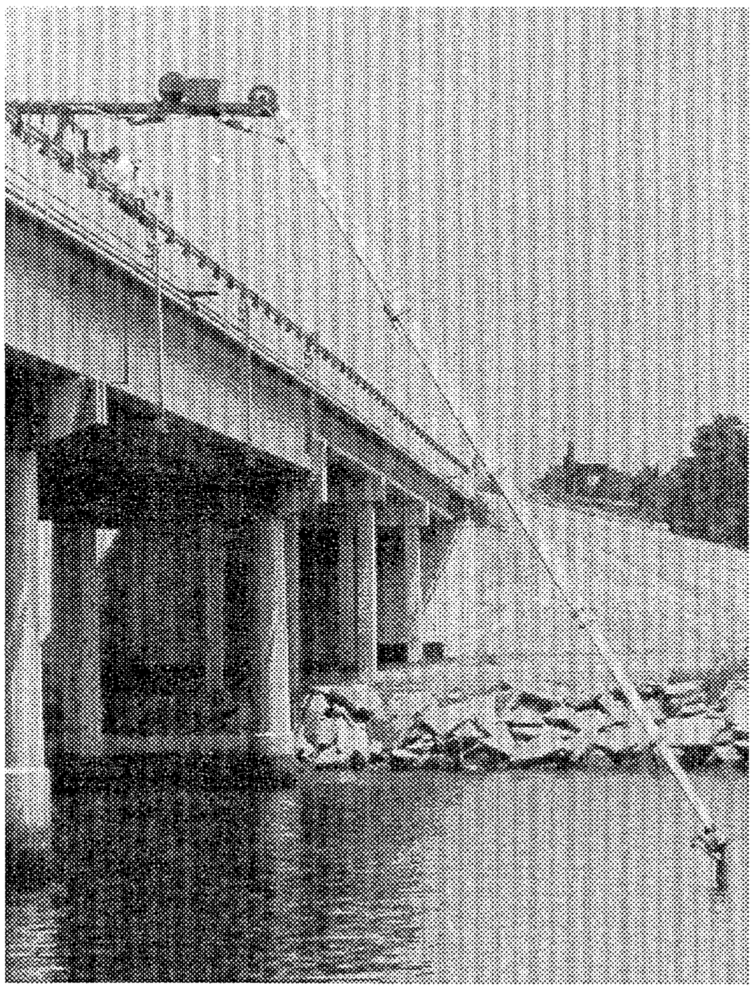
Remote Methods of Underwater Inspection of Bridge Structures

PUBLICATION NO. FHWA-RD-99-100

JULY 1999



U.S. Department of Transportation
Federal Highway Administration

Research, Development & Technology
Turner-Fairbank Highway Research Center
6300 Georgetown Pike
McLean, VA 22101-2296

FOREWORD

This report describes a study of remote methods of underwater inspection of bridge structures. It describes the conceptual design of two systems- one system was a relatively simple remotely controlled arm to deploy a sonar probe from a bridge deck, the other system was a remotely controlled work platform that could be used for above and below water inspection and repair of bridges. The first system which was fabricated and tested under flood conditions is called a bridge scour inspector and is considered a potential answer to the recent call for portable scour instrumentation to be used to inspect bridges during severe floods. This study was completed in 1995, but the final report from the original study included very detailed design drawings that were not considered necessary for general distribution. This version of the report focuses on the conceptual designs and performance of the bridge scour inspector during field tests, but the design details have been filed by the Federal Highway Administration for future reference as needed.

T. Paul Teng, P.E.
Director, Office of Infrastructure
Research and Development

NOTICE

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
FHWA-RD-99-100		

4. Title and Subtitle REMOTE METHODS OF UNDERWATER INSPECTION OF BRIDGE STRUCTURES	5. Report Date Julv 1999
	6. Performing Organization Code
7. Author(s) W. R. Bath	8. Performing Organization Report No.
9. Performing Organization Name and Address Sonsub Inc. 10905 Metronome Houston, TX 77043-2202	10. Work Unit No. (TRAIS) NCP 3D3C1-672
	11. Contract or Grant No. DTFH61-C-00060
12. Sponsoring Agency Name and Address Office of Infrastructure R&D Federal Highway Administration 6300 Georgetown Pike McLean, VA 22101-2296	13. Type of Report and Period Covered Final Report Sept. 1991 - December 1995
	14. Sponsoring Agency Code

15. Supplementary Notes

FHWA Contract Administrator: Carlos Rodriquez
Contracting Officer's Technical Representative (COTR): J. Sterling Jones
Contributors for publication version: David A. Bertoldi and Donna Pearson, GKY & Assoc.

16. Abstract

A portable trailer mounted bridge scour inspector was developed and tested under flood conditions for inspecting bridge scour in the vicinity of piers from the bridge deck. The bridge scour inspector features a remotely controlled arm to deploy a sonar probe in the water from a vehicle located on the bridge deck. The arm can be fixed in a cross section mode to take cross sections immediatedly upstream and downstream of a bridge or it can be maneuvered in the vicinity of piers including some distance under the bridge deck to inspect for deep local scour holes. Two servo motors constantly adjust the orientation of the sonar probe to maintain a vertical beam. A mechanical positioning system tracks the X,Y,Z coordinates of the sonar probe wherever it is in the water relative to some reference location on the pridge deck. The bridge scour inspector was field tested under low water condition in Texas and was field tested under extreme flood conditions in Georgia during the 1994 Tropical Storm Alberta flooding. Limitations of the bridge scour inspector include the max distance of 15 meters (50 feet) that the arm can be extended below the bridge deck to reach water, a minimum distance of 4.5 meters (15 feet) needed to avoid total submergence and possible malfunction of servo motor controls, and operation difficulties on truss type bridges because the arm must be retracted each time it passes an elevated obstruction. It is a viable bridge scour inspection device for many bridges during extreme flood conditions when it is too dangerous to put a boat in the water.

A more sophisticated remotely controlled work station was designed conceptually and is desribed in the report but it was not fabricated for testing. This system could be used above and below the water surface to inspect and repair bridges.

17. Key Words	18. Distribution Statement
Bridge Scour, -Instrumentation, Bridge Inspection,Robotics	No restrictions. This document is available to the public through the National Technical Information Service, Springfield, VA 22161

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	101	

Form DOT F 1700.7 (6-72) Reproduction of completed page authorized

SI* (MODERN METRIC) CONVERSION FACTORS

APPROXIMATE CONVERSIONS TO SI UNITS

Symbol	When You Know	Multiply By	To Find	Symbol
		LENGTH		
in	inches	25.4	millimeters	mm
ft	feet	0.305	meters	m
yd	yards	0.914	meters	m
mi	miles	1.61	kilometers	km
		AREA		
in^2	square inches	645.2	square millimeters	mm^2
ft^2	square feet	0.093	square meters	m^2
yd^2	square yards	0.836	square meters	m^2
ac	acres	0.405	hectares	ha
mi^2	square miles	2.59	square kilometers	km^2
		VOLUME		
fl oz	fluid ounces	29.57	milliliters	mL
gal	gallons	3.785	liters	L
ft^3	cubic feet	0.028	cubic meters	m^3
yd^3	cubic yards	0.765	cubic meters	m^3
NOTE: Volumes greater than 1 000 I shall be shown in m^3.				
		MASS		
oz	ounces	28.35	grams	g
lb	pounds	0.454	kilograms	kg
T	short tons (2000 lb)	0.907	megagrams (or 'metric ton")	Mg (or "t")
		TEMPERATURE (exact)		
°F	Fahrenheit temperature	5(F-32)/9 or (F-32)/1.8	Celcius temperature	°C
		ILLUMINATION		
fc	foot-candles	10.76	lux	lx
fl	foot-Lamberts	3.426	candela/m^2	cd/m^2
		FORCE and PRESSURE or STRESS		
lbf	poundforce	4.45	newtons	N
lbf/in^2	poundforce per square inch	6.89	kilopascals	kPa

APPROXIMATE CONVERSIONS FROM SI UNITS

Symbol	When You Know	Multiply By	To Find	Symbol
		LENGTH		
mm	millimeters	0.039	inches	in
m	meters	3.28	feet	ft
m	meters	1.09	yards	yd
km	kilometers	0.621	miles	mi
		AREA		
mm^2	square millimeters	0.0016	square inches	in^2
m^2	square meters	10.764	square feet	ft^2
m^2	square meters	1.195	square yards	yd^2
ha	hectares	2.47	acres	ac
km^2	square kilometers	0.386	square miles	mi^2
		VOLUME		
mL	milliliters	0.034	fluid ounces	fl oz
L	liters	0.264	gallons	gal
m^3	cubic meters	35.71	cubic feet	ft^3
m^3	cubic meters	1.307	cubic yards	yd^3
		MASS		
g	grams	0.035	ounces	oz
kg	kilograms	2.202	pounds	lb
Mg (or "t")	megagrams (or 'metric ton")	1.103	short tons (2000 lb)	T
		TEMPERATURE (exact)		
°C	Celcius temperature	1.8C + 32	Fahrenheit temperature	°F
		ILLUMINATION		
lx	lux	0.0929	foot-candles	fc
cd/m^2	candela/m^2	02919	foot-Lamberts	fl
		FORCE and PRESSURE or STRESS		
N	newtons	0.225	poundforce	lbf
kPa	kilopascals	0.145	poundforce per square inch	lbf/in^2

* SI is the symbol for the International System of Units. Appropriate rounding should be made to comply with Section 4 of ASTM E380.

(Revised September 1993)

TABLE OF CONTENTS

Chapter	Title	Page No.
I.	INTRODUCTION	1
II.	BRIDGE INSPECTOR FEATURES	5
	MANIPULATORARM	.
	HYDRAULIC SYSTEM	6
	ARMFUNCTION..	6
	UMBILICALWINCH	.
	CONTROLS	.
	BATTERY POWER	8
	TRAILER	8
	COMPUTER	10
	DATA ACQUISITION HARDWARE	11
	DATA ACQUISITION SOFTWARE	11
	MODESOFOPERATION	13
	STREAMBED CONTOUR MEASUREMENT	13
	ISOLATED PIER INSPECTION	13
	ONE-PASS CROSS-SECTION SURVEY	14
III.	FIELD EXPERIENCE	17
IV.	SUGGESTED ENHANCEMENTS	25
	TRUCK MOUNTING	25
	ELEVATION CHANGE MEASUREMENT	26
	SUBMERSIBLEHEAD	26
	VELOCITY MEASUREMENT	27
	ARM EXTENSIONS AND OFFSET ATTACHMENTS	27
	OTHER DATA ACQUISITION INTERFACES	29
	SIGNALPROCESSING	29
V.	PROBESYSTEM	31
	FUNCTIONS..	31
	TELESCOPE ARM AND PROBE PLATFORM POSITIONING	35
	COMPUTERDISPLAY	37
	OPERATION	38
	COMPUTER	39

TABLE OF CONTENTS
(Continued)

Chapter	Title	Page No.

APPENDICES

A. PARTS LIST 41
B. MECHANICAL DRAWINGS 47
C. HYDRAULICSCHEMATIC 51
D. ELECTRICAL SCHEMATICS 53
E. BRIDGE PROGRAM LISTINGS 59

LIST OF FIGURES

Figure No.	Page No.
1. Bridge Inspector on a low bridge	2
2. Bridge Inspector in operation	3
3. Bridge Inspector in travel mode	5
4. Probe head	7
5. Bridge Inspector trailer	9
6. Computer monitor and keyboard	10
7. Calculation method	12
8. Isolated pier inspection	14
9. Odometerwheel	15
10. Preparing for operations on the San Jacinto River Bridge	17
11. San Jacinto River bottom contour	18
12. San Jacinto River bottom profile	19
13. Flint River at Route 19 contour survey - pier no. 1	20
14. Flint River at Route 19 contour survey - pier no. 2	21
15. Flint River at Route 19 contour survey - pier no. 3	22
16. Oglethorpe Bridge over Flint River at Albany, GA - profile	23
17. Ocmulgee River at Highway 280 looking downstream - profile	24
18. Summation of moments diagram for truck-mounted system	26
19. Suggested offset extension for Bridge Inspector arm	28
20. Bridge Inspector arm with Columbus weight probe	30
21. Probe system	32
22. Probe system control consoles	33
23. Graphics display screen	34
24. Probe control system block diagram	36
25. Work platform electrical block diagram	40
26. Mechanical drawings of Bridge Inspector and parts list	47
27. Bridge Inspector hydraulic circuit	51
28. Consolette wiring schematic	53
29. Hand control box layout	54
30. Console wiring diagram	55
3 1. Bridge Inspector general arrangement diagram	57

I. INTRODUCTION

Development of the bridge inspection systems described in this report began with a desire to extend the current state of bridge inspection technology to include remotely operated systems. Remotely operated systems are highly desirable for scour measurement during floods and for zero-visibility inspection conditions. Two different systems were designed to meet the unique needs of these two applications. One system, called the Bridge Inspector, is a relatively simple machine that uses a sonar probe to measure the contour of the streambed. The other system, called the Probe System, is an extremely sophisticated, remotely controlled work system that can be used above and below the water surface to inspect and repair bridge piers. Both systems are described in this report. In addition, a prototype of the simpler scour measurement system was built and tested under actual field conditions.

Scour measurement is accomplished with the Bridge Inspector. The Bridge Inspector provides the capability to quickly and safely measure scour erosion around underwater bridge foundations during flood-stage conditions. Traditional methods of bridge inspection, such as divers and hand lines, are simply not adequate for the high flow rates encountered during floods. The intent of this work was to create a relatively low-cost prototype system to test the feasibility of using remote methods to measure bridge scour. The systems described in this report should be viewed as a field-usable test bed and not as a prototype for a commercial product.

The original plan for the prototype system was described as an "inexpensive, basic, and minimal system for the limited application of deploying a depth sounding transducer or fathometer." The original plan envisioned a small, lightweight machine that could operate from the bed of an ordinary pickup truck. Originally, the plan called for mounting the system on a skid frame that could be mounted on a boat or small barge as needed. In addition, the original plan called for a system that would be limited to bridges that were within 6 m (20 ft) of the high-water surface (figure 1). A simple fathometer would measure the water depth and the resulting data would be manually recorded. It was hoped that the system would be small and light enough to be powered by electric motors that could be operated by the truck's electrical system.

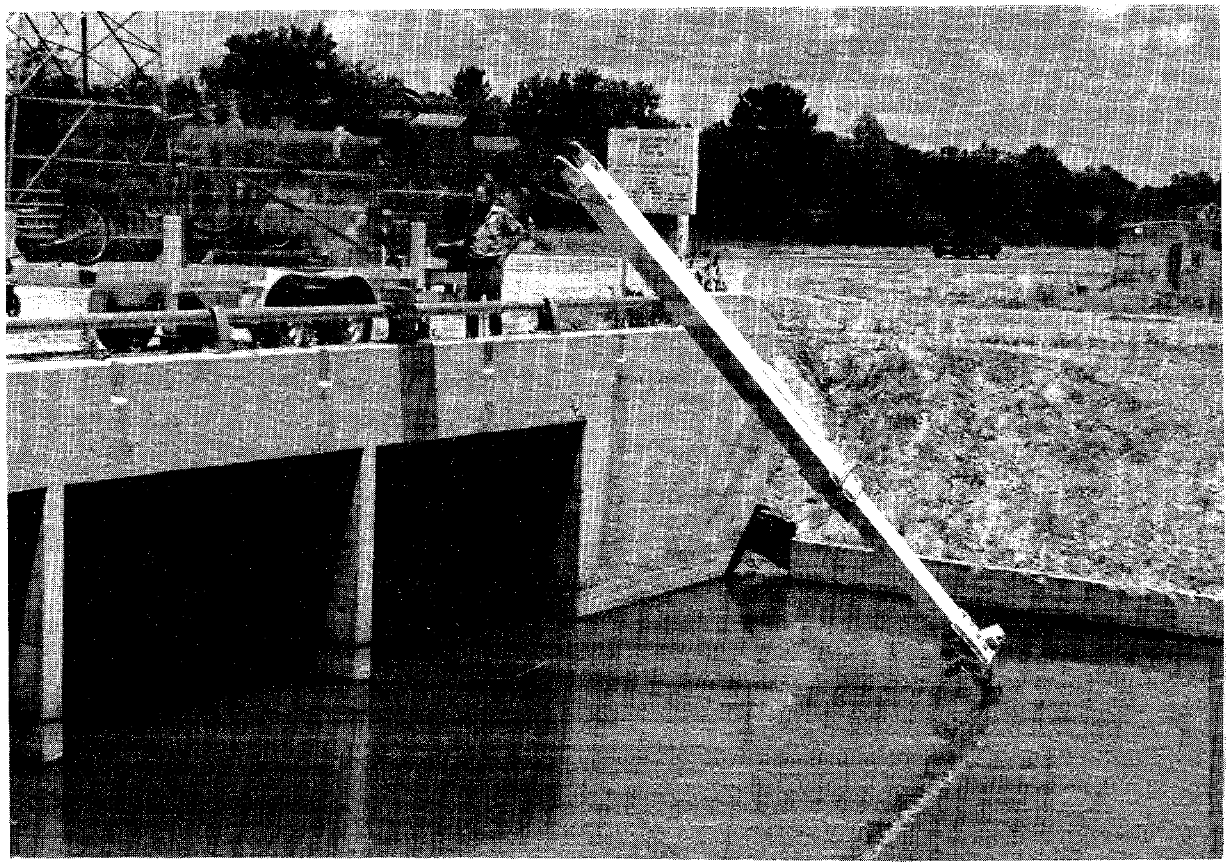


Figure 1. Bridge Inspector on a low bridge.

