CMHCS(1%)=99.1234 \
  else XCS(1%)=RD(1%,3)+RDG(1) \
      CMHCS(1%)=SLMC(1%)*XCS(1%)+SMHGB(1%)

if RDG(15)=99.1234 or RDG(1)=99.1234 \
  then MG(14)=99.1234 \
  else XG(14)=XCS(1%)-RDG(15)/2 \
      (MG(14)+SLVC(1%)*XG(14)+SMVGB(1%))

for J% = 13 to 1 step -1
  if RDG(J%+1)=99.1234 or RDG(1)=99.1234 \
    then MG(J%)=99.1234 \
    :goto 20.365
  XG(J%)=XCS(1%)-RDG(J%+1)/2
  for K% = 18 to J%+2 step -1
    if RDG(K%) ne 99.1234 \
      then XG(J%)=XG(J%)-RDG(K%)
  next K%
  MG(J%)=SLVC(1%)*XG(J%)+SMVGB(1%)
20.365 next J%

if RDG(1)=99.1234 \
  then CMHCS(1%)=99.1234 \
  :goto 20.38
XGS(1%)=XCS(1%)
for K% = 15 to 2 step -1
  if RDG(K%) ne 99.1234 \
    then XGS(1%)=XGS(1%)-RDG(K%)
next K%
CMHCS(1%)=SLMC(1%)*XGS(1%)+SMHGB(1%)
CM% = CM%+22 :call fn.DISPM%(1,18,49,str$(CM%))

goto 20.38

REM> ALL EXCITORS.
20.37
if RDE(1)=99.1234 \
  then CMHCS(1%)=99.1234 \
  else XCS(1%)=RD(1%,3)+RDE(1) \
      CMHCS(1%)=SLMC(1%)*XCS(1%)+SMHGB(1%)

if RDE(15)=99.1234 or RDE(1)=99.1234 \
  then ME(14)=99.1234 \
  else XE(14)=XCS(1%)-RDE(15)/2 \
      (ME(14)+SLVC(1%)*XE(14)+SMVGB(1%))

for J% = 13 to 1 step -1
  if RDE(J%+1)=99.1234 or RDE(1)=99.1234 \
    then ME(J%)=99.1234 \
    :goto 20.371
  XE(J%)=XCS(1%)-RDE(J%+1)/2
  for K% = 18 to J%+2 step -1
    if RDE(K%) ne 99.1234 \
      then XE(J%)=XE(J%)-RDE(K%)
  next K%
  ME(J%)=SLVC(1%)*XE(J%)+SMVGB(1%)
20.371 next J%

if RDE(1)=99.1234 \
  then CMHCS(1%)=99.1234 \
  :goto 20.38
XGS(1%)=XCS(1%)
for K% = 15 to 2 step -1
  if RDE(K%) ne 99.1234 \
    then XGS(1%)=XGS(1%)-RDE(K%)
next K%
CMHCS(1%)=SLMC(1%)*XGS(1%)+SMHGB(1%)
CM% = CM%+22 :call fn.DISPM%(1,18,49,str$(CM%))

20.38
next 1%
if CM% < 100 then call fn.DISPM%(1,18,49,"-str$(CM%)- COMPLETED" \
  :else call fn.DISPM%(1,18,49,str$(CM%)- COMPLETED")
CM% = CM%+CM%

REM-----1-----2-----3-----4-----5-----6-----7-----
REM> WRITE TO WORK FILE.
create "WORK1.DDD" read 80 as 8
J% = 0
F7% = ""
for I% = 50% to 99%
  print using F7%# 8, 1-J%:STGB(1%,1),STGB(1%,2),STGB(1%,3),STGB(1%,4),STGB(1%,5),0,0
  print using F7%# 8, 2-J%:STGB(1%,1),STGB(1%,2),STGB(1%,3),STGB(1%,4),STGB(1%,5),0,0
  print using F7%# 8, 3-J%:CMVCS(1%),CMVCS(1%),CMHCS(1%),CMHCS(1%),0,0,0
  J% = J%+3
next I%
print using F7%# 8, 1-J%:MG(1),MG(2),MG(3),MG(4),MG(5),MG(6),MG(7)
print using F7%# 8, 2-J%:MG(8),MG(9),MG(10),MG(11),MG(12),MG(13),MG(14)
print using F7%# 8, 3-J%:ME(1),ME(2),ME(3),ME(4),ME(5),ME(6),ME(7)
print using F7%# 8, 4-J%:ME(8),ME(9),ME(10),ME(11),ME(12),ME(13),ME(14)

```

41 *****


```

NLEN=LEN(SNS(REPN))+LEN(DOS)
IF DOS="Manually" \
  then DTOP15=" "
  Reference Shaft: "SNS(REPN)+LEFTS(BLANKS,34-NLEN)+Positioned "DOS" by User" \
  else DTOP15=" "
  Reference Shaft: "SNS(REPN)+LEFTS(BLANKS,29-NLEN)+Positioned To "DOS" Alignment" \
  if AUTOS="Off" \
  then DTOP25=" "
  else DTOP25=" "
  Auto-Position: "AUTOS" \
  Auto-Position: Using "AUTOS" Bore"
-----3-----4-----5-----6-----7-----
REM VARIABLE NAMING CONVENTION.
REM
REM 1 2 3 4 5
REM + + + + +
REM | | | | | C=Coupling, B=Bearing, D=Disk, S=Shaft Shim, G=Gland
REM | | | | | G=Governor and, C=Collector and
REM | | | | | V=Vertical, H=Horizontal
REM | | | | | D=Design, C=Current, M=Move, Projected
REM | | | | | S=Relative Shaft position, C=shaft position in Casing
REM -----2-----3-----4-----5-----6-----7-----
REM INITIALIZE & DIMENSION VARIABLES.
  %N=NSM
  DIM DVG(1%+1),DHG(1%+1),DVE(1%+1),DHE(1%+1)
  DIM CVG(1%+1),CHG(1%+1),CVE(1%+1),CHE(1%+1)
  DIM PVG(1%+1),PHG(1%+1),PVE(1%+1),PHE(1%+1)
  DIM MVG(1%+1),MHG(1%+1),MVE(1%+1),MHE(1%+1)

  DIM SMVCC(1%),SMVCB(1%)
  DIM SMVCC(1%),SMVCB(1%)
  DIM SMVGC(1%),SMVGB(1%)
  DIM SMVGC(1%),SMVGB(1%)
  DIM SMVDF(1%),SMVCF(1%),SMHDF(1%),SMHCF(1%)

  DIM PACB(1%,2),PACB(1%,5),RD(1%,4)
  DIM STCB(1%,2),STCB(1%,5)

  DIM CDVGO(1%),CDVGB(1%)
  DIM CDHGO(1%),CDHGB(1%)
  DIM CDVGO(1%),CDVGB(1%)
  DIM CDHGO(1%),CDHGB(1%)

  DIM CCVGO(1%),CCVGB(1%)
  DIM CCHGO(1%),CCHGB(1%)
  DIM CCVGO(1%),CCVGB(1%)
  DIM CCHGO(1%),CCHGB(1%)

  DIM CPVGO(1%),CPVGB(1%)
  DIM CPHGO(1%),CPHGB(1%)
  DIM CPVGO(1%),CPVGB(1%)
  DIM CPHGO(1%),CPHGB(1%)

  DIM CMVGO(1%),CMVGB(1%)
  DIM CMHGO(1%),CMHGB(1%)
  DIM CMVGO(1%),CMVGB(1%)
  DIM CMHGO(1%),CMHGB(1%)

  DIM CMVGE(1%),CMVGB(1%)
  DIM CMHGE(1%),CMHGB(1%)
  DIM MG(1%),MR(1%)

REM-----1-----2-----3-----4-----5-----6-----7-----
DIM DATAS(16)

REM READ ROTOR GEOMETRY (RD=ROTOR DIMENSION).
  for I%=SSM% to FSH%
    read I,10-1%:DATAS(1),DATAS(2),DATAS(3),DATAS(4),\
      DATAS(5),DATAS(6),DATAS(7),DATAS(8)
    RD(1%,2)=VAL(DATAS(2))
    RD(1%,4)=VAL(DATAS(4))
  next I%

REM READ BEARING GEOMETRY (PACB=PAD ANGLE GOV. SIDE BEARING, PACB=COL. SIDE).
  for I%=SSM% to FSH%
    read I,17-1%:DATAS(1),DATAS(2),DATAS(3),DATAS(4),DATAS(5),\
      DATAS(6),DATAS(7),DATAS(8),DATAS(9),DATAS(10)

    if DATAS(1)="" then PACB(I%,1)=99.1234 else PACB(I%,1)=VAL(DATAS(1))
    if DATAS(2)="" then PACB(I%,2)=99.1234 else PACB(I%,2)=VAL(DATAS(2))
    if DATAS(3)="" then PACB(I%,3)=99.1234 else PACB(I%,3)=VAL(DATAS(3))
    if DATAS(4)="" then PACB(I%,4)=99.1234 else PACB(I%,4)=VAL(DATAS(4))
    if DATAS(5)="" then PACB(I%,5)=99.1234 else PACB(I%,5)=VAL(DATAS(5))

    if DATAS(6)="" then PACB(I%,1)=99.1234 else PACB(I%,1)=VAL(DATAS(6))
    if DATAS(7)="" then PACB(I%,2)=99.1234 else PACB(I%,2)=VAL(DATAS(7))
    if DATAS(8)="" then PACB(I%,3)=99.1234 else PACB(I%,3)=VAL(DATAS(8))
    if DATAS(9)="" then PACB(I%,4)=99.1234 else PACB(I%,4)=VAL(DATAS(9))
    if DATAS(10)="" then PACB(I%,5)=99.1234 else PACB(I%,5)=VAL(DATAS(10))
  next I%

REM READ DESIGN/DESIRED COUPLINGS.
  for I%=SSM% to FSH%
    read I,30-1%:DATAS(1),DATAS(2),DATAS(3),DATAS(4),\
      DATAS(5),DATAS(6),DATAS(7),DATAS(8)
    if DOS="Desired" then goto 20.01
    DVE(1%)=VAL(DATAS(1))
    DHG(1%)=VAL(DATAS(2))
    DHE(1%)=VAL(DATAS(3))
    DVG(1%)=VAL(DATAS(4))
    goto 20.02
20.01 if DATAS(2)="" then DVE(1%)=VAL(DATAS(1)) else DVE(1%)=VAL(DATAS(2))
    if DATAS(4)="" then DHG(1%)=VAL(DATAS(3)) else DHG(1%)=VAL(DATAS(4))
    if DATAS(6)="" then DHE(1%)=VAL(DATAS(5)) else DHE(1%)=VAL(DATAS(6))
    if DATAS(8)="" then DVG(1%)=VAL(DATAS(7)) else DVG(1%)=VAL(DATAS(8))
20.02 next I%
REM READ DESIGN/DESIRED GIL & GLAND BORES.
  for I%=SSM% to FSH%
    read I,37-(2*(I%-1)):DATAS(1),DATAS(2),DATAS(3),DATAS(4),\
      DATAS(5),DATAS(6),DATAS(7),DATAS(8)
    read I,38-(2*(I%-1)):DATAS(9),DATAS(10),DATAS(11),DATAS(12),\
      DATAS(13),DATAS(14),DATAS(15),DATAS(16)
    if DOS="Desired" then goto 20.04
    CDHGO(1%)=VAL(DATAS(1))
    CDHGO(1%)=VAL(DATAS(3))
    CDVGO(1%)=VAL(DATAS(5))
    CDHGO(1%)=VAL(DATAS(7))
    CDVGO(1%)=VAL(DATAS(9))
    CDHGO(1%)=VAL(DATAS(11))
    CDVGO(1%)=VAL(DATAS(13))
    CDHGO(1%)=VAL(DATAS(15))
    goto 20.04
20.04 if DATAS(2)="" then CDHGO(1%)=VAL(DATAS(1)) else CDHGO(1%)=VAL(DATAS(2))
    if DATAS(4)="" then CDHGO(1%)=VAL(DATAS(3)) else CDHGO(1%)=VAL(DATAS(4))
    if DATAS(6)="" then CDVGO(1%)=VAL(DATAS(5)) else CDVGO(1%)=VAL(DATAS(6))

```