A preliminary review of equipment that would be useful to this project was conducted. At the same time, informal discussions were conducted with many bridge engineers and inspectors to determine the most advantageous specifications for the prototype. As the project progressed, it became apparent that a larger prototype would have greater flexibility and, therefore, wider application for field testing. The equipment for remote-sensing systems and data acquisition was found to be readily available for this application. Once the results of the preliminary review were analyzed, it was decided that the prototype would have the following increased capabilities:

- Operation from bridge decks up to 15 m (50 ft) above the water surface.
- Automatic data calculation and collection with onboard computer.
- Automatic measurement and tracking of position on the bridge deck.
- Dry-Land Altimeter for surface-profile data collection.
- Continuous operation while moving across the bridge.
- Hydraulically operated stabilizing and leveling wheels.
- Automatic calculation and data collection of elevation changes.

In its final form, the prototype Bridge Inspector (figure 2) features an 18-m (60-ft) long, hydraulically operated telescopic boom mounted on a dedicated tandem wheel trailer. The boom is fitted with a machine-readable sonar altimeter and a sophisticated, computer-operated control and data collection system. The Bridge Inspector system is fully described in chapter II and the electrical, hydraulic, and mechanical drawings for this system, along with the parts and program listing, are shown in the appendices.

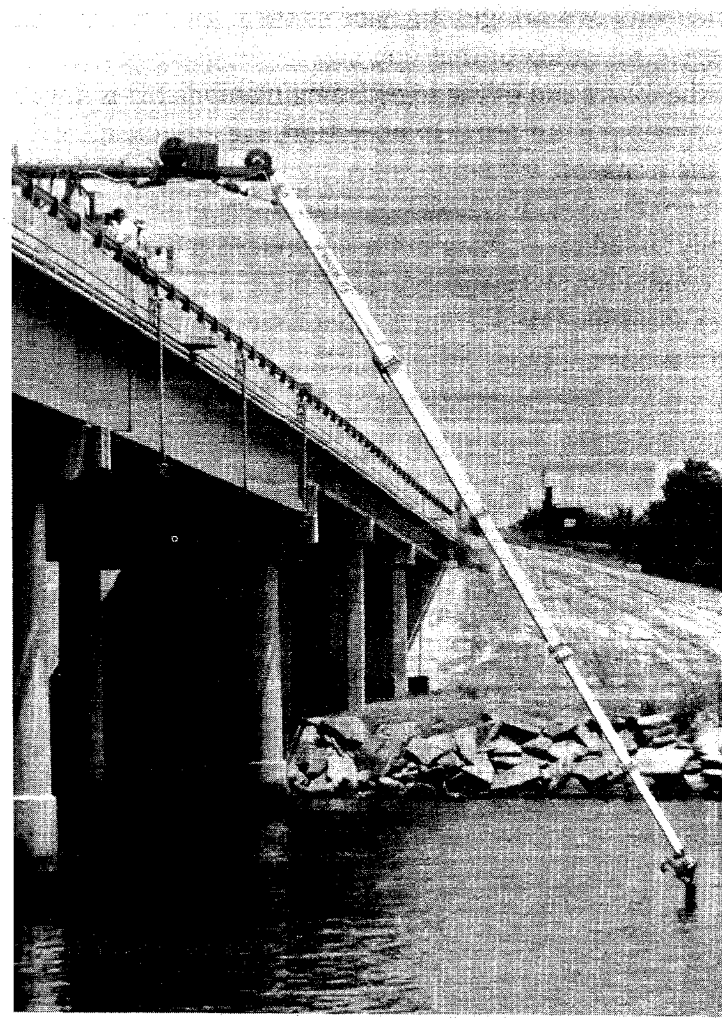


Figure 2. Bridge Inspector in operation.

A second part of this project was to design a much more elaborate inspection and repair system that could perform more elaborate functions. This system would be capable of working in zero visibility underwater to deploy and operate a multitude of devices, such as high-pressure water jet systems, power chisels, or nondestructive testing (NDT) equipment. This machine would combine reach, strength, and automation features to provide a robotic (remotely operated) system that could perform extensive inspection and repair functions. The system designed in this project is called the Probe System.

The Probe System is a truck-mounted system with a combination of articulated and telescoping arm sections capable of placing a sophisticated work package at any point from below the bridge deck to the streambed. The work package features state-of-the-art telerobotic manipulators, sensors, and low-light cameras. The probe system uses the cascaded manipulator principle of operation where successively smaller and more dexterous manipulators are used to move the tool to the workpiece. The Probe System begins with an articulated boom truck normally used by bridge inspection personnel as an elevating platform for inspecting the underside of a bridge. In place of the inspector's platform, the Probe System has a telescoping manipulator that operates from the lower end of the articulated boom. Mounted on the lower end of the telescoping manipulator is a work package with a dexterous, telerobotic manipulator arm. In operation, the articulated and telescoping manipulators act as the transport system to move the dexterous manipulator work package into position. Although this system was designed to operate underwater down to the streambed, it is equally suited to working in the air underneath the bridge deck. The Probe System could easily conduct visual inspection as well as many other cleaning, painting, and repair activities under the bridge deck. The Probe System and its sophisticated control system for zero-visibility operations are more fully described in chapter V of this report.

II. BRIDGE INSPECTOR FEATURES

The Bridge Inspector is a battery-powered, self-contained, trailer-mounted inspection system; The Bridge Inspector has a hydraulically operated arm that can reach over the side of a bridge and place a sonar probe into the water below the bridge. An onboard computer calculates the position of the sonar probe based on the angle of the boom and the distance between the boom pivot and the probe. The arm is manually controlled by the operator while the data are automatically collected by the onboard computer. The system has additional sensors to allow the computer to calculate the position of the boom pivot with reference to a known position (normally one end of the bridge). With the arm folded for travel (figure 3), the Bridge Inspector is 2.4 m (8 ft) wide, 7.6 m (25 ft) long and 2.4 m (8 ft) high. The trailer configuration was chosen so that the system could be towed with a highway department dump truck.

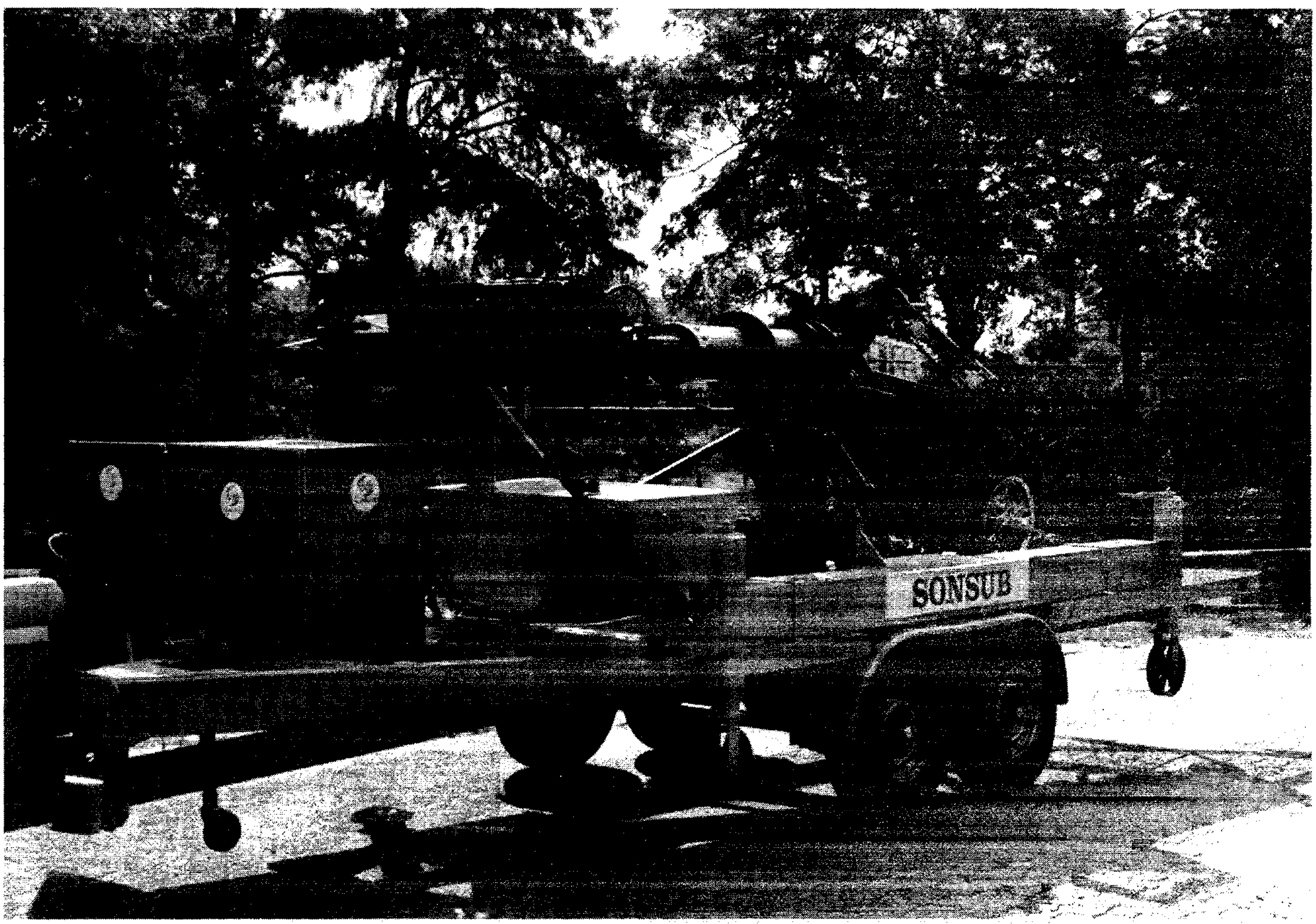


Figure 3. Bridge Inspector in travel mode.

MANIPULATOR ARM

The Bridge Inspector's arm is part of a S-degrees of freedom (DOF) (five-function) manipulator that is hydraulically operated to provide 60 degrees of movement parallel to the bridge deck, 30 degrees out from the deck and 30 degrees underneath the bridge. The 18-m (60-ft) long manipulator arm is mounted on a 3-m (10-f-l) long, folding horizontal support (elbow) member that is mounted in a bearing housing (shoulder) that is, in turn, mounted on a 120-degree rotation turntable. The turntable, the folding elbow, and the shoulder bearing provide the capability to move the pivot point of the arm from the travel position over the trailer deck to a point 1.8 m (6 ft) off the side of the trailer. This capability allows the manipulator arm to reach out over the side of the bridge and down to the water. On a bridge 15 m (50 ft) above the water, the arm can circumscribe an area 15 m (50 ft) wide from 7.5 m (25 ft) underneath the bridge to 7.5 m (25 ft) out from the bridge deck.

HYDRAULIC SYSTEM

Hydraulic power for each function of the manipulator (except telescoping the arm) is supplied by a 750-W (1-hp) electric motor driving a gear pump. The manipulator functions are powered by hydraulic cylinders (rams) that are pivot-connected to the mechanical members. The hydraulic cylinders are operated by a modular control valve mounted directly on the hydraulic power unit. The control valve uses solenoid-operated selector valves in a series circuit to select the function to be operated and the direction that the function will move. A variable volume-flow control valve then adjusts the flow of hydraulic fluid to the selected function based on a variable voltage signal from the control computer. Excess flow from the constant-volume gear pump is diverted back to the reservoir by the flow control valve.

ARM FUNCTION

A sonar (altimeter) probe is mounted on the lower end of the manipulator arm. The manipulator arm's function is to position the sonar probe in the water. The arm also acts as a measurable link to the probe so that the probe's position with respect to the arm pivot can be calculated and stored in the onboard computer. The aluminum manipulator arm is fabricated in seven segments. Each arm segment is mounted on polyethylene rollers. The telescopic manipulator arm is gravity-extended and cable-retracted by a variable-speed electric winch. The winch is operated by the system's computer to automatically maintain the vertical position of the probe with respect to the water surface. Float switches on the probe housing provide the command signal to extend or retract the arm as necessary to maintain a constant depth of the probe in the water. The sonar probe is mounted on a 2-DOF manipulator (figure 4) so that the probe can be held vertically against the flow of the water. The probe manipulator is powered by two electric linear actuators. The linear actuators are positioned at right angles to each other so that they can drive the probe 30 degrees away from vertical in any direction, with respect to the arm. The electric actuators are controlled by gravity-activated level switches mounted on the probe housing. The probe is automatically controlled by the float switches to project about 150 mm (6 in) into the water and is held

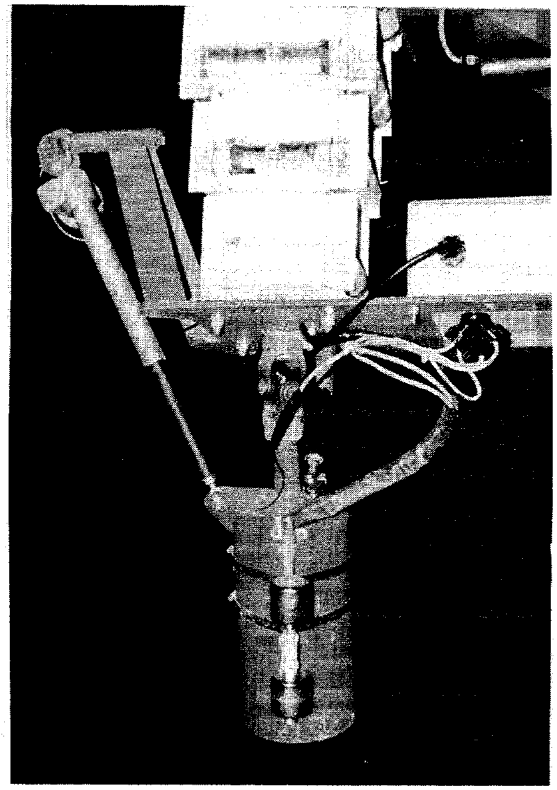


Figure 4. Probe head.

vertical by the level switches. The 150-mm (6-in) diameter probe pierces the surface of the water approximately 150 mm (6 in) so that hydrodynamic drag is minimized.

UMBILICAL WINCH

The cable that retracts the telescopic arm is called the umbilical cable. The umbilical cable has seven electrical conductors surrounded by a wire-rope strength member. The umbilical conductors are the connecting cable between the computer and the probe package. The probe package includes the sonar probe, the float switches, and the linear-actuator power supply.. The umbilical cable is reeled on a specially designed winch. The winch drum is mounted directly on the output shaft of a right-angle gearbox. The winch is powered by a 750-W (1-hp) electric motor. The winch drum is fitted with a seven-conductor slip-ring assembly for connection to the umbilical cable. The cable passes over a deflection sheave on top of the manipulator arm support member (elbow) before being connected to a pad eye on the probe package at the lower end of the boom.

CONTROLS

A manual joystick control commands the Bridge Inspector's computer to operate the variable-flow hydraulic functions that move the arm. The joystick is mounted in a hand-held mini-console that is connected to the computer by a 4.6-m (15-A) tether. The tether allows the operator to stand at the bridge rail or ride in the truck during operations. The mini-console has position and water-depth displays that enable the operator to maneuver the probe around submerged objects and other areas of interest. The mini-console also contains controls to switch between manual and automatic control of the winch so that the arm can be easily deployed from the travel position. Another control switches the joystick from operational mode to deployment mode. In deployment mode, the joystick and winch controls are used to operate the elbow, shoulder, and rotate functions.

BATTERY POWER

A 24-V direct current (dc) onboard, rechargeable battery pack supplies electric power to operate all the functions, including the hydraulic pump, the winch, and the computer. The battery pack provides several hours of operation (depending on the type of use) before recharging is necessary. A heavy-duty battery charger is included to recharge the batteries. The onboard battery charger allows the use of an auxiliary engine-driven generator to extend the operating time indefinitely. The battery pack proved to be very convenient because it allows the operator to use the computer at any time without the need to connect it to an outside power supply or to start up an engine generator.

TRAILER

The trailer is specially designed for this application (figure 5). The trailer uses standard, spring-mounted tandem axles. The wheels are equipped with hydraulic brakes that are operated by a mechanical load-transfer device in the trailer hitch. The trailer frame is welded-steel construction with a wide-flange profile frame and an expanded metal deck. The manipulator arm is mounted on a centrally located post with a tapered roller bearing at the top and a polyethylene plain bearing at the bottom. The manipulator post is welded to a torsional support structure that is part of the trailer frame. Hydraulically deployed steel rollers on each corner of the support structure (trailer frame) provide a stable base for inspection operations. The rollers on the right side of the trailer also provide a fulcrum point that improves the overturning-moment capability and reduces the size of the counterweight. A fixed-mounted counterweight on the left side of the trailer is required to balance the weight of the manipulator arm and the overturning moment produced by hydrostatic drag of the sonar probe in the water-flow stream. The rollers are mounted on heavy-duty telescoping struts. A manually operated control valve is provided at the right-front corner of the trailer to deploy the rollers. In addition to removing the heavy overturning-moment reaction from the spring-mounted trailer wheels, the rollers also provide a fixed attitude of the trailer frame with respect to the bridge deck. The rollers are adjusted during assembly to hold the trailer parallel to the bridge deck to facilitate measurement of elevation changes as the system is moved across the bridge.

The Bridge Inspector is intended to operate in flood conditions. A significant hazard of operating in flood-stage conditions is large tree limbs, lumber, or similar floating debris being carried along with the stream flow. Under certain conditions, a tree limb could easily snag on the sonar probe or manipulator arm in a way that would greatly increase the loads acting on the manipulator. Although the operator should be on guard to move the arm and probe out of

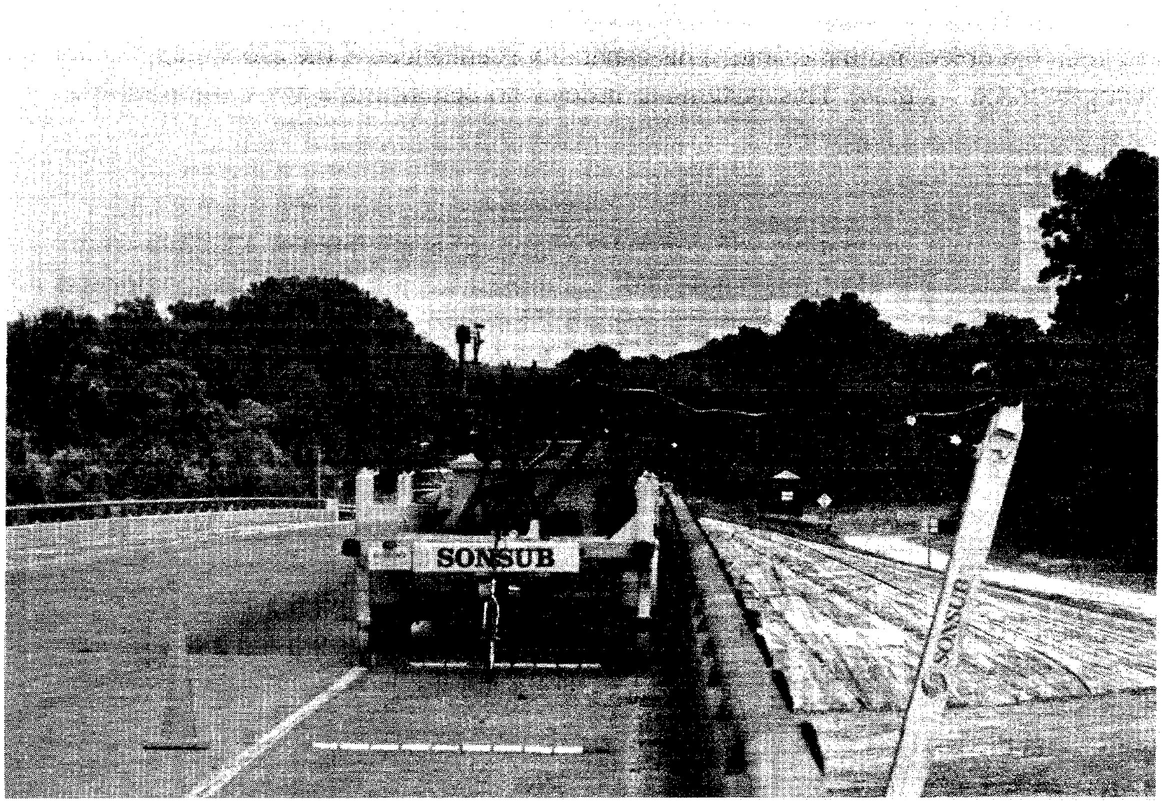


Figure 5. Bridge Inspector trailer.

the way of these hazards, the possibility still exists.' The Bridge Inspector is protected against this hazard by two safety systems, one built into the hydraulic system and one on the mounting of the probe. Although there is no guarantee that these systems will prevent damage to the Bridge Inspector in every case, they do increase the chances that the machine will survive the encounter.

The hydraulic system protects the arm by controlling the hydraulic cylinders that operate the arm extend and shoulder rotate functions. These hydraulic cylinders are hydraulically locked in position by counterbalance (CB) valves. The CB valve acts as a load brake by holding the arm cylinder in position if there is a loss of hydraulic pressure. These valves are spring-loaded to provide fail-safe protection for the suspended loads. The CB valves are mounted directly on the hydraulic cylinder port connections so that a broken hydraulic hose cannot cause the arm to fall. The CB valves are designed to allow large, unintentional, external forces acting on the arm to release the position-locking feature of the CB valve in a controlled manner. The CB valves are cross-port connected so that larger-than-normal external forces acting on the arm will allow the arm to move to relieve the force.

The sonar probe is mounted on a universal joint that is bolted to the lower end of the manipulator arm. The probe is held in a vertical position by the linear actuators. The linear actuators are mounted on a bracket that provides a shearing point for the mounting screws. These screws are designed to break away if a high side load is applied to the probe housing. This facility is provided to allow the probe to swing freely to release a floating object rather than to keep the object from passing. This capability is enhanced if the probe depth is limited to small penetration depths. This system has little or no benefit if the arm is submerged.

COMPUTER

The onboard IBM-compatible 386 computer (figure 6) is mounted on the right-front corner of the trailer. The electronic angle and rotation sensors on the various components of the system are connected to the computer by hard-wired cables. The computer is housed in an environmentally protected cabinet mounted on the right side of the trailer near the trailer tongue. The computer's video screen displays selected operating parameters and collected data. The computer stores the data and calculations on an internal hard disk. A 90-mm (3.5-in) floppy drive for data transfer and a mouse for operating the mapping software are also included in the computer's weather-tight enclosure. A standard computer keyboard is provided to allow the operator to keep field notes in the computer memory. The keyboard and mouse are located in a pull-out tray inside the computer cabinet. The sonar position display on the computer monitor maintains a visible track of probe travel and bridge profile. The probe-travel tracking is used by the operator to verify complete coverage of the inspection area. The bridge-profile display provides visible confirmation that the contour of the bridge and the land below the bridge are being tracked.

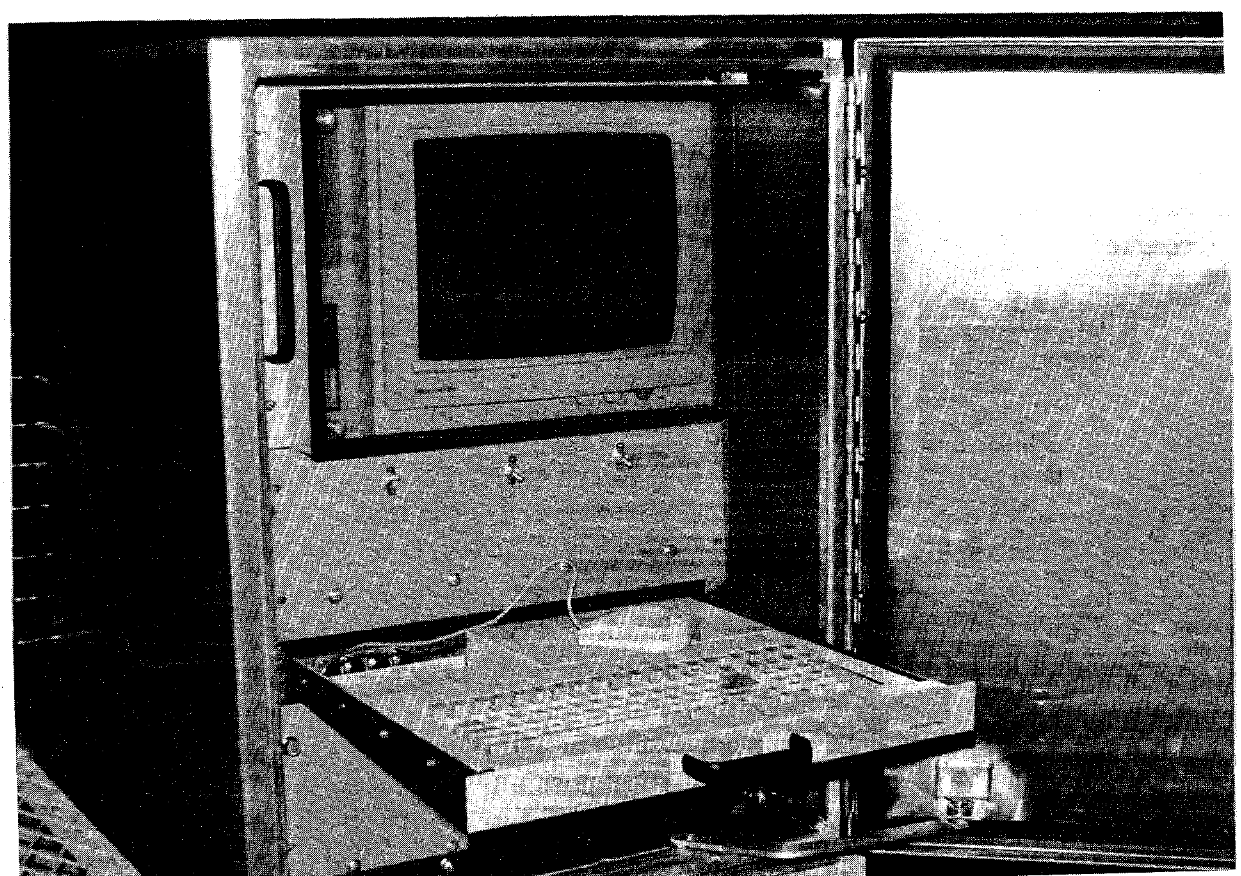


Figure 6. Computer monitor and keyboard.

DATA ACQUISITION HARDWARE

The data acquisition system is based on a 483-mm (19-in) Electronic Industrial Association (EIA) rack-mounted computer with an integral 356-mm (14-in) video graphics adaptor (VGA) cathode ray tube (CRT) display. The computer is a 33-MHz Intel 386 machine that is IBM Industrial Standards Architecture (ISA) compatible. The computer system consists of standard peripheral devices such as a hard drive; a 90-mm (3.5-in), 1.44-Mb floppy drive; a parallel printer port; and two serial communication ports.

The data acquisition interface to the computer is two National Instruments PCLab multi-function input/output (I/O) boards that slot into the computer's ISA bus. These boards are configured as 16 single-ended analogue inputs, 2 analogue outputs, 24 digital I/O devices, and 2 counters. The digital I/O device is hardware configured as 8 transistor-transistor logic (TTL) inputs and 16 TTL outputs. All inputs and outputs are accessed through two custom interface cards. These cards provide high-level driver interface to the system driver hardware (such as the solenoid valves). These cards also provide a convenient tie-in point to the data acquisition card for inputs from the various sensors and from the user joystick and control switch console.

The control switch console has various switches that enable or disable automatic functions, such as probe auto-insertion depth, and provide mode selection logic input to determine the purpose of the joystick inputs. The switch inputs are at 0- to 5-V logic levels that directly interface to the data acquisition board's digital input channel. The joystick outputs are tied directly through the interface board to the analogue inputs on the data acquisition cards. Three of the four available analogue outputs are scaled in the computer and drive the liquid crystal display (LCD) meters on the control mini-console. The remaining analogue output is dedicated to the hydraulic proportional speed control valve. The speed control output is first buffered through a high-current driver operational amplifier.

Various sensors also tie into the acquisition board through the interface card. These sensors consist of three inclinometer sensors, two optically encoded rotary sensors, the sonar, and the boom stop switch. Two of the inclinometers are used to determine the boom angles for both pitch and yaw. These angles, in conjunction with the boom length counter, determine the probe head position relative to the upper pivot of the manipulator arm. The third inclinometer determines the trailer attitude along the longitudinal axis. This sensor input, in conjunction with the trailer odometer, is used to calculate both the trailer's travel distance and its change in the vertical plane over the distance traveled. These calculations are used to generate the physical bridge profile, as well as to help locate the probe's position in the real world (relative to the start of the bridge). The remaining sensor is the laser distance sensor. This sensor is interfaced directly to the computer via a serial communication port. This sensor is used to profile the dry land below the bridge.

DATA ACQUISITION SOFTWARE

The data acquisition software was written and compiled in Microsoft's QuickBasic, version 4.5. This custom software utilizes the library routines from National Instruments

(Nidaq, version 4.4) to access the Lab PC Plus multi-function I/O cards. The software's purpose is to: (1) assist in calibration of the sensors; (2) poll the sensors, joysticks, and switches during operations and control some of the various functions via feedback from the sensors; (3) scale the input readings from voltage to the real-world data types that the voltages represent; (4) display the inputs in both text and graphical on-screen formats; and (5) log the data to disk files. The calculation method (figure 7) uses standard trigonometric functions to compute the probe position with respect to the arm pivot.

BL = Arm Length (Arm pivot to sonar probe)
Vd = Vertical **Distance** to water **surface**
Pang = Pitch Angle (Perpendicular to bridge)
Yang = Absolute Angle (Parallel to lxidge)
Aang = Absolute **Angle (Armtovertical centerline)**
Pdeg = Pitch Angle (degrees)
Ydeg = Yaw Angle **(degrees)**
δDist = Distance from vertical **axis to arm**
Pang = $(\Pi/180)$(Pdeg)
Yang = $(\Pi/180)$(Ydeg) **for Vd = 1**
mist = $\tan(Pang)^2 + \tan(Aang)^2$
Aang = $Atan(\sqrt{\tan}(Pang)^2 + \tan(Aang)^2$
Vd = cos(Aang)(BL)
Xdisp = $\sqrt{\tan}(Pang)(Vd)$
Ydisp = tan(Yang)(Vd)

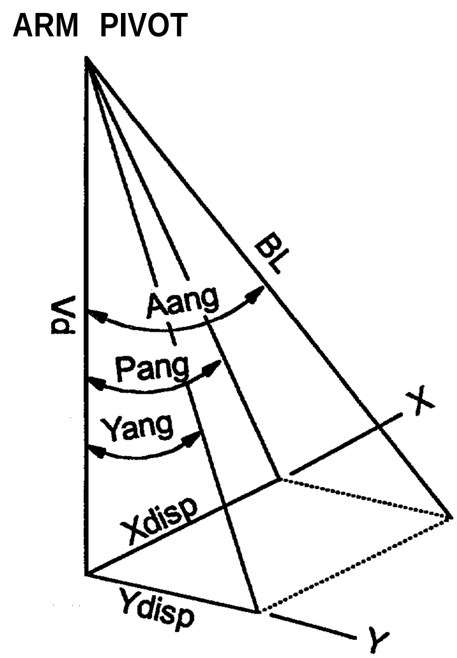


Figure 7. Calculation method.

The custom software is included in the systems computer, both in compiled executable form and in source-code form. The Nidaq library files are also included with the system in source-code format,

The disk files are logged in a space-delimited ASCII format. This format is the most universally accepted data format available and, as such, can be imported directly into a vast range of commercially available computer software packages for data manipulation such as presentation and graphing.

The Bridge Inspector software also includes Delta Graph 4, a commercially available program for generating graphs from the logged data files. This program (as well as a host of other available software packages) can import the ASCII logged data file and generate a variety of graphic displays from the data.

MODES OF OPERATION

The Bridge Inspector can be operated in any of three modes. The operational modes are "streambed contour measurement," "isolated pier inspection," and "one-pass cross-section survey." Streambed Contour Measurement is similar to the original plan for the bridge inspection system. In this operation, the Bridge Inspector is positioned on the bridge deck at a point near the center of the area to be inspected. The manipulator arm is deployed and the operator sweeps the arm horizontally (parallel to the bridge deck) and vertically (perpendicular to the bridge deck) to move the probe over the area to be surveyed. Isolated Pier Inspection is a manual operation usually conducted near a suspect pier. In this operation, the operator uses the depth readout on the control mini-console and visual control of the arm to inspect specific areas of interest around an individual pier, without reference to the coordinates of the probe. One-Pass Cross-Section Survey is a method of quickly obtaining a cross section of the bridge both above and below the water line. In, this operation, the Bridge Inspector is brought to the starting end of the bridge. The odometer measurement wheel and dry-land altimeter are deployed and the computer is initialized. The Bridge Inspector is then towed across the bridge until it is over the water. The manipulator arm is then deployed to place the sonar probe directly below the trailer and at a specific distance out from the bridge. The Bridge Inspector is then towed across the bridge until the other bank (side) of the water is reached. The sonar probe is then retracted and the tow is continued to the end of the bridge.