```

if DATA8(8)="" then CMHGG(1%)=VAL(DATA8(7)) else CMHGG(1%)=VAL(DATA8(8))
if DATA8(9)="" then CMVGG(1%)=VAL(DATA8(8)) else CMVGG(1%)=VAL(DATA8(9))
if DATA8(12)="" then CMHGG(2%)=VAL(DATA8(11)) else CMHGG(2%)=VAL(DATA8(12))
if DATA8(14)="" then CMVGG(1%)=VAL(DATA8(13)) else CMVGG(1%)=VAL(DATA8(14))
if DATA8(16)="" then CMHGG(3%)=VAL(DATA8(15)) else CMHGG(3%)=VAL(DATA8(16))
20.05 next %
REM> READ CURRENT COUPLINGS.
dim DA(8)
for I%=55% to F5%+1
readw 10,54+(4*(I%-1));DA(1),DA(2),DA(3),DA(4),\
CVC(1%),CNE(1%),CVG(1%),CHG(1%)
next I%
REM> READ CURRENT OIL & GLAND BORES.
for I%=55% to F5%
20.08 readw 10,75+(4*(I%-1));D1S,D2S,D3S,D4S,CCVGG(1%),CMHGG(1%)
readw 10,76+(4*(I%-1));D1S,D2S,D3S,D4S,CCVGG(1%),CMHGG(1%)
readw 10,77+(4*(I%-1));D1S,D2S,D3S,D4S,CCVGG(1%),CMHGG(1%)
readw 10,78+(4*(I%-1));D1S,D2S,D3S,D4S,CCVGG(1%),CMHGG(1%)
next I%

for I%=1 to 16
DATA8(I%)=NULLS
next I%
dim DATA9(0), DA(0)
close 10
REM>-----2-----3-----4-----5-----6-----7-----
REM> READ WORK FILE.
open "WORK.000" rec1 60 as 5
J%=1
for I%=55% to F5%
readw 5, 1-J%:SMVGG(1%),SMVGG(1%),SMHGG(1%),SMHGG(1%),SMVGG(1%),SMVGG(1%),SMHGG(1%),SMHGG(1%)
readw 5, 2-J%:CMVGG(1%),CMVGG(1%),CMVGG(1%),CMVGG(1%),CMHGG(1%),CMHGG(1%),CMHGG(1%),CMHGG(1%)
readw 5, 3-J%:CPVGG(1%),CPVGG(1%),CPVGG(1%),CPVGG(1%),CPHGG(1%),CPHGG(1%),CPHGG(1%),CPHGG(1%)
readw 5, 4-J%:SMVGG(1%),SMVGG(1%),SMHGG(1%),SMHGG(1%)
J%=J%+4
next I%
for I%=55% to F5%+1
readw 5, 1-I%:PVE(1%),PVE(1%),PVG(1%),PVG(1%),PHE(1%),PHE(1%),PHG(1%),PHG(1%)
J%=J%+1
next I%
delete 5
open "WORK1.000" rec1 80 as 6
J%=0
for I%=55% to F5%
readw 6, 1-J%:STGB(1%,1),STGB(1%,2),STGB(1%,3),STGB(1%,4),STGB(1%,5)
readw 6, 2-J%:STGB(1%,1),STGB(1%,2),STGB(1%,3),STGB(1%,4),STGB(1%,5)
readw 6, 3-J%:CMVGG(1%),CMVGG(1%),CMHGG(1%),CMHGG(1%)
J%=J%+3
next I%
readw 6, 1-J%:ME(1),ME(2),ME(3),ME(4),ME(5),ME(6),ME(7)
readw 6, 2-J%:ME(8),ME(9),ME(10),ME(11),ME(12),ME(13),ME(14)
readw 6, 3-J%:ME(1),ME(2),ME(3),ME(4),ME(5),ME(6),ME(7)
readw 6, 4-J%:ME(8),ME(9),ME(10),ME(11),ME(12),ME(13),ME(14)
J%=J%+4
next I%
REM>-----2-----3-----4-----5-----6-----7-----
REM> PRINT BEARING PAD SHIM CHANGE REPORT.
REM> DISPLAY MESSAGE.
20.401 call fn.DISPM(1,24,5,\
"DO YOU WANT TO DISPLAY OR PRINT THE RESULTS? (D/P/ESC) ")
20.402 call fn.DISPM(1,24,6," ")
I%=I%+1
if I%=55 OR I%=100 then print chr$(I%); :D1S%+1 :goto 20.404
if I%=80 OR I%=132 then print chr$(I%); \
:call fn.DISPM(1,24,64,"===== PRINTING") :D1S%+0 :goto 20.403
if I%=27 then ROUTE%+1 :obtain "ESC"
goto 20.402

REM> ACTIVATE PRINTER.
20.403 printer width 0
print P103;
print P103;
REM> SETUP FOR REPORT.
20.404 dim PAB(32)
open 0003 rec1 81 as 10

REM> BEGIN REPORT.
20.405 I%=55%
20.406 if I% lt 55% then I%=55%
if I%>1 and RQ(I%,2)=0 then PAC%+1 else PAC%+0
if I%lt55% and RQ(I%,4)=0 then PAC%+1 else PAC%+0

if D1S%=0 then gosub 20.91 else gosub 20.92
print " BEARING PAD SHIM CHANGE REPORT"
print " ===== "
if D1S%=0 then \
print " Required bearing shim changes to align unit"
if I%=REP% AND AUTOS="off" or I% ne REP% then print else \
print " Note: Auto Position is active using "+AUTOS+" Bore"
if D1S%=0 then print
if D1S%=0 then print
J%=len(STR$(I%))+14
print lat$(BLANKS,40-J%/2)+STR$(I%)+ " (or #"+STR$(I%)+") SHAFT"
print lat$(BLANKS,40-J%/2)+I%+ "EQUALS, J%"
print