STREAMBED CONTOUR MEASUREMENT

The computer uses two angle sensors and a rotation sensor to calculate the position of the sonar probe. These sensors provide the angle of the arm in two directions and the length of the arm so that the computer can triangulate the position of the probe with respect to the arm pivot. The angle sensors are mounted at right angles to each other, near the pivot end of the arm. The rotation sensor is mounted on a deflection sheave at the arm pivot. The computer reads the rotation of the deflection sheave and calculates the length of the arm (distance from the pivot to the sonar probe) based on the fixed diameter of the deflection sheave. The computer automatically records the angle of the arm, the position of the probe, and the depth of the water at that point when the horizontal position of the probe moves by a preselected increment. The manually selected increment is field-adjustable by the operator to match specific inspection conditions.

ISOLATED PIER INSPECTION

The Bridge inspector can be used to inspect the area around a particular pier or other submerged feature below a bridge (figure 8). In this mode, the operator moves the arm around the area of interest while using the depth display on the hand-held mini-console to inspect or identify the underwater feature. Many different types of sensors (or cameras) could be used for this type of inspection. Cameras and other sensors are addressed in chapter IV of this report.

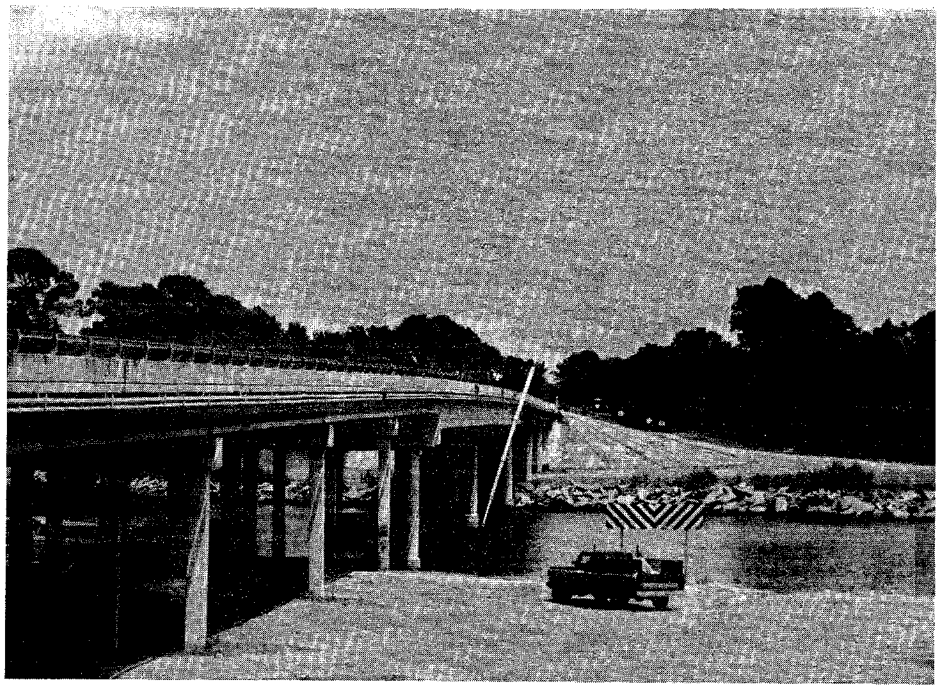


Figure 8. Isolated pier inspection.

ONE-PASS CROSS-SECTION SURVEY

The Bridge Inspector uses an electronic inclinometer and a rotation sensor (odometer) to collect data about the position of the trailer with regard to the starting (initialization) point on the bridge. The rotation sensor is mounted on an odometer wheel (figure 9) that is hinge-mounted to the rear of the trailer. The computer uses data from the rotation sensor to calculate the distance traveled by the trailer. The inclinometer is mounted directly on the trailer frame and is used to measure the angle of the trailer as it travels over the bridge. The computer uses these sensors to calculate length and crown of the bridge deck. The computer averages the change in angle of the inclinometer over a 3-m (10-ft) long distance to determine if the trailer is going up or down an inclined bridge deck. The computer plots the change in elevation as a function of the distance traveled from the starting point to give a profile of the bridge deck.

The dry-land altimeter is used in conjunction with the bridge profile tracking to add the profile of the land below the bridge to the bridge profile data. The dry-land altimeter is mounted on a manually deployed boom that is hinge-mounted to a vertical post on the right side of the trailer near the rear stabilizer wheel. The dry-land altimeter is a laser optical distance measurement sensor with a useful range of 0 to 20 m (0 to 65 ft). The sensor operates by emitting a collimated laser beam that is reflected from the target surface and collected by the sensor. Range is converted to a frequency that may be precisely measured.

The altimeter uses a 6-mW infrared laser diode to provide very high-precision, short-term repeatability over diffuse reflective surfaces.

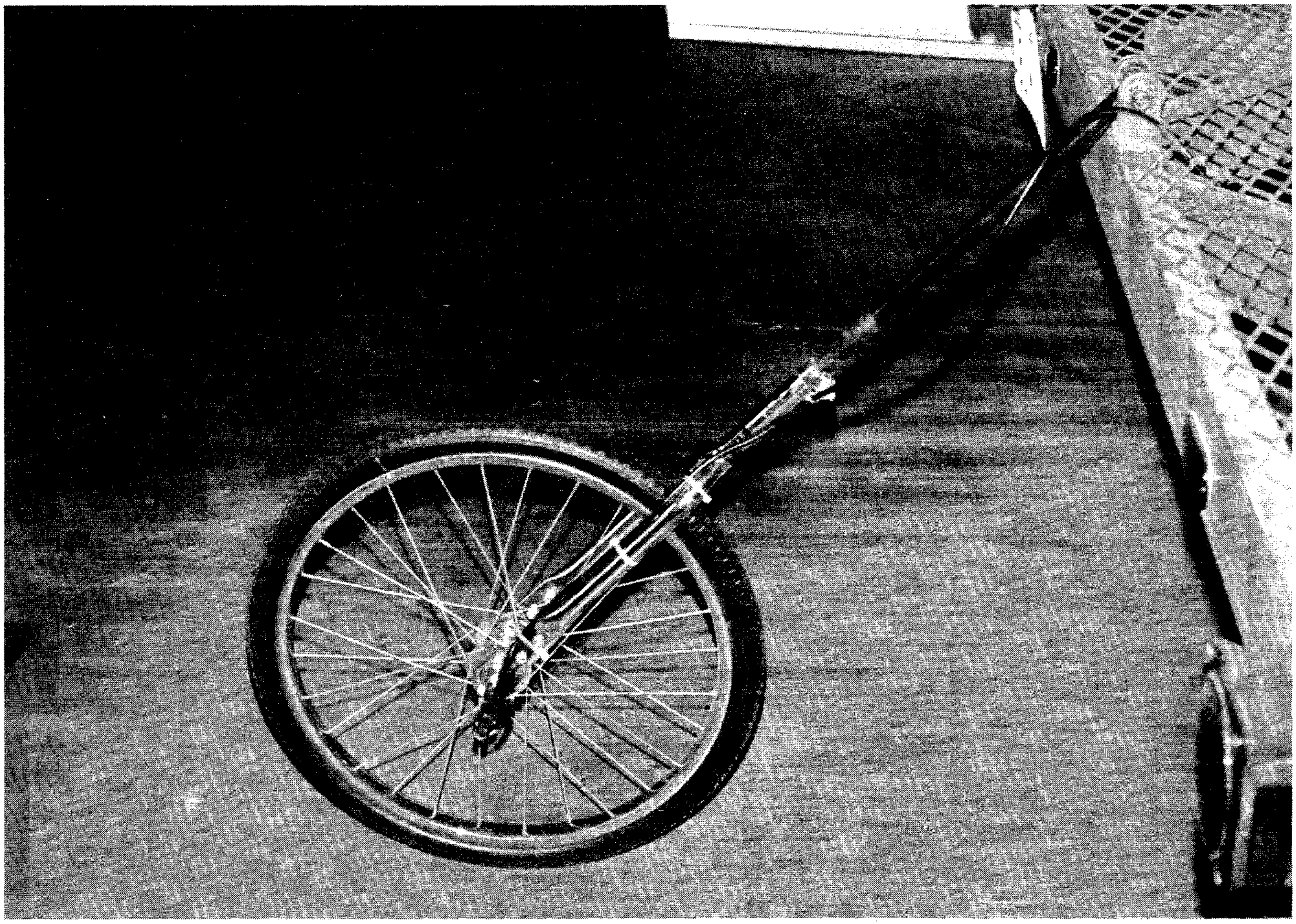


Figure 9. Odometer wheel.

The operator uses the joystick control to sweep the boom under the bridge, around the foundation, and across the inspection area. The computer keeps track of the position of the probe and records the water depth at each point. The data file collected can be displayed in spreadsheet format or as a contour map. In bridge profile mode, a calibrated trailing-wheel odometer keeps track of the system's position as it is towed across the bridge. An acoustic distance sensor mounted on a separate boom measures the distance to the land and water surface below the bridge. An inclinometer on the trailer frame measures the inclination of the bridge. The computer records this data and displays a profile of the bridge on the computer screen and logs it into the computer memory. Scour survey and bridge profile can be conducted simultaneously so that one trip across the bridge will produce a complete data file.

III. FIELD EXPERIENCE

The Bridge Inspector was demonstrated in Houston, Texas, and Memphis, Tennessee, and was field tested during the 1994 floods in Georgia. The field tests in Georgia revealed several limitations in the current design. These limitations are addressed in chapter IV of this report.

The Bridge Inspector was first demonstrated on the San Jacinto River bridge on U.S. Highway 90 near Houston, Texas. The San Jacinto River bridge is a two-lane concrete design that is approximately 14 m (45 ft) above the water (figure 10).

Figure 10. Preparing for operations on the San Jacinto River bridge.

The Bridge Inspector was easily maneuvered near the side of the bridge and the sonar probe was deployed into the water. Operation of the arm from the hand-held mini-console proved to be easy and convenient. The system was easily switched from deployment to operational mode. Movement of the manipulator arm from the travel position to the operational position proved to be intuitive enough that no special training is required. Once the arm is in the operational position, the operator uses a single toggle switch to change the joystick control from deployment to operation. In the operational mode, the operator can then position the arm at a convenient location before using the winch control to lower the probe into 'the water. A second toggle switch is used to change control of the winch from manual to automatic mode. The initial data collection increment was found to provide an excessively high number of data points. The data increment was reset to 0.3 m (1 ft) and a streambed contour measurement survey was conducted. The results of this survey were plotted using the Bridge Inspector's mapping software to produce a bottom contour map (figure 11).

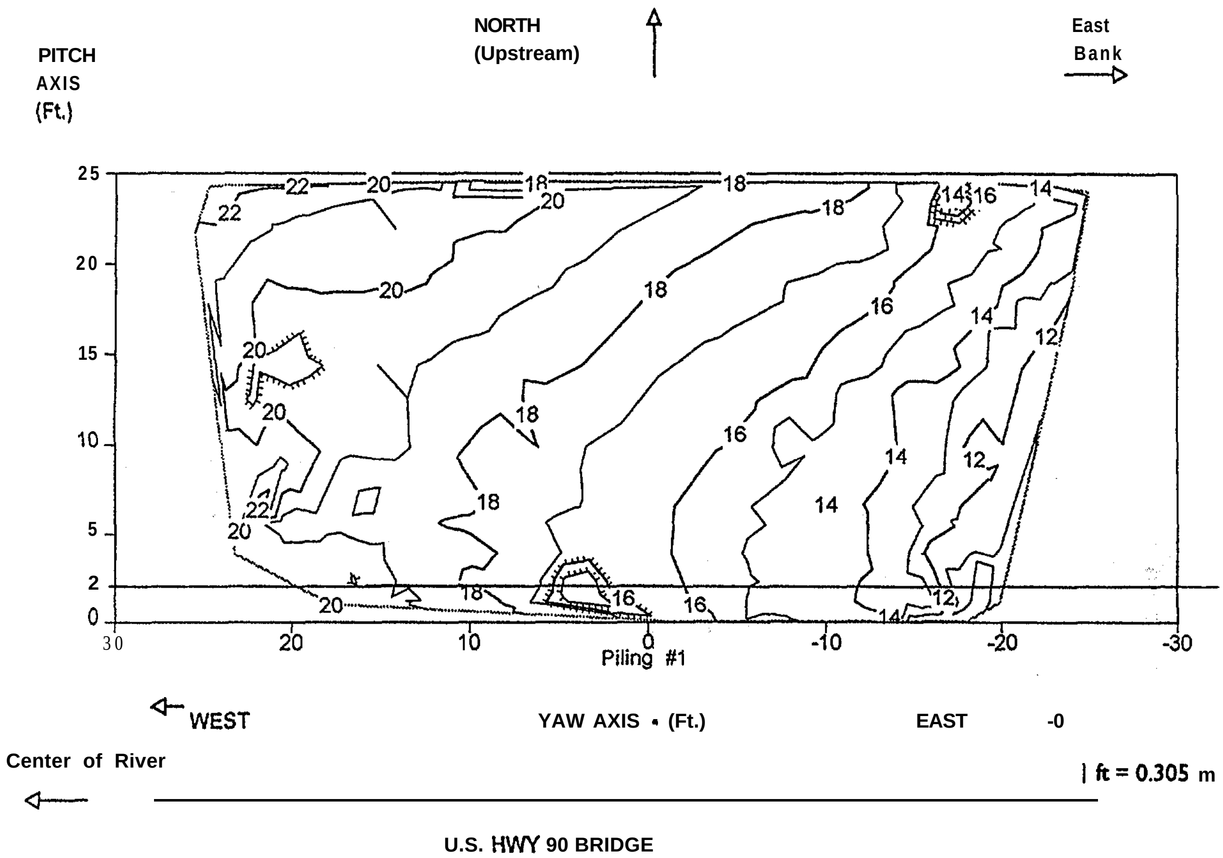


Figure 11. San Jacinto River bottom contour.

The data from this survey was also used to make a bottom profile (figure 12) of the contour map for a line parallel to and 0.6 m (2 ft) out from the bridge deck.

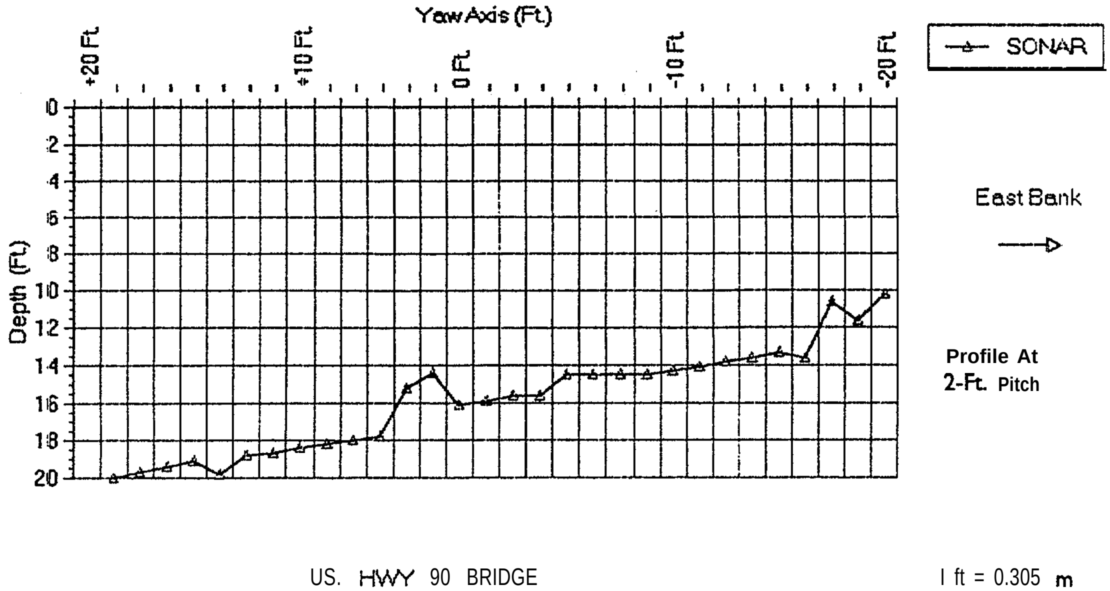


Figure 12. San Jacinto River bottom profile.

The Bridge Inspector was also demonstrated at the Southeastern Hydraulic Engineers' Meeting in Memphis, Tennessee, in November 1993. In addition to the Bridge Inspector, the U.S. Geological Survey (USGS) and another company demonstrated their bridge inspection system. The USGS demonstrated a boat-deployed system, while the Bridge Inspector and the other inspection system were deployed from the bridge deck. The bridge used for the demonstration was on State Highway 50 over the St. Francis River near Madison, Arkansas. The bridge was a simple concrete structure approximately 15 m (50 ft) above the water. Due to the large number of attendees, the demonstration was limited to presentation and operation of the equipment and no data were collected.

In 1994, the Bridge Inspector was field tested on several bridges in Georgia. This work, carried out under the difficult conditions of a flood-ravaged State was a wonderful opportunity to learn the limitations of the current system and to observe the need for useful enhancements. A discussion of several suggested enhancements is included in chapter IV of the report.

Inspection operations were conducted on bridge 19-3 in Taylor County, Georgia. This bridge spans the Flint River near Oglethorpe. A streambed contour measurement survey of three piers was conducted (figures 13, 14, and 15). At the time that the inspection was conducted, the water flow had slowed significantly and the bridge engineer believed that any scour holes that may have been present had now been filled in with normal river bottom material. Measurements taken during the survey confirmed this belief No significant scour holes were found.

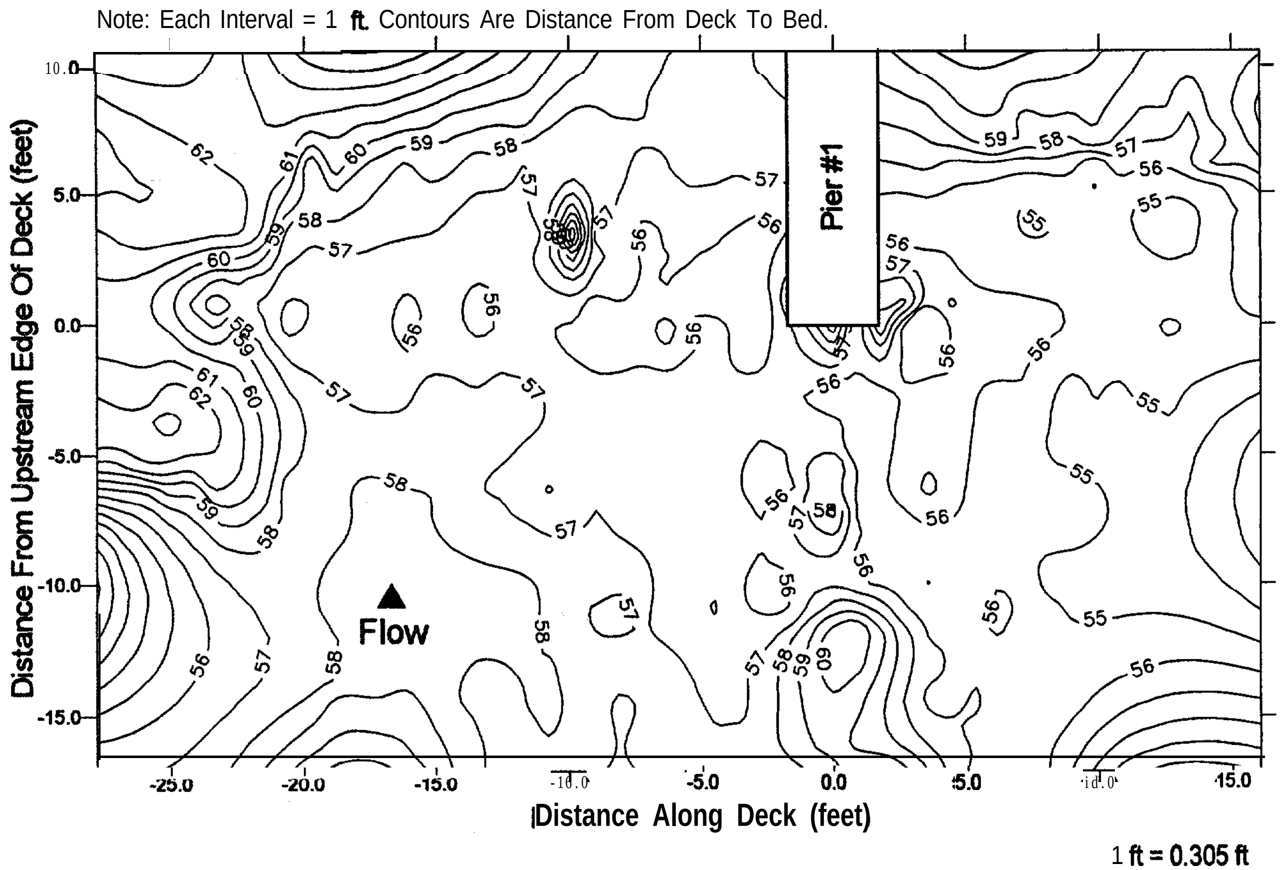


Figure 13. Flint River at Route 19 contour survey - pier no, 1.

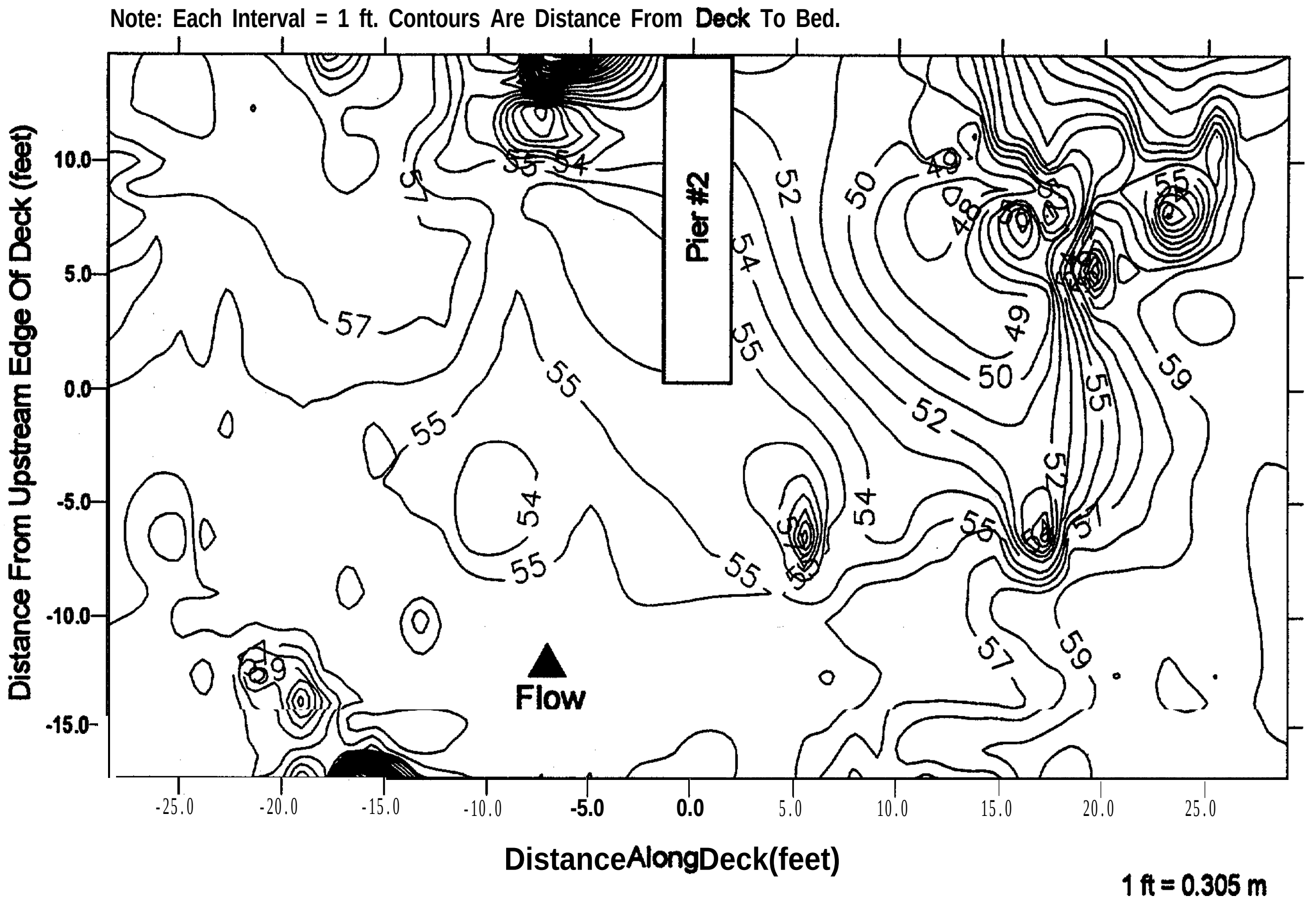


Figure 14. Flint River at Route 19 contour survey - pier no. 2.

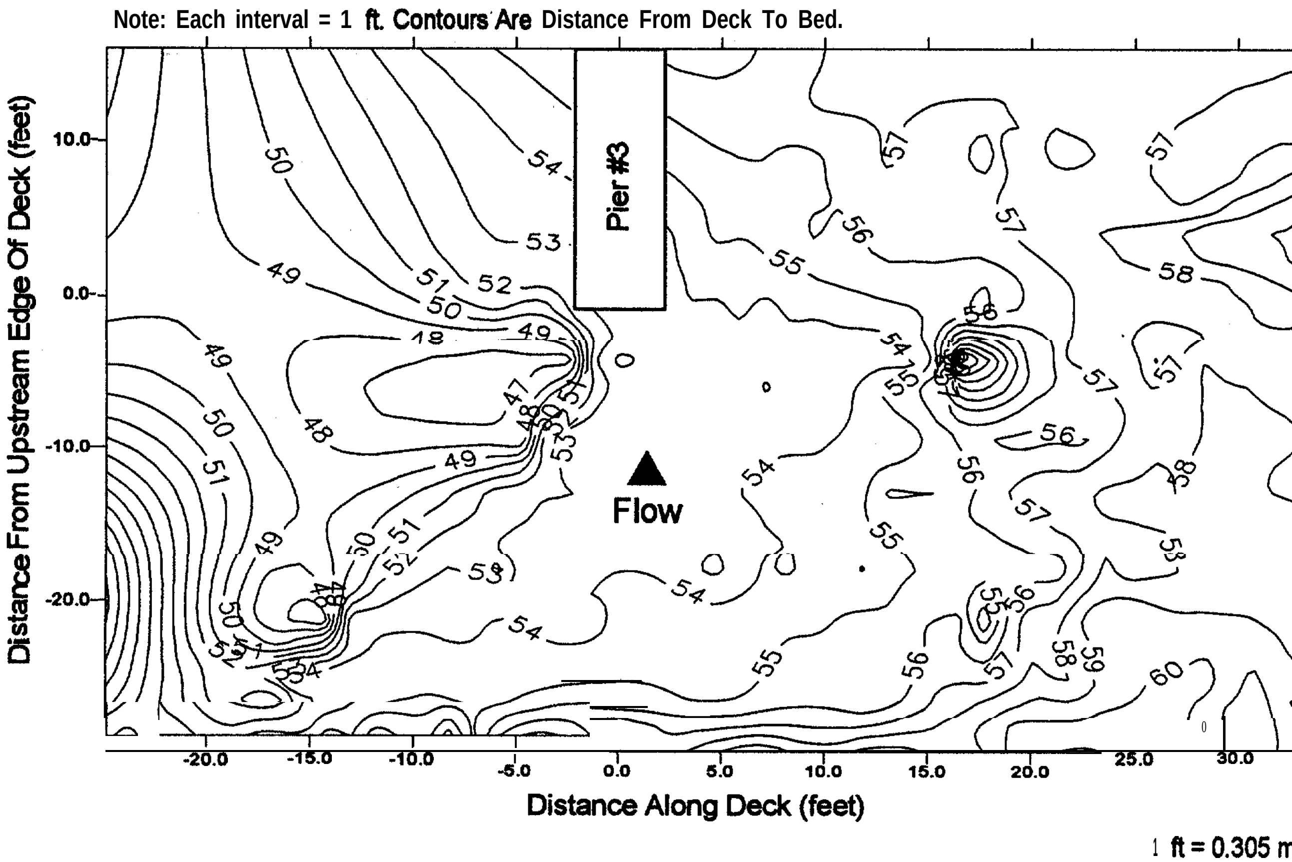


Figure 15. Flint River at Route 19 contour survey - pier no. 3.

The Bridge Inspector was then taken to bridge 128 over the Flint River near Flintside. The upstream side of the bridge was inspected by streambed contour measurement. The survey concluded that no significant scour holes were present.

The third bridge was at the Flint River on Highway 82 in Albany (figure 16). The bridge had been previously inspected by divers who measured the distance from the bridge deck to the streambed at 14.6 m (48 ft). A one-pass cross-section survey was conducted and the deck-to-streambed distance was found to have increased to 15.9 m (52 ft). This was a significant finding because the bridge piers are 16.8 m (55 ft) below the deck.

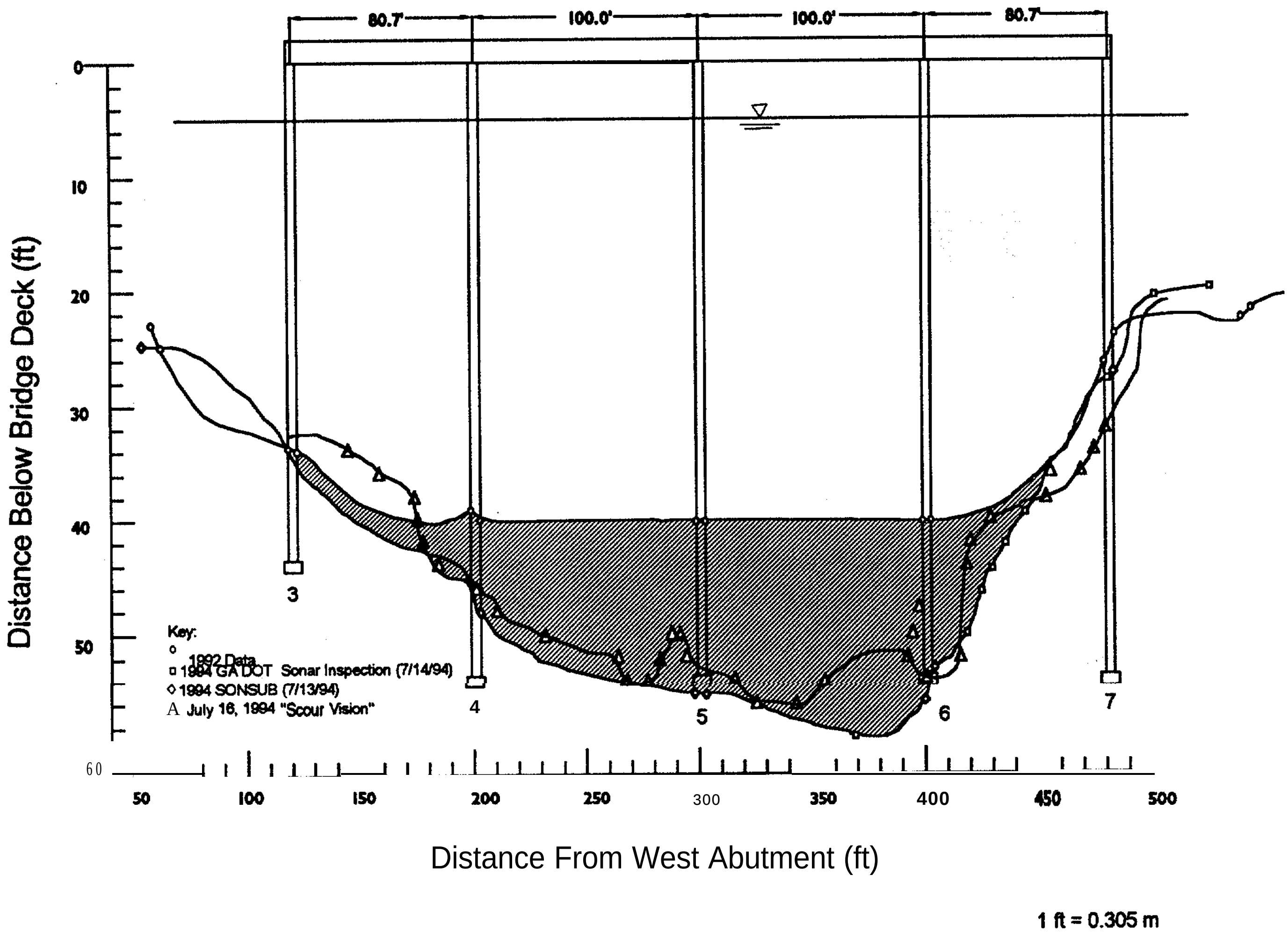


Figure 16. Oglethorpe Bridge over Flint River at Albany, GA - profile.

A one-pass cross-section survey was also conducted on the Ocmulgee River on Highway 280 near Abbeville (figure 17). The experience of actual field testing in the Georgia floods highlighted the biggest problem with a trailer-mounted inspection system. The operators often encountered situations where the road surface was the only dry land to be found. It was simply not safe to try to use the shoulder of the road to turn the trailer around. As is often the case with rural bridges, if the shoulder is flooded, the trailer may have to be towed many miles before a sufficient turnaround area is found.