20.407 if D1S%=0 then print
print " Bearing #"+str$(I%+2-1)
print " Governor Rad PadW Angle Shim Change"
print " -----"
F1S% " # 25# 5 #.###"
F2S% " # 25# 25# NONE"
F3S% " NO PADS DEFINED"
if PAC%+1 then print :print :goto 20.408
for J%=1 to 5
if PAB(I%,J%) ne 99.1234 then goto 20.41
next J%
print
print F33
20.408 for J%=1 to 5 :print :next J%
goto 20.41
20.41 for J%=1 to 5
print
if PAB(I%,J%)=99.1234 then print :goto 20.42
if abs(STGB(I%,J%))<0.00005 \
then print using F2S%;J%,PAB(I%,J%) :goto 20.42
if STGB(I%,J%)>0 \
then print using F1S%;J%,PAB(I%,J%), " Add",STGB(I%,J%) \
else print using F1S%;J%,PAB(I%,J%), "Remove",abs(STGB(I%,J%))
next J%
20.42 next J%
20.43 print
print
gosub 20.88
if D1S%=0 then goto 20.431
if PNT%>0 then I%=F5%
if PNT%>3 then I%=I%-2

```

```

on PNT% goto 20.466, 20.431, 20.49, 20.48, 20.49, 20.49
20.431 if DIS%=1 then call fn.CLR%(MT%,78,8,1,23,80)
print " Bearing ="+str$(IN%2)
print " Collector End Pass Angle Shift Change"
print " -----"
if PAC%=1 then print :print :goto 20.432
for J%=1 to 3
if PACB(IN,J%) ne 99.1234 then goto 20.44
next J%
print F35
print F35
20.432 for J%=1 to 3 :print :next J%
goto 20.46
20.44 for J%=1 to 3
print
if PACB(IN,J%)=99.1234 then print :goto 20.45
if abs(STCB(IN,J%))<0.00005 \
then print using F25;J%,PACB(IN,J%) :goto 20.46
if STCB(IN,J%)>0 \
then print using F15;J%,PACB(IN,J%)," And",STCB(IN,J%) \
else print using F15;J%,PACB(IN,J%),"Remove",abs(STCB(IN,J%))
20.45 next J%
print
20.46 print
gosub 20.85
if DIS%<0 then goto 20.481
if PNT%<0 then IS=PNT%
if PNT%<3 then IS=IS-2
on PNT% goto 20.466, 20.481, 20.49, 20.49, 20.49, 20.49
20.481 if DIS%<1 then call fn.CLR%(MT%,78,8,1,23,80)
if DIS%<0 then print :print AST5 :print :print
if PAC%<1 then goto 20.482
if SHVGB(IN)=0 then PRS(1)="None" :goto 20.482
if SHVGB(IN) gt 0 then PRS(1)=fn.RCS(SHVGB(IN),"N.###") :PRS(2)="Up" \
:PRS(3)=" " :goto 20.482
if SHVGB(IN) lt 0 then PRS(3)=" " \
:PRS(3)=fn.RCS(abs(SHVGB(IN)),"N.###") :PRS(10)="Down"
20.482 if PAC%<1 then goto 20.47
if SHVGB(IN)=0 then PRS(11)="None" :goto 20.47
if SHVGB(IN) gt 0 then PRS(11)=fn.RCS(SHVGB(IN),"N.###") :PRS(12)="Up" \
:PRS(13)=" " :goto 20.47
if SHVGB(IN) lt 0 then PRS(13)=" " \
:PRS(13)=fn.RCS(abs(SHVGB(IN)),"N.###") :PRS(20)="Down"
20.47 if PAC%<1 then goto 20.471
if SHVGB(IN)=0 then PRS(4)="None" :goto 20.471
if SHVGB(IN) gt 0 then PRS(4)=fn.RCS(SHVGB(IN),"N.###") :PRS(5)="Left" :goto 20.471
if SHVGB(IN) lt 0 then PRS(5)=fn.RCS(abs(SHVGB(IN)),"N.###") :PRS(7)="Right"
20.471 if PAC%<1 then goto 20.48
if SHVGB(IN)=0 then PRS(14)="None" :goto 20.48
if SHVGB(IN) gt 0 then PRS(14)=fn.RCS(SHVGB(IN),"N.###") :PRS(15)="Left" :goto 20.48
if SHVGB(IN) lt 0 then PRS(16)=fn.RCS(abs(SHVGB(IN)),"N.###") :PRS(17)="Right"
20.48 if DIS%<1 then goto 20.481
print " CALCULATED SHAFT MOVE AT BEARINGS"
print " ====="
20.481 print " C=Coupling, X=Bearing"
print
print using " /-###/ // /-###/ // " :PRS(1),PRS(2),PRS(11),PRS(12)
print using " " :PRS(3), PRS(13)
print
print using " " :PRS(4),PRS(5),PRS(14),PRS(15)
print using " " :PRS(6),PRS(7),PRS(16),PRS(17)
print using " " :PRS(8), PRS(18)
print using " " :PRS(9),PRS(10),PRS(19),PRS(20)
print
if DIS%<0 then print
gosub 20.85
if DIS%<0 then goto 20.48
if PNT%<0 then IS=PNT%
if PNT%<3 then IS=IS-2
on PNT% goto 20.431, 20.46, 20.48, 20.49, 20.49, 20.49
20.48 for J%=1 to 20 :PRS(J%)+NULL$ :next J%
if DIS%<1 then if PNT%<5 then goto 20.485
IS=IS+1
if IS is PNT% then goto 20.488
if DIS%<1 then goto 20.50
print AST5
print BOT5
for IM=1 to 10
print
next IM
REM-----2-----3-----4-----5-----6-----7-----
Include C2K1

```