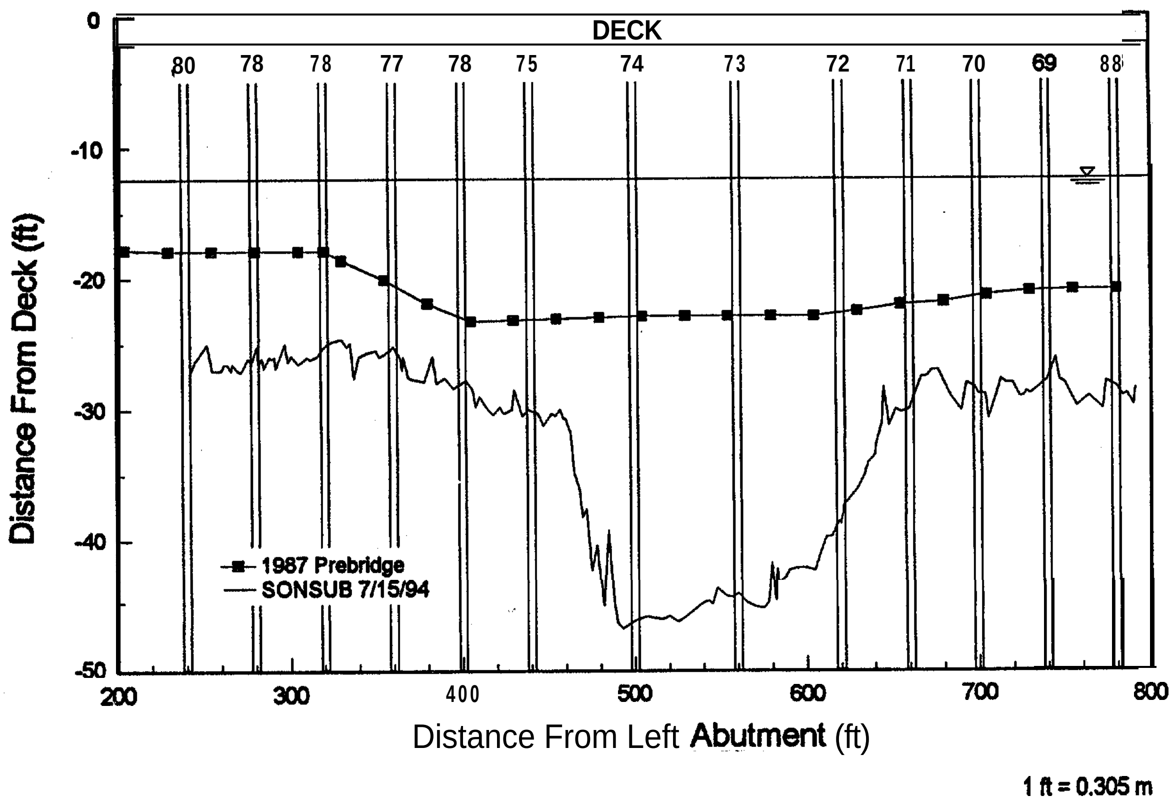


Figure 17. Ocuhnulgee River at Highway 280 looking downstream - profile.

A significant limitation of the system was found to be loss of signal (acoustic return to the sonar probe) in high-flow conditions. It was theorized that the loss of signal may have been caused by flow separation at the probe face due to vortex shedding around the 152-mm (6-in) diameter probe housing or by washing the return signal downstream before it could reach the probe. Although there was no means available to measure the flow rate, it was estimated at one point to be 4.5 m/s (15 ft/s). At this time, the water depth was measured to be approximately 15 m (50 ft). An acoustical pulse traveling at 1500 m/s (5000 ft/s) would take 0.02 s to travel from the surface to the streambed and back to the surface. During this time, the water mass would move about 0.1 m (4 in). This distance is not enough to account for the loss of signal, so washing away is not believed to be the cause of the loss of signal. Further work needs to be conducted to confirm that flow separation at the end of the probe is the cause. This work would also lead to the development of a probe configuration that would be less susceptible to this limitation.

IV. SUGGESTED ENHANCEMENTS

Field testing with the Bridge Inspector provided many insights into the wide range of capability-increasing enhancements that could be provided. Many of these enhancements involve wider array of sensors, but some are of a more basic nature. The enhancements that will be presented in this report include truck mounting, submersible head, velocity measurement, arm extensions, offset attachments, and signal processing.

TRUCK MOUNTING

A truck-mounted manipulator would have significantly more versatility than the trailer-mounted system. The amount of time and space needed to turn a truck around is significantly less than that needed to turn around a truck-and-trailer combination. One of the problems encountered during the flooding in Georgia is that a two-lane road surface is often the only safe place to make a U-turn. The truck-and-trailer combination was often not able to negotiate this maneuver and was, therefore, forced to make a significant detour to get in position to inspect the other side of the bridge. Although the trailer design has a lower initial cost, the operating difficulty is significantly higher than with a truck-mounted design. The trailer was acceptable for the initial prototype, but truck mounting is certainly desirable for future work. An alternative system might be to mount the manipulator arm on a heavy steel frame that could then be mounted on a truck as needed. In this design, the truck would need to be equipped with the same kind of load-bearing stabilizer wheel that is used on other truck-mounted bridge inspection equipment. It is also possible that the stabilizer wheel could be mounted on the manipulator frame so that the truck would act only as a counterweight and a propulsion system.

There are several advantages to mounting the system on a truck. One is that a short-wheelbase truck can be more easily maneuvered on a narrow road. A more significant feature is that the manipulator arm can be deployed from either side of the truck. Operations from either side of the truck might make it unnecessary to turn the truck around to inspect the other side of the bridge. An added benefit is that when the system is being used on an active roadway, the truck cab can be pointed against the flow of traffic to provide a measure of crash protection for the operating crew.

The summation of moments diagram (figure 18) indicates that a truck with a dead weight of at least 1565 kg (3450 lb) and a carrying capacity of 3697 kg (8150 lb) could be used in place of the trailer. The truck would need to be fitted with an-outrigger wheel on the side toward the arm (on both sides if operations to both sides are to be permitted). It is obvious that the size of the truck could be reduced if the pitch of the arm was limited in movement out from the side of the bridge.

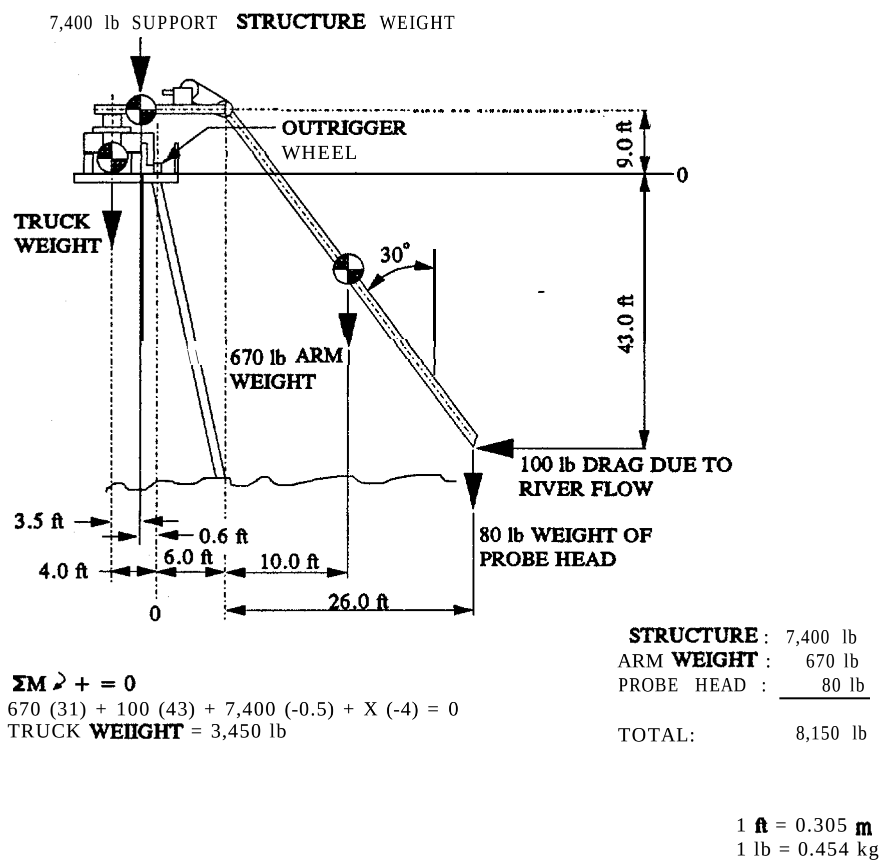


Figure 18. Summation of moments diagram for truck-mounted system.

ELEVATION CHANGE MEASUREMENT

Changes in elevation of the bridge deck used in the one-pass cross-section mode are calculated from data received from an inclinometer mounted on the trailer frame. This system requires that the sensor be calibrated with the trailer in a level position. This requirement dictates that the stabilizing wheels be adjusted to hold the trailer (and thus the inclinometer) parallel to the bridge deck. If the manipulator arm is mounted on a truck, it will be necessary to mount the inclinometer on a secondary frame or small-wheeled trailer. This arrangement will complicate the U-turn ability of the truck and, therefore, the design should incorporate features to make this maneuver as simple as possible.

SUBMERSIBLE HEAD

Another limiting factor encountered during field testing in Georgia was that the water was sometimes too high (too close to the bridge deck) to allow the system to be used. The Bridge Inspector's arm is 18 m (60 ft) long and fabricated in seven sections. With the overlap necessary to make the arm telescope, the arm can only be shortened to 4.5 m (15 ft) in the fully collapsed position. The arm pivot is approximately 2.7 m (9 ft) above the bridge deck. Therefore, when the arm is deployed straight down, the lower end of the arm is 1.8 m (6 ft) below the bridge deck. The sonar probe adds about 0.3 m (1 ft) to this length, making the minimum distance of the bridge deck to the water surface 2.1 m (7 ft). Using the Bridge Inspector on a bridge where the water is higher means that the probe and part of the boom must be submerged in the water. In the case of the prototype, the linear actuators and their power supply were not designed for submerged operations and thus could not be used in this application.

There are several alternatives that could be used. One alternative is to have the ability to raise the boom pivot so that the minimum distance is reduced. This could be easily accomplished with a telescoping pivot post or similar mechanism. Another alternative is to use a submersible linear actuator and remove the power supply from the end of the arm. Still another alternative is to use a different type of probe that could be completely submerged and would not require a continuous vertical attitude.

Any work done in this area should recognize the extra force on the manipulator arm due to the drag of the water flow around the arm. In a stream flowing at 6.1 m/s (20 ft/s), this drag equates to about 5.9 kg/m (4 lb/A) of submerged boom. If the arm was submerged approximately 2.1 m (7 ft) as described above, there would be no problem with tipping the system. However, if the arm was extended so that a large overturning moment was introduced by the angle (pitch out) of the arm plus a large drag due to water flow, then a tipping hazard would exist. An additional hazard of submerging the arm is the possibility of snagging a floating tree or other debris that could turn the truck over or possibly pull it off the bridge.

VELOCITY MEASUREMENT

The ability to measure the velocity of the water flow during the flood-stage conditions encountered in Georgia would have been very useful to the bridge engineers. A relatively low-cost sensor could easily be added to the probe package for velocity measurements. Since this is not a continuously monitored function, the sensor could be connected to an external display. This ability would be facilitated by having spare wires in the sensor umbilical cable that could be temporarily connected to the display device.

ARM EXTENSIONS AND OFFSET ATTACHMENTS

Another low-cost feature that would provide additional capability is arm extensions. These extensions would be used as needed to extend the physical limits of the arm so that the probe and other devices could go into hard-to-reach places. Several different extensions are envisioned (figure 19). A simple range extender could be a relatively slender fiberglass pole mounted in place of the probe package. This pole could be used to place a simple probe into the water as much as 6.1 m (20 ft) below the end of the arm. A shorter and more robust arm could be mounted at a right angle to the end of the arm so that a probe or camera could be

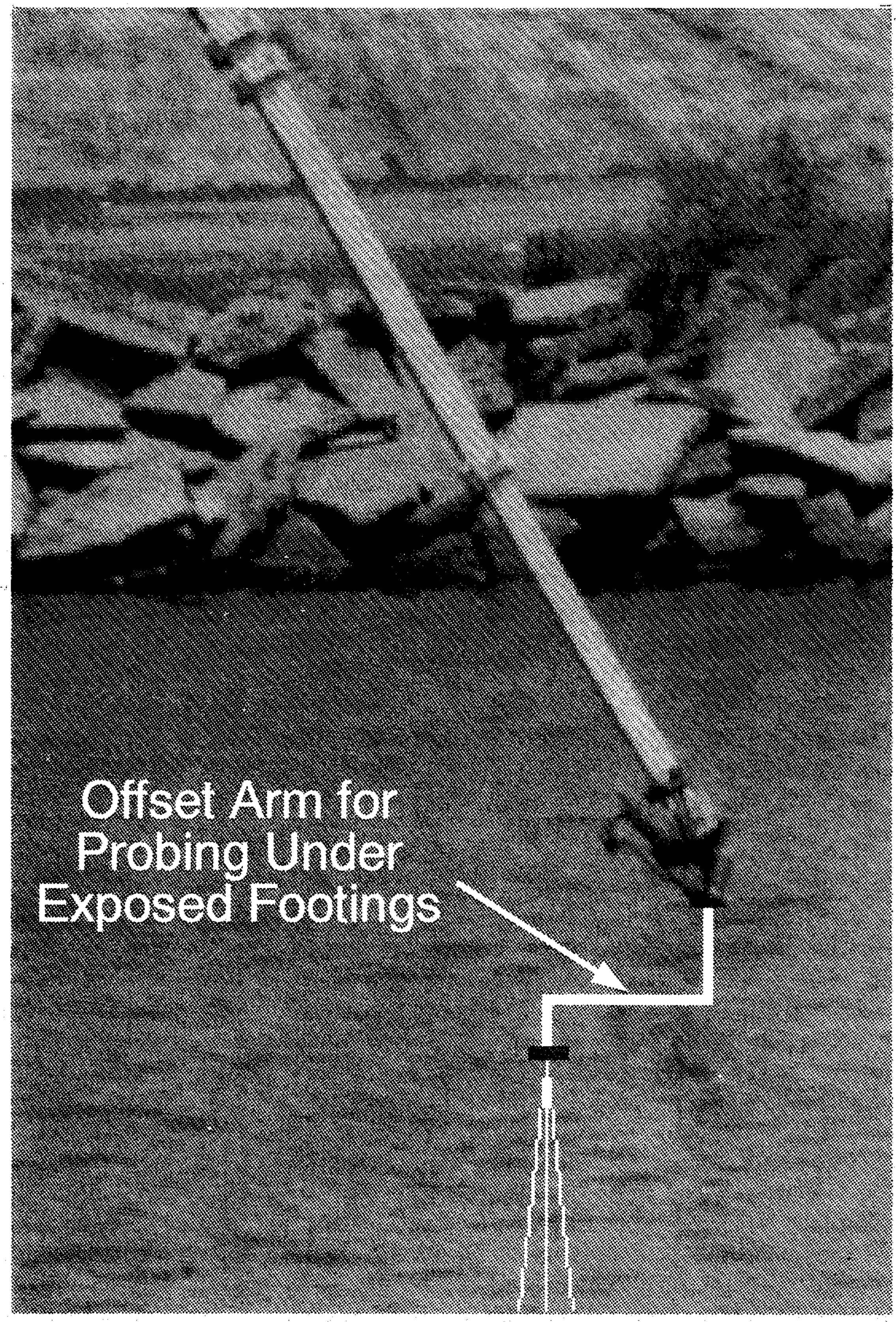


Figure 19. Suggested offset extension for Bridge Inspector arm.

maneuvered around and behind a pier. An inventory of extensions could be designed and kept with the Bridge Inspector for use in special situations.

OTHER DATA ACQUISITION INTERFACES

Field experience during the Georgia floods demonstrated the need for the ability to quickly switch between a variety of data acquisition devices. A quick disconnect (QD)-type attachment on the lower, end of the arm would provide the ability to deploy a wide range of sensor and other devices (figure 20). Different types of sonar probes could be available to fit the QD attachment. An underwater-rated plug and socket at the end of the arm would facilitate changing from one sensor to another. In some cases, a simple fish-tinder-type sensor would be preferable to the relatively large sonar probe that is currently fitted.

The ability to attach a rope-and-pulley arrangement to the end of the arm would allow the deployment of a surface unit such as the USGS Bridge Board, which is a water ski with a sonar probe attached to the bottom.

SIGNAL PROCESSING:

False readings from the sonar altimeter proved to be a nagging problem during field operations. In its current configuration, the Bridge Inspector has no provision for identifying or discarding inaccurate sonar readings. False readings from the sonar altimeter are thought to be generated from several different actions. These include loss of contact with the water surface due to wave action or operational limits on movement of the manipulator arm, sensing floating objects in the water, or vortex shedding at the probe face.

The current system hardware incorporates a resistive-capacitive (RC) low-pass filter to reduce high-frequency noise. This system was installed to eliminate naturally occurring high-frequency noise. However, this system also results in a slowed response to changes from the sonar probe. A better method would be to eliminate, false readings or at least indicate false readings so the operator can do something about them. One method that might be used would be to incorporate a digital filter into the system software. The digital filter would be used to reject more of the false readings and to identify false readings. A digital sorting algorithm would have to be developed that would match the characteristics of the rest of the system. The basis for the algorithm could be to observe the most common recurring numbers and to use only the recurring numbers as the logged reading. False readings might be eliminated by studying current signal returns and comparing them to the previous returns. Obviously, this type of software would require a good deal of development and testing before it could give reliable data.

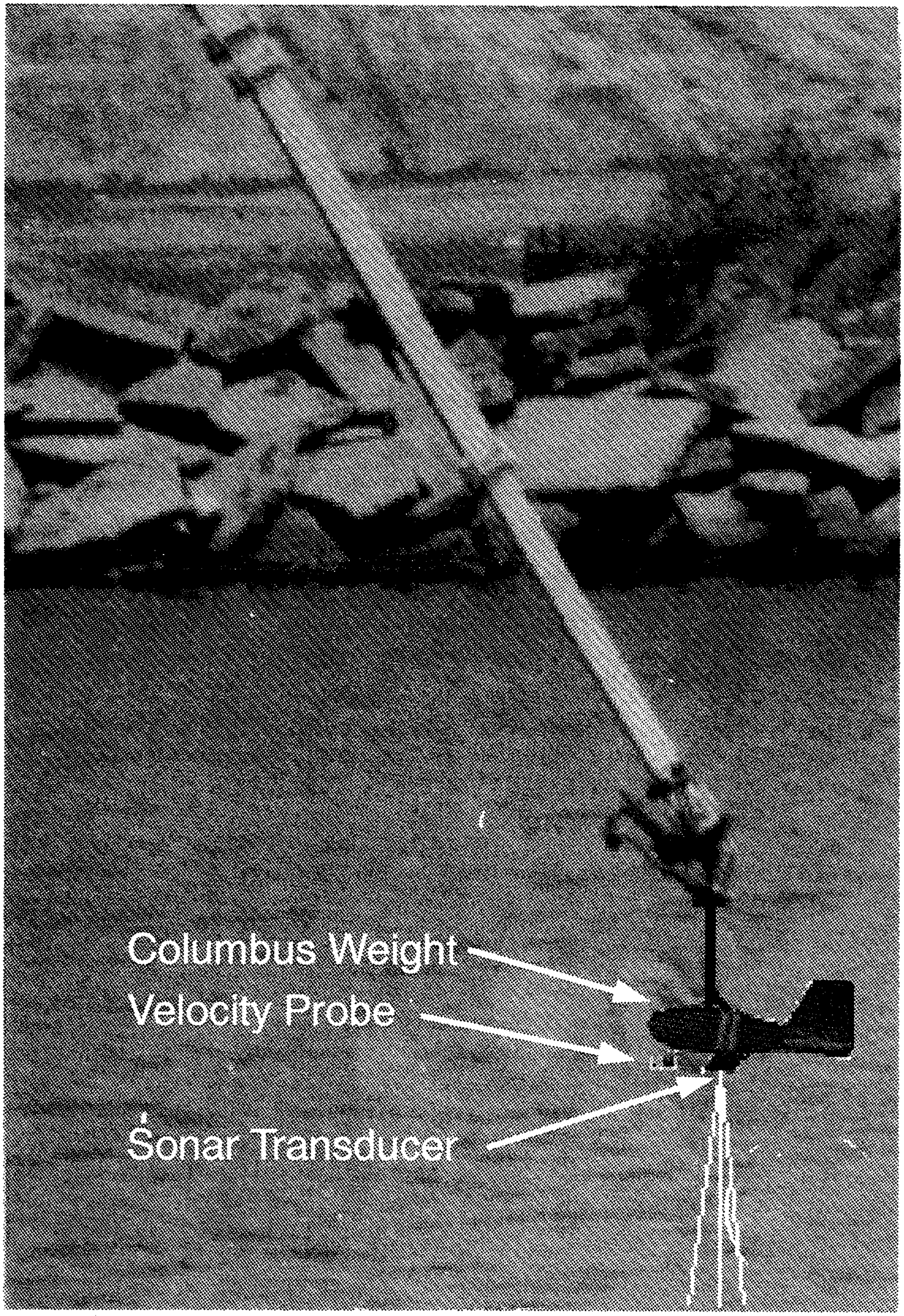


Figure 20. Bridge Inspector arm with Columbus weight probe.

V. PROBE SYSTEM

The advanced capability system designed as part of this project is called the Probe System. The Probe System (figure 21) consists of a truck-mounted deployment arm, a telescoping manipulator arm, a probe work platform with a spatially correspondent (SC) manipulator, and a control van. The deployment arm (shoulder) is mounted on a large, dedicated truck. The truck is designed so that it blocks only one lane of traffic during operation. Manual controls on the truck are used to operate the deployment arm functions. Manual operation with local control is preferred for deployment because of the hazards of traffic and overhead bridge structure. The truck has an onboard diesel hydraulic power unit to power the deployment arm functions. The telescoping manipulator (elbow) is specially designed to operate from the deployment arm. The telescoping manipulator has 3 degrees of freedom as well as position sensors for computer interface. The telescoping manipulator is powered from the deployment arm power unit. The probe work platform is mounted on a three-function wrist connection. The probe work platform is fitted with horizontal distance sensors, a scanning sonar system, a downward-looking sonar, a video camera, lights, and an SC manipulator. The control van houses two control consoles (figure 22) that operate all the functions of the elbow, wrist, probe work platform, and SC manipulator. Video monitors include the sonar displays, computer graphics display (figure 23), and video camera picture. An onboard diesel generator mounted outside the control van provides electrical power for the system.

FUNCTIONS

A spatially correspondent force-feedback manipulator arm mounted on the probe work platform will be used to deploy sensors and conduct other intervention tasks. A video camera mounted on the arm will give the operator visual reference to the structure when the probe head is above the water level. Although we are assuming "black water" conditions during most operations, the camera can also be used in more optimum water conditions for a visual inspection of the structure. The camera will have automatic iris control and remote focusing capability. An underwater light with an adjustable intensity control will be used in conjunction with the video camera. A scanning sonar will be utilized for visual reference purposes when the probe head is underwater. The scanning sonar is used to locate the structure and guide the operator in moving the probe work platform to the structure in poor or nonexistent visibility conditions. A downward-looking scanning sonar is used to help the operator keep the work platform off the streambed over the wide range of motions available at the probe. This not only gives distance to the streambed, but also allows the operator to "see" obstructions and bottom contours at the sides of the probe work platform. This system will minimize collisions of the probe with unexpected obstructions and will help prevent entanglement of the probe.

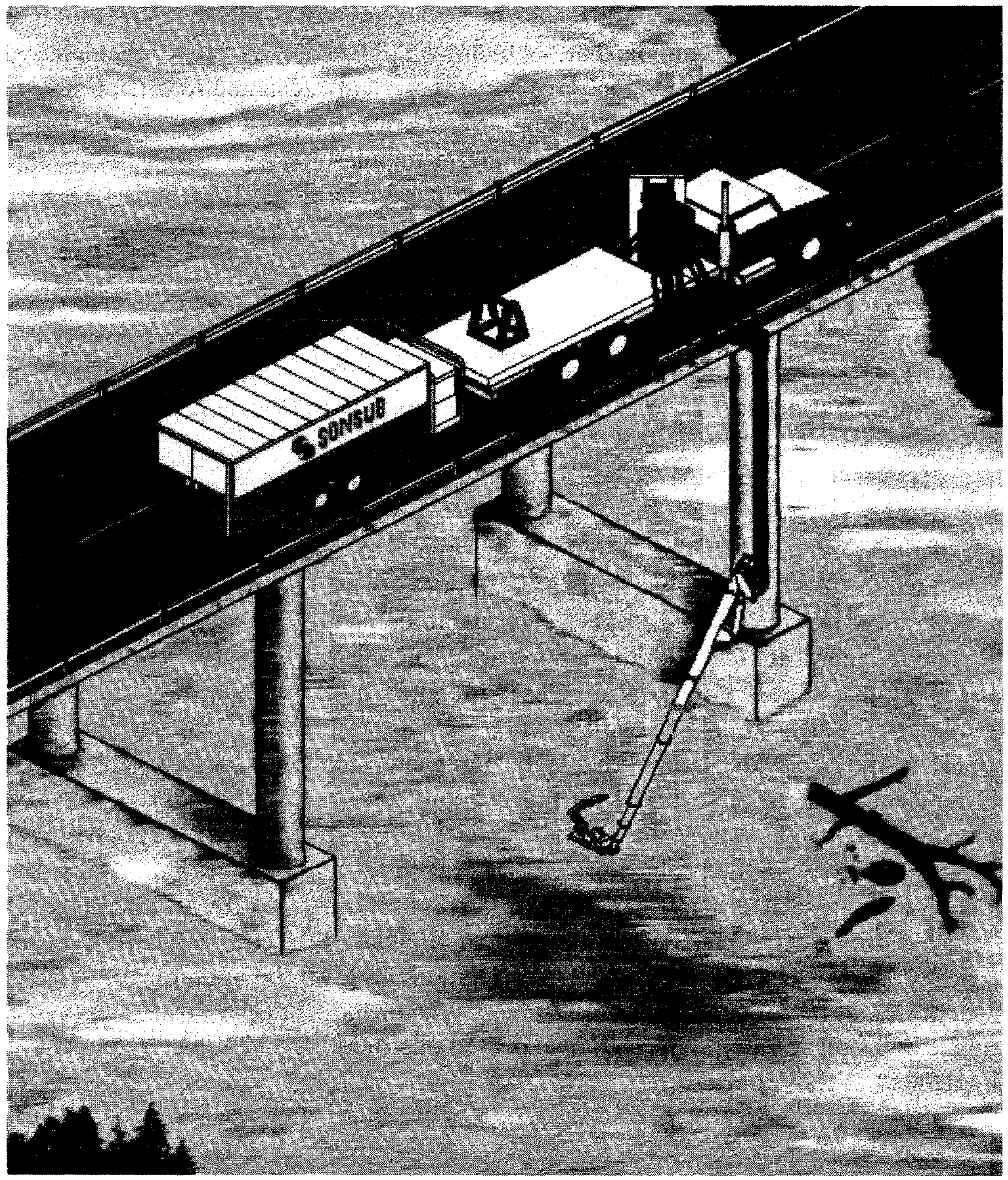


Figure 2 1. Probe system.

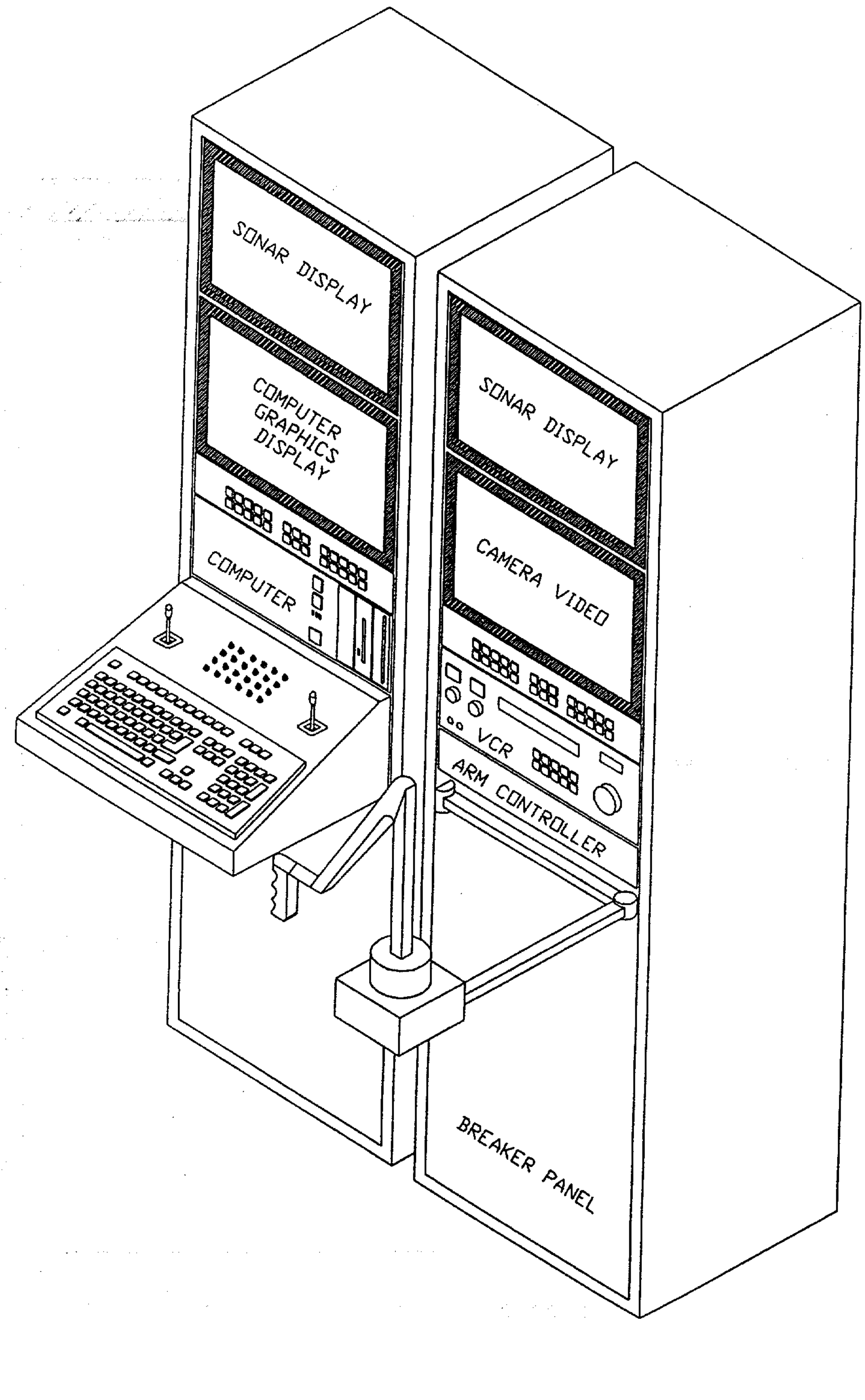


Figure 22. Probe system control consoles.

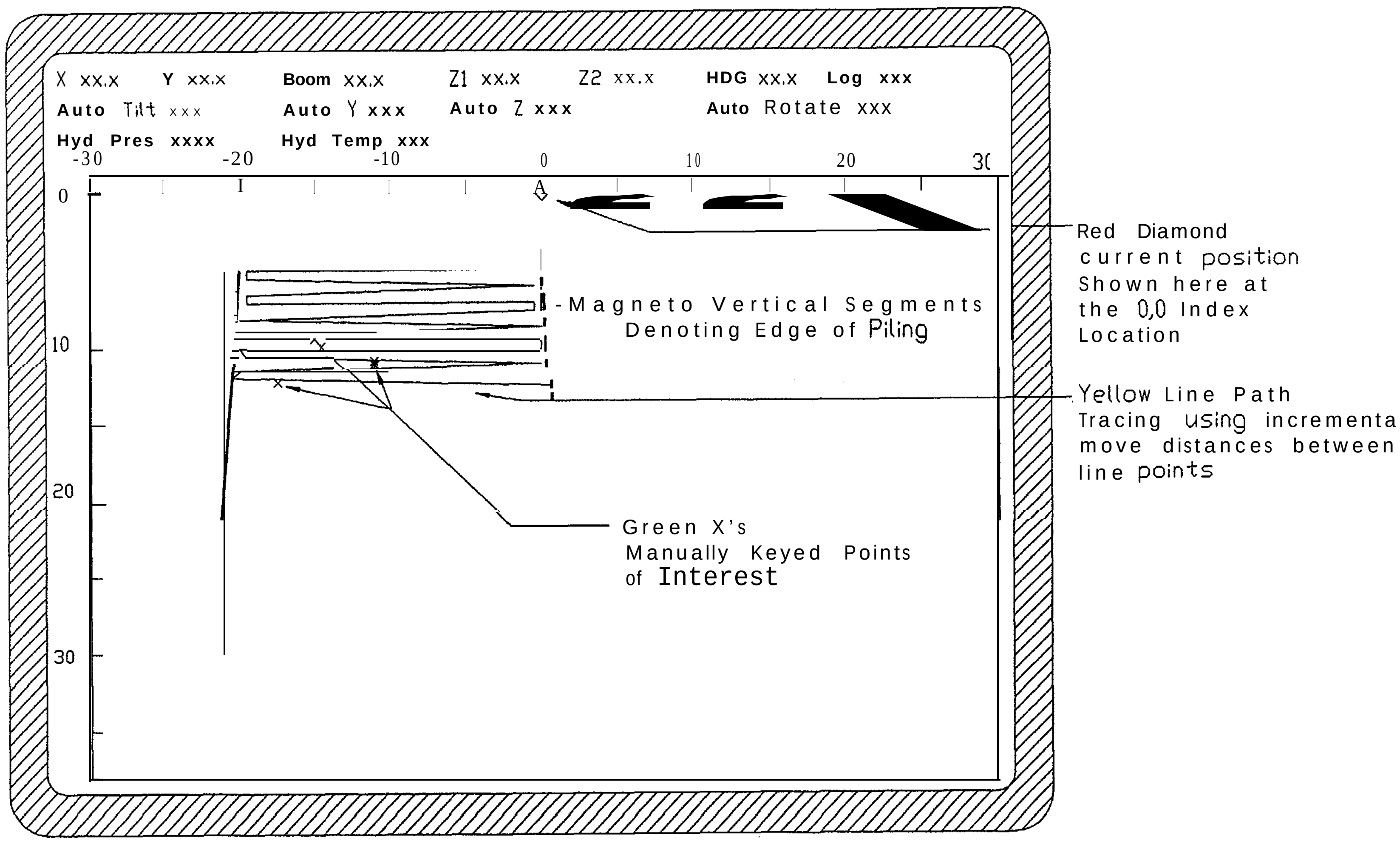


Figure 23. Graphics display

Manual hydraulic valves will be used to operate the deployment arm. Joystick and automatic **functions** (figure 24) will control the telescoping arm and the probe work platform. Joystick controls, auto-control functions, and the necessary sensors for auto-control functions of the elbow and probe work platform are described below.