```

print
print
print using = / .NNN/ // / .NNN/ // "PRS( 1),PRS( 2), PRS(17),PRS(18)
print using = / / .NNN/ /nf/ / / .NNN/ /nf/ "PRS( 3), PRS(19)
print using = / / .NNN/ /nf/ / / .NNN/ /nf/ "PRS( 4),PRS( 5), PRS(20),PRS(21)
print using = /2345 Shaft/ " " /2345 Shaft/ "left(BLANK,6-len(SNS(1)))-SNS(1)" Shaft",SNS(1)
-1)" Shaft"
print
print
print using = / .NNN/ /gh/ / " / .NNN/ /gh/ / " "PRS( 6),PRS( 7), PRS(22),PRS(23)
print using = / .NNN/ /os/ / / / .NNN/ /os/ / "PRS( 8),PRS( 9),PRS(10),PRS(24),PRS(25),PRS(26)
print using = / / .NNN/ /os/ / / / .NNN/ /os/ / "PRS(11), PRS(27)
print using = / / .NNN/ /os/ / / / .NNN/ /os/ / "PRS(12),PRS(13), PRS(28),PRS(29)
print using = / / .NNN/ /os/ / / / .NNN/ /os/ / "PRS(14),PRS(15),PRS(16),PRS(30),PRS(31),PRS(32)
if DIS%0 then print
gosub 20.85
if DIS%0 then goto 20.88
if PNT%6 then IS=PNT%-1
if PNT%3 then IS=IS-2
on PNT% goto 20.82, 20.89, 20.88, 20.86, 20.89, 20.89
20.88 for J%+1 to 32 :PRS(J%)+NULL$ :next J%
if DIS%1 then if PNT%5 then goto 20.486
IS=IS+1
if IS le PNT%-1 then goto 20.81
if DIS%1 then goto 20.80
print AST3
print BOTS
for I%+1 to 16
print
next I%
NNNN-----1-----2-----3-----4-----5-----6-----7-----
%include C2K2

```

```

NNNN-----1-----2-----3-----4-----5-----6-----7-----
NNNN Copyright (C) COMPAQ Incorporated 1984,5,6
NNNN ALL RIGHTS RESERVED
NNNN TITLE: C2-TURBINE GENERATOR ALIGNMENT SYSTEM
NNNN VERSION: 00.1
NNNN REVISION: 012-21-89
NNNN FILE: C2K2 - CALCULATIONS: OIL & GLAND BORE REPORT
NNNN-----1-----2-----3-----4-----5-----6-----7-----
NNNN PRINT OIL AND GLAND BORE REPORT.
20.80 IS=SSM%
20.81 if IS le SSM% then IS=SSM%
if DIS%0 then gosub 20.81 else gosub 20.82
print " OIL & GLAND BORE REPORT"
print " "
if DIS%0 then \
print " Shaft position High or Left is Positive"
if IS=REP% and AUTO%="Off" or IS no REP% then print else \
print " Note: Auto Position is active using "+AUTO%+" Screw"
if DIS%0 then print print
J=len(SNS(1))+14
print left(BLANK,40-J%)+SNS(1)+" (or 1="+str$(IS)+") SHAFT"
print left(BLANK,40-J%/2)+left(EQUALS,J%)
print
if CMVGG(1%) gt 0 then PRS(1)="High" :goto 20.811
if CMVGG(1%) lt 0 then PRS(1)="Low " :goto 20.811
PRS(1)="None"
20.811 if CMVGG(1%) gt 0 then PRS(2)="High" :goto 20.812
if CMVGG(1%) lt 0 then PRS(2)="Low " :goto 20.812
PRS(2)="None"
20.812 if CMVGG(1%) gt 0 then PRS(3)="High" :goto 20.813
if CMVGG(1%) lt 0 then PRS(3)="Low " :goto 20.813
PRS(3)="None"
20.813 if CMVGG(1%) gt 0 then PRS(4)="High" :goto 20.814
if CMVGG(1%) lt 0 then PRS(4)="Low " :goto 20.814
PRS(4)="None"
20.814 if left(SNS(1),3)="GER" then \
print " Oil HZ Gland HZ Gland Oil" \
:print " "
:goto 20.815
if left(SNS(1),3)="EXC" then \
print " Outboard Oil Inboard Oil Inboard Oil Outboard Oil" \
:print " "
:goto 20.815
print " Oil Stead Gland Stead Gland Oil "
print " "
20.815 if DIS%8 then print
print " "
print using " " " " Current " " " " CMVGG(1%),CMVGG(1%),CMVGG(1%),CMVGG(1%)
print using " " " " Projected " " " " CMVGG(1%),CMVGG(1%),CMVGG(1%),CMVGG(1%)
print using " " " " " " " " CMVGG(1%),CMVGG(1%),CMVGG(1%),CMVGG(1%)

```

```

REM= Copyright (C) COMPAD Incorporated 1984,5,6
REM= ALL RIGHTS RESERVED
REM=
REM= TITLE: C2-TURBINE GENERATOR ALIGNMENT SYSTEM
REM= VERSION: 00.1
REM= REVISION: 212-21-00
REM= FILE: C2K1 - CALCULATIONS: COUPLING REPORT
REM=
REM=-----1-----2-----3-----4-----5-----6-----7-----
REM= PRINT COUPLING REPORT.
20.50 IF IS% EQ 0 THEN GOTO 20.51
20.51 IF IS% EQ 0 THEN GOTO 20.51 ELSE GOTO 20.52
      print "COUPLING REPORT"
      print "*****"
      if DISE% = 0 then \
        print "Governor End High or Left is Positive" \
        print "Open Bottom or Right is Positive" \
      else print
      if DISE% = 0 then print :print
      J% = len(SNS(1))-SNS(1)+1
      print left$(BLANKS,40-J%/2)-SNS(1)+"--"+SNS(1+1)+" (or "+chr$(64+IS%)+" COUPLING"
      print left$(BLANKS,40-J%/2)-left$(EDUALS,J%)
      print
      if MVE(1) < 0 then PRS(1)=SNS(1)+" is High" goto 20.511
      if MVE(1) > 0 then PRS(1)=SNS(1)+" is Low" goto 20.511
      PRS(1)=" OK"
20.511 if MVO(1) < 0 then PRS(2)="Bottom is Open" goto 20.512
      if MVO(1) > 0 then PRS(2)="Bottom is Closed" goto 20.512
      PRS(2)=" OK"
20.512 print "Rim Face"
      print "-----VERTICAL-----"
      print using "###.#### Current ##.#### CVE(1), CVQ(1)"
      print using "###.#### Projected ##.#### PVE(1), PVB(1)"
      print using "###.#### /design/ ##.#### DVE(1), DESD, DVQ(1)"
      print using "##.#### Variance ##.#### MVE(1), MVO(1)"
      print using "/p is High /ottom is Close/" PRS(1), PRS(2)
      if DISE% = 0 then print
      gosub 20.55
      if DISE% = 0 then goto 20.52
      if PMT% = 6 then IS%=PMT%-1
      if PMT% = 9 then IS%=14-2
      on PMT% goto 20.61, 20.52, 20.59, 20.69, 20.68, 20.69
20.52 if DISE% then call fn.CLW(MTS,70,0,1,20,60)
      if MRE(1) < 0 then PRS(1)=GMS(1)+" is Left" goto 20.521
      if MRE(1) > 0 then PRS(1)=GMS(1)+" is Right" goto 20.521
      PRS(1)=" OK"
20.521 if MRQ(1) < 0 then PRS(2)="Right is Open" goto 20.522
      if MRQ(1) > 0 then PRS(2)="Right is Closed" goto 20.522
      PRS(2)=" OK"
20.522 if DISE% = 1 then \
      print "Rim Face"
      print "-----HORIZONTAL-----"
      print using "###.#### Current ##.#### CHE(1), CHQ(1)"
      print using "###.#### Projected ##.#### PRE(1), PHQ(1)"
      print using "###.#### /design/ ##.#### DHE(1), DESD, DHQ(1)"
      print using "##.#### Variance ##.#### MRE(1), MRQ(1)"
      print using "/p is Right /ights is Close/" PRS(1), PRS(2)
      if DISE% = 0 then print :print
      gosub 20.55
      if DISE% = 0 then goto 20.53
      if PMT% = 6 then IS%=PMT%-1
      if PMT% = 9 then IS%=14-2
      on PMT% goto 20.61, 20.63, 20.69, 20.69, 20.69, 20.69
20.60 if OISE% then call fn.CLW(MTS,70,0,1,20,60)
      if OISE% = 0 then print ASTE :print :print
      PRS(1)=NULL : PRS(2)=NULL
      if SMVCC(1) < 0 then PRS(4)="None" goto 20.531
      if SMVCC(1) > 0 then PRS(4)=fn.RCS(SMVCC(1)):"N.####": PRS(5)="Left" goto 20.531
      PRS(6)=fn.RCS(abs(SMVCC(1))):"N.####": PRS(7)="Right"
20.531 if SMVCC(1+1) < 0 then PRS(20)="None" goto 20.532
      if SMVCC(1+1) > 0 then PRS(20)=fn.RCS(SMVCC(1+1)):"N.####": PRS(21)="Left" goto 20.532
      PRS(22)=fn.RCS(abs(SMVCC(1+1))):"N.####": PRS(23)="Right"
20.532 if SMVCC(1) < 0 then PRS(11)="None" goto 20.533
      if SMVCC(1) > 0 then PRS(11)=fn.RCS(SMVCC(1)):"N.####": PRS(12)="Up" : PRS(13)="Down" goto 20.533
      PRS(14)=fn.RCS(abs(SMVCC(1))):"N.####": PRS(15)="Open" : PRS(16)="Close"
20.533 if SMVCC(1+1) < 0 then PRS(17)=fn.RCS(SMVCC(1+1)):"N.####": PRS(18)="Up" : PRS(19)="Down" goto 20.542
      if SMVCC(1+1) > 0 then PRS(17)=fn.RCS(SMVCC(1+1)):"N.####": PRS(18)="Up" : PRS(19)="Down"
      PRS(24)=fn.RCS(abs(SMVCC(1+1))):"N.####": PRS(25)="Open" : PRS(26)="Close"
20.54 if SMVCF(1) < 0 then PRS(9)="None" : PRS(10)="--" goto 20.541
      if SMVCF(1) > 0 then PRS(9)=fn.RCS(SMVCF(1)):"N.####": PRS(10)="Close" : PRS(11)="--" goto 20.541
      PRS(12)=fn.RCS(abs(SMVCF(1))):"N.####": PRS(13)="Open" : PRS(14)="Close"
20.541 if SMVCF(1+1) < 0 then PRS(25)="None" : PRS(26)="--" goto 20.542
      if SMVCF(1+1) > 0 then PRS(25)=fn.RCS(SMVCF(1+1)):"N.####": PRS(26)="Open" : PRS(27)="Close"
      PRS(34)=fn.RCS(abs(SMVCF(1+1))):"N.####": PRS(35)="Open" : PRS(36)="Close"
20.542 if SMVCF(1) < 0 then PRS(14)="--" : PRS(15)="None" goto 20.543
      if SMVCF(1) > 0 then PRS(14)=fn.RCS(SMVCF(1)):"N.####": PRS(15)=fn.RCS(SMVCF(1)):"N.####": PRS(16)="Close" goto 20.543
      PRS(17)=fn.RCS(abs(SMVCF(1))):"N.####": PRS(18)="Open"
20.543 if SMVCF(1+1) < 0 then PRS(30)="--" : PRS(31)="None" goto 20.544
      if SMVCF(1+1) > 0 then PRS(30)=fn.RCS(SMVCF(1+1)):"N.####": PRS(31)="Open" : PRS(32)="Close"
      PRS(34)=fn.RCS(abs(SMVCF(1+1))):"N.####": PRS(35)="Open" : PRS(36)="Close"
20.55 if DISE% = 0 then \
      print "CALCULATED COUPLING MOVES" \
      print "*****"

```

```

print using "      ##.####  ##.####  /assign/  ##.####  ##.####":C0V00(1%),C0V00(1%),0655,C0V00(1%),C0V00(1%)
print
print using "      ##.####  ##.####  Variance  ##.####  ##.####":C0V00(1%),C0V00(1%),C0V00(1%),C0V00(1%)
print using "      /cm/      /qf/      /cm/      /cm/":PRS(1),PRS(2),PRS(3),PRS(4)

PRINT
if DIS#0 then print
posub 20.95
if DIS#0 then goto 20.92
if PNT#0 then I#F0M%
if PNT#3 then I#I#-2
on PNT# goto 20.61, 20.62, 20.66, 20.66, 20.66, 20.66

20.92 if DIS#1 then call fn.CLM$(M%,79.9,1,23,80)
if C0V00(1%)>0 then PRS(5)="Left" goto 20.621
if C0V00(1%)<0 then PRS(5)="Right" goto 20.621
PRS(5)="None"
20.621 if C0V00(1%)>0 then PRS(6)="Left" goto 20.622
if C0V00(1%)<0 then PRS(6)="Right" goto 20.622
PRS(6)="None"
20.622 if C0V00(1%)>0 then PRS(7)="Left" goto 20.623
if C0V00(1%)<0 then PRS(7)="Right" goto 20.623
PRS(7)="None"
20.623 if C0V00(1%)>0 then PRS(8)="Left" goto 20.624
if C0V00(1%)<0 then PRS(8)="Right" goto 20.624
PRS(8)="None"

20.624 if DIS#0 then goto 20.625
if left$(SNG(1%),2)="#M" then \
print "      Oil      H2 Gland      H2 Gland      Oil" \
:print "      =====      =====      =====      =====" \
:goto 20.625
if left$(SNG(1%),2)="#XC" then \
print "      Outboard Oil Inboard Oil      Inboard Oil Outboard Oil" \
:print "      =====      =====      =====      =====" \
:goto 20.625
print "      Oil      Steam Gland      Steam Gland      Oil" \
print "      =====      =====      =====      ====="
20.625 print "      -----HORIZONTAL-----"

print
print using "      ##.####  ##.####  Current  ##.####  ##.####":C0H00(1%),C0H00(1%),C0H00(1%),C0H00(1%)
print
print using "      ##.####  ##.####  Projected  ##.####  ##.####":C0P00(1%),C0P00(1%),C0P00(1%),C0P00(1%)
print using "      ##.####  ##.####  /assign/  ##.####  ##.####":C0M00(1%),C0M00(1%),PRES,C0M00(1%),C0M00(1%)
print
print using "      ##.####  ##.####  Variance  ##.####  ##.####":C0V00(1%),C0V00(1%),C0V00(1%),C0V00(1%)
print using "      /cm/      /qf/      /cm/      /cm/":PRS(5),PRS(6),PRS(7),PRS(8)

20.627 if DIS#0 then print : \
print "      ---Governor End-----      ---Collector End---"
print
posub 20.95
if DIS#0 then goto 20.628
if PNT#0 then I#F0M%
if PNT#3 then I#I#-2
on PNT# goto 20.61, 20.626, 20.66, 20.66, 20.66, 20.66

20.628 if DIS#1 then call fn.CLM$(M%,79.9,1,23,80)
if DIS#0 then print ABTS :print :print

if left$(SNG(1%),2)="#M" then goto 20.63
if left$(SNG(1%),2)="#P" then goto 20.63
if left$(SNG(1%),3)="#OCM" then goto 20.64
if left$(SNG(1%),3)="#XC" then goto 20.64
goto 20.8

20.63 if C0V00(1%)>0 then PRS(9)="None" goto 20.631
if C0V00(1%)<0 then PRS(9)=fn.RCS(C0V00(1%)):"#.####":PRS(10)="Left" goto 20.631
PRS(11)=fn.RCS(ABS(C0V00(1%)):"#.####":PRS(12)="Right"
20.631 if C0V00(1%)>0 then PRS(13)="None" goto 20.632
if C0V00(1%)<0 then PRS(13)=fn.RCS(C0V00(1%)):"#.####":PRS(14)="Left" goto 20.632
PRS(15)=fn.RCS(ABS(C0V00(1%)):"#.####":PRS(16)="Right"

20.632 if C0V00(1%)=99.1234 then goto 20.633
if C0V00(1%)>0 then PRS(17)="None" goto 20.633
if C0V00(1%)<0 then PRS(17)=fn.RCS(C0V00(1%)):"#.####":PRS(18)=fn.RCS(C0V00(1%)):"#.####":goto 20.633
PRS(19)="Remove":PRS(19)=fn.RCS(ABS(C0V00(1%)):"#.####":PRS(20)=fn.RCS(C0V00(1%)):"#.####":goto 20.633
PRS(21)=fn.RCS(ABS(C0V00(1%)):"#.####":PRS(22)="Right"

20.633 if C0V00(1%)=99.1234 then goto 20.634
if C0V00(1%)>0 then PRS(23)="None" goto 20.634
if C0V00(1%)<0 then PRS(23)=fn.RCS(C0V00(1%)):"#.####":PRS(24)=fn.RCS(C0V00(1%)):"#.####":goto 20.634
PRS(25)="Remove":PRS(25)=fn.RCS(ABS(C0V00(1%)):"#.####":PRS(26)=fn.RCS(C0V00(1%)):"#.####":goto 20.634
PRS(27)=fn.RCS(ABS(C0V00(1%)):"#.####":PRS(28)="Right"

20.634 if match("WEST",MFRS,1)>0 then goto 20.635
if left$(SNG(1%),2)="#M" then goto 20.71 else goto 20.72
20.635 if left$(SNG(1%),2)="#P" then goto 20.75 else goto 20.76

20.64 if C0H00(1%)=99.1234 then goto 20.641
if C0H00(1%)>0 then PRS(29)="None" goto 20.641
if C0H00(1%)<0 then PRS(29)=fn.RCS(C0H00(1%)):"#.####":PRS(30)="Left" goto 20.641
PRS(31)=fn.RCS(ABS(C0H00(1%)):"#.####":PRS(32)="Right"
20.641 if C0H00(1%)=99.1234 then goto 20.642
if C0H00(1%)>0 then PRS(33)="None" goto 20.642
if C0H00(1%)<0 then PRS(33)=fn.RCS(C0H00(1%)):"#.####":PRS(34)="Left" goto 20.642
PRS(35)=fn.RCS(ABS(C0H00(1%)):"#.####":PRS(36)="Right"

20.642 if left$(SNG(1%),3)="#XC" then goto 20.643
dim PRAS(14), PRBS(14)
for J#1 to 14
if M0(J#)=99.1234 then goto 20.643
if M0(J#)>0 then PRAS(J#)=fn.RCS(M0(J#)):"#.####":PRBS(J#)="Add" goto 20.643
PRBS(J#)=fn.RCS(ABS(M0(J#)):"#.####":PRAS(J#)="Remove"
20.643 next J#
if match("WEST",MFRS,1)>0 then goto 20.77
goto 20.73

20.644 dim PRAS(14), PRBS(14)
for J#1 to 14
if M0(J#)=99.1234 then goto 20.645
if M0(J#)>0 then PRAS(J#)=fn.RCS(M0(J#)):"#.####":PRBS(J#)="Add" goto 20.645
PRBS(J#)=fn.RCS(ABS(M0(J#)):"#.####":PRAS(J#)="Remove"
20.645 next J#
if match("WEST",MFRS,1)>0 then goto 20.78
goto 20.74

REM> OE = HP.

```

REN> WEST - GEN.
20 77 11 DISHWB then 3

```

print "
print "
print " Governor C=Coupling, X=Bearing, F=Frame Edge Collector"
print " End"
print "
print " H2 Cooler"
print using " 1) /mov/ /..NNN/ 6) /mov/ /..NNN/ " ;PRAS(1),PRAS(1),PRAS( 8),PRAS( 8)
print using " 2) /mov/ /..NNN/ 9) /Add / /..NNN/ /..NNN/ /..NNN/ " ;PRAS(9),PRAS(10),PRAS(2),PRAS(2),PRAS( 9),P
RBS( 9),PRAS(15),PRAS(16)
print using " 3) /mov/ /..NNN/ 10) /Add / /..NNN/ " ;PRAS(3),PRAS(3),PRAS(10),PRAS(10)
print using " C=---F---X 4) /mov/ /..NNN/ 11) /Add / /..NNN/ X---F---C " ;PRAS(4),PRAS(4),PRAS(11),PRAS(11)
print using " 5) /mov/ /..NNN/ 12) /Add / /..NNN/ " ;PRAS(5),PRAS(5),PRAS(12),PRAS(12)
print using " 6) /mov/ /..NNN/ 13) /Add / /..NNN/ /..NNN/ /..NNN/ " ;PRAS(11),PRAS(12),PRAS(6),PRAS(6),PRAS(12),
PRAS(13),PRAS(17),PRAS(18)
print using " 7) /mov/ /..NNN/ 14) /Add / /..NNN/ " ;PRAS(7),PRAS(7),PRAS(14),PRAS(14)
print "
print "
print " Skin Packs
print "
print " 1< 1> 2< 3< 4< 5< 6< 7< 8< 9<10<11>12>13>14>
goto 20.79

```

REM> WEST + EXE.

20.78 if DIS=0 then \

```

print "
print " Governor C=Coupling, X=Bearing, F=Frame Edge Collector"
print " End"
print "
print " Air Cooler"
print using " 1) /mov/ /..NNN/ 6) /mov/ /..NNN/ " ;PRAS(1),PRAS(1),PRAS( 8),PRAS( 8)
print using " 2) /mov/ /..NNN/ 9) /Add / /..NNN/ /..NNN/ /..NNN/ " ;PRAS(9),PRAS(10),PRAS(2),PRAS(2),PRAS( 9),P
RBS( 9),PRAS(15),PRAS(16)
print using " 3) /mov/ /..NNN/ 10) /Add / /..NNN/ " ;PRAS(3),PRAS(3),PRAS(10),PRAS(10)
print using " C=---F---X 4) /mov/ /..NNN/ 11) /Add / /..NNN/ X---F---C " ;PRAS(4),PRAS(4),PRAS(11),PRAS(11)
print using " 5) /mov/ /..NNN/ 12) /Add / /..NNN/ " ;PRAS(5),PRAS(5),PRAS(12),PRAS(12)
print using " 6) /mov/ /..NNN/ 13) /Add / /..NNN/ /..NNN/ /..NNN/ " ;PRAS(11),PRAS(12),PRAS(6),PRAS(6),PRAS(13),
PRAS(17),PRAS(18)
print using " 7) /mov/ /..NNN/ 14) /Add / /..NNN/ " ;PRAS(7),PRAS(7),PRAS(14),PRAS(14)
print "
print "
print " Skin Packs
print "
print " 1< 1> 2< 3< 4< 5< 6< 7< 8< 9<10>11>12>13>14>

```

20.79 for J=1 to 14
PRAS(J)=NULLS :PRAS(J)=NULLS

next J
dim PRAS(0), PRAS(0)
goto 20.95

REM> NO CASTING.

20.8 if DIS=1 then goto 20.86

for J=1 to 18
print
next J

20.85 print
if DIS=0 then print
goto 20.85
if DIS=0 then goto 20.86
if PNT=5 then J=PMN
if PNT=9 then J=14-2
on PNT goto 20.82, 20.88, 20.86, 20.86, 20.86, 20.88

20.88 for J=1 to 20 :PRAS(J)=NULLS :next J
if DIS=1 then if PNT=5 then goto 20.82
J=14-1
if J is FOM then goto 20.81
close IO
if DIS=1 then call fn.CLS(1,7) :goto 20.401

print ASTS
print BOTS
for IN=1 to 16
print
next IN
unspole
call fn.CLS(1,7)
goto 20.401

REM>-----1-----2-----3-----4-----5-----6-----7-----

REM> PRINT PAGE HEADER.

20.81 if IN=50% then goto 20.915

print ASTS
print BOTS

for IN=1 to 16 :print :next IN

20.915 print tab(24); "C2-TURBINE GENERATOR ALIGNMENT SYSTEM"
print
print tab(6); UNITS+lefts(BLANKS,70-len(UNITS+COMPS))+COMPS
print TOPS
print ASTS
print STOP1
print STOP2
print
return

REM>-----1-----2-----3-----4-----5-----6-----7-----

REM> DISPLAY SCREEN HEADER.

20.92 call fn.CLS(1,7) :call fn.CLS(MTS,70,0,1,20,90) :call fn.KOMN
print STOP1
print STOP2+lefts(BLANKS,72-len(OTOP2+TIMS+DATE))+TIMS+" "+DATE
return

REM>-----1-----2-----3-----4-----5-----6-----7-----

REM> PRINT CONTINUE ON SCREEN.

20.95 if DIS=0 then return

print

print "Press F1:PrvDer, F2:NxtDer, F3:PrvPg, F4:NxtPg, F5:PrvRpt, F6:NxtRpt or ESC";

20.951 IN=fn.INP
if IN=27 then PNT=8 :J=FOM :goto 20.88
if IN is 100 or IN is 101 then goto 20.951
PNT=IN-100
for IN=1 to 75
call fn.DEL
next IN
return