TELESCOPE ARM AND PROBE PLATFORM POSITIONING

Auto-Tilt: Uses mercury level switches to automatically maintain probe work platform in a horizontal position.

Auto-Rotate: Uses horizontal distance sensors to keep probe platform rotated flush with the structure. If one sensor reading is greater than a 0.61-m **(2-ft)** difference from the other sensor, the auto-rotate function will halt. An alarm will be sounded and the x-axis (elbow rotate) function will not allow further movement in the direction of the greater reading sensor. This is necessary when working close to the edge of pilings, etc. Manual joystick control of the probe rotation and elbow rotation will be permitted only after the alarm has been acknowledged. The horizontal distance sensors are also used to determine edges in the graphics log mode.

Auto-Y: Controls elbow tilt function. Uses elbow extension and elbow tilt sensors to maintain Y set position of the probe platform. When auto-Y is enabled, the elbow tilt joystick will be used to change the Y set position.

Auto-Z: Uses horizontal distance sensors to maintain user-set distance **from** the structure by controlling the elbow extension. In manual-Z mode, the joystick will control elbow extension. In auto-i! mode, the same joystick will control the user-set distance.

Elbow Rotate: Joystick control of the elbow rotate function. This controls the X position of the boom end. When auto-rotate is enabled on the probe end and the horizontal distance sensor readings are greater than a 0.61-m **(2-ft) difference,** X movement in the direction of the greater reading sensor will be inhibited unless an override acknowledge key is pressed.

Elbow Tilt: Joystick control of the elbow tilt function. This controls the Y position of the probe platform.

Elbow Extend: Joystick control of the elbow extend function when the auto-i! function is off. When the auto-i! function is on, this joystick will control the Z set distance from the structure. The elbow extend function will be disabled when the elbow has reached maximum extension and an alarm condition has occurred, alerting the operator that maximum extension has been reached.

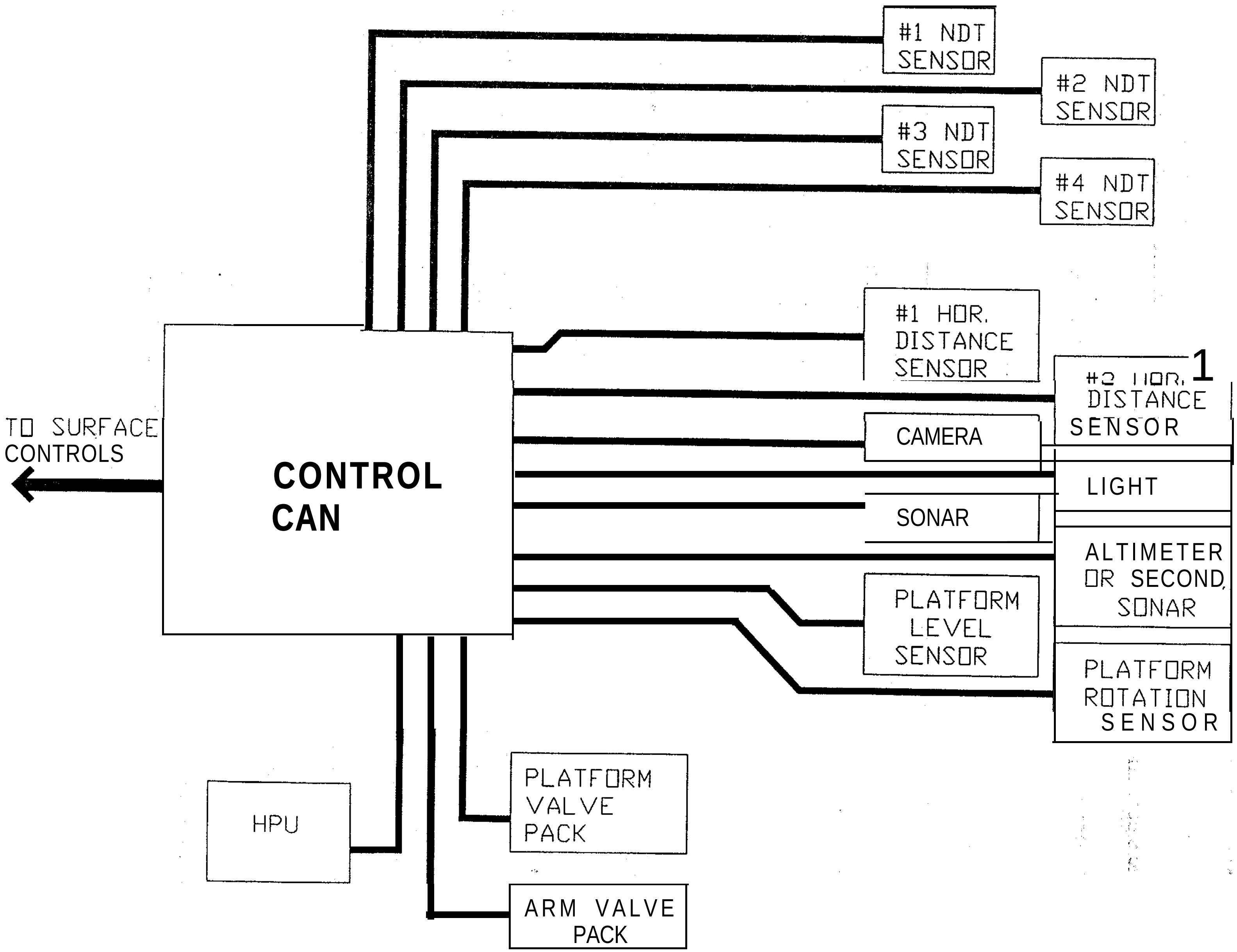


Figure 24. Probe control system block diagram.

X Coordinate, Y Coordinate: Uses elbow rotate, elbow tilt, and elbow extend sensors to calculate the X, Y coordinates of the probe end from the indexing reference mark.

Z Distance: Uses horizontal distance sensors to give distance from probe platform to the structure.

COMPUTER DISPLAY

Sensor Readouts: The display position has different functions depending upon mode selection. Prior to indexing, the X and Y display will show the joint angles. After indexing, the X and Y display will show the distance in feet from the index location. The display can be toggled between the two readouts. Other readouts will remain the same in both modes.

Prior to Indexing	After Indexing
X: Displays elbow rotate angle	Displays X coordinate in 0.0305-m (0.1-ft) increments from index.
Y: Displays elbow tilt angle	Displays Y coordinate in 0.0305-m (0.14) increments from index.
Boom:	Displays the elbow extension in 0.0305-m (0. I-R) increments.
Z1:	'Displays the left horizontal distance sensor reading in 0.0305-m (0.1-ft) increments.
Log:	Displays the status of the log mode ON-OFF-MAN.

On: Indicates that logging is automatically performed by the computer.

Note: Computer tracking display trace will be updated whenever the X or Y coordinate reading changes by 0.3 m (1 ft) and the corner markers will be displayed whenever one horizontal distance indicator reads greater than a 0.61-m (2-ft) distance from the other horizontal distance indicator. Corner indicator X and Y coordinates will be logged to file as will manually keyed points of interest.

Off: Indicates that no logging is performed. Trace mode on display is disabled as is the corner markers insertion. No data points are written to file.

Man: Corner markers and points of interest must be inserted manually (by key press) and will be logged to file.

Auto-Tilt: Indicates that automatic leveling of probe platform is on or off.

Auto-Z: Indicates that probe auto-rotate function is on or off. Additional sensor readouts may be displayed if required (such as hydraulic pressure and temperature, alarms may be set if readings are outside of preset ranges, etc.).

Graphics: The graphics screen will be used **after** indexing at the prominent point.

Red Diamond: A red diamond will be displayed on the graphics screen to indicate the current position of the probe end. The current position will be shown in feet relative to the index mark. The index mark will be set'at zero-X and zero-Y coordinates.

Yellow Trace: When the auto-log mode is on, a yellow line will be drawn on the graphics screen to indicate the path of the probe end. This will enable the operator to visually reference which portions of the structure have been examined. Tracing will be disabled if the horizontal distance sensors are more than 1.5 m **(5-ft) from** the structure or **if the** auto-log mode is off. In the manual log mode, single line points may be keyed in for display by key press.

Green X's: A green X symbol will be generated by a manual key press and will be displayed at the coordinates when the key press occurs. These symbols are used to denote points of interest on the structure (i.e., possible faults or anomalies). These symbols can be generated when the log mode is on or manual.

Magenta Segments: A magenta-colored vertical line segment will be used to denote the edges of the structure for visual reference. When the log mode is on, these segments will be generated if one sensor reading is greater than 0.61 m (2 **ft) from** the other horizontal sensor. When the log mode is manual, these can be generated by a manual key press. When the log mode is off, no symbols are generated.

OPERATION

Deployment arm is manually operated (with visual control **from** bridge roadway):

1. Elevate shoulder.
2. Swing shoulder over railing.
3. Swing extension joint out.
4. Lower shoulder, keeping extension joint vertical.
5. When shoulder is horizontal, center extension joint to 0 degrees vertical.
6. Extend extension joint to water level.

Telescoping arm and wrist is remotely operated from the control console:

1. With extension joint at water level and 0 degrees vertical, turn on lower control and power package.
2. With hydraulic power unit (HPU) on, wrist joint will become level.
3. Locate structure with scanning sonar.

4. Turn auto-i! switch on (this will keep wrist at same altitude as elbow in-out is used).
5. Move elbow close to one corner of structure using sonar **and/or** video.
6. Establish index mark. Press zero reference key on computer. This establishes index mark as **0, 0** on coordinate system for repeatability.
7. Move elbow to face of structure. Flex wrist until square with structure (the two horizontal distance sensors must read within 0.61 m (2 **ft)** of each other).
8. Turn auto-flex switch on (this will keep probe head square with structure and will also be used to "map out" boundaries of structure. When one sensor reads more than 0.61-m **(2-ft)** difference **from** the other sensor, indicator will light and **auto-**flex will hold at last position. This can also be used to prevent further elbow movement in the direction of the greater reading sensor unless an override switch is **thrown).**
9. Start inspection and mapping of structure. Coordinate boundaries may be **defined first** or as inspection is taking place.

COMPUTER

The computer will be used as the surface **I/O** device and for auto-function control (figure 25). It will also be used for logging coordinate systems and for anomalies record keeping. The computer graphics screen **will be** used for **status** readouts and for displaying the mapping coordinate system.

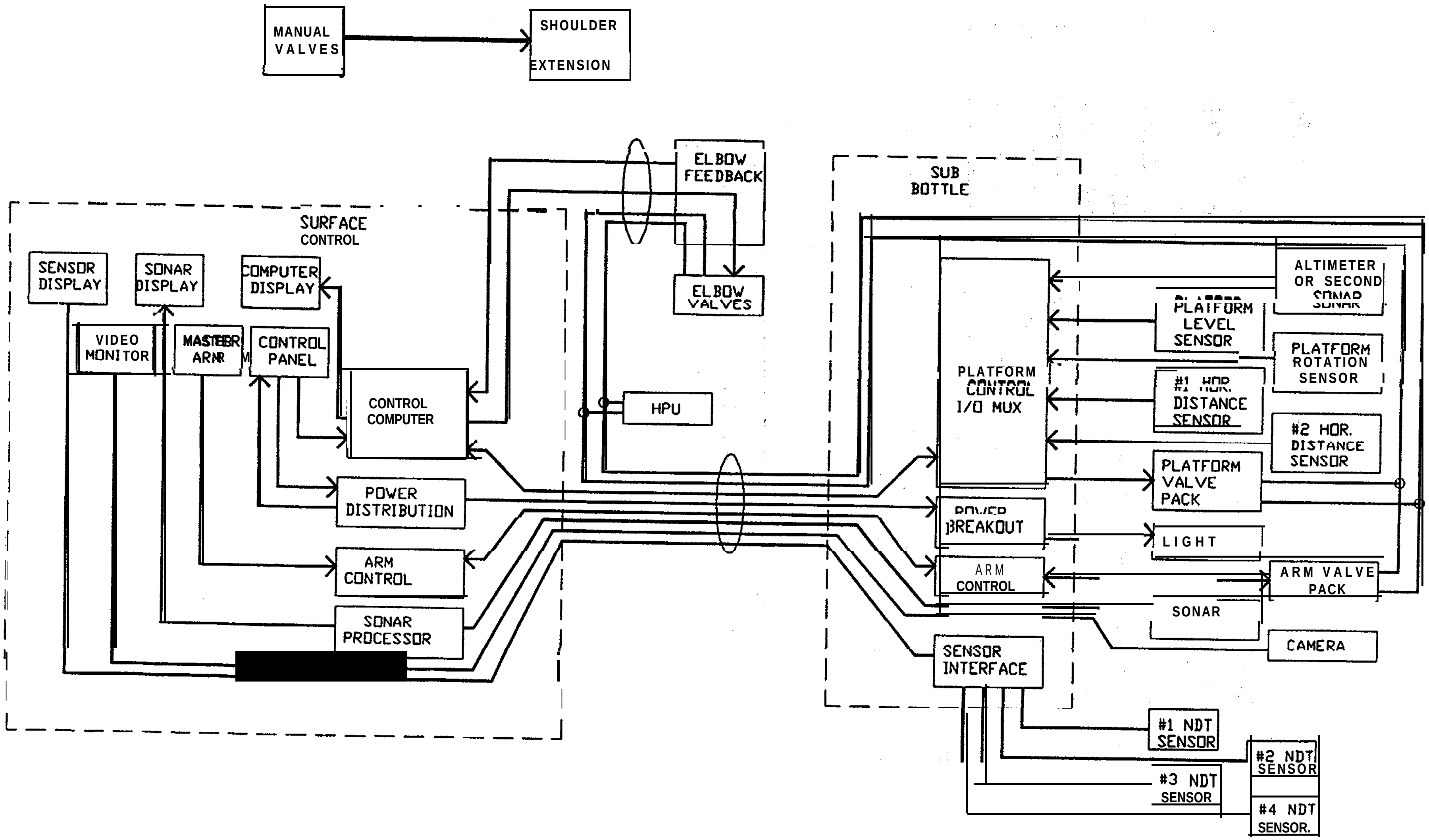


Figure 25. Work platform electrical block diagram.

APPENDIX A. PARTS LIST

CODES: T = TRAILER, A = ARM. H = HYDRAULICS. C = CONTROLS, F = FRAME. 0 = OPERATIONAL, DR = DRAWINGS

QTY	DESCRIPTION	CODE	VENDOR OR MFG & PART #	VENDOR	PHONE
1	Turn-Down Plate Washer	A	BH-N227DO50-P1	Electro-Mechanical Srvcs	
4	Stabilizer Leg Screws 1 & 1/2 -8	A	BH-N227DO48-P1	Electro-Mechanical Srvcs	
1	Lot Trailer Work Modifications	T	NA	Merriworth Trading	(713) 466-7998
4	Stabilizer Leg Screws	T	BH-N227DO48-P1	Dynamic Coating	
4	1- 1/2 x 8 Hex Nuts	T	NA	Dynamic Coating	
1	Extra Work to Trailer	T	NA	Merriworth Trading	(713) 466-7998
1	Permanent Magnet dc Motor	T	5C-24156	M.E.T.	(507) 625-6117
1	Contaq Sensor Modification	C	RMA 1471	Contaq	(802) 453-3332
1	Tran... Trailer Frame	T	NA	Texas Hot Shot	
4	Leg Stabilizer Stop screws	T	AH-N227D022-P1	Merriworth Trading	(713) 466-7998
4	Stabilizer Leg Modifications	T	AH-N227C022-P1	Merriworth Trading	(713) 466-7998
1	Nema of 19" Rack & Accessories	C	213088	Hennessy Products	(717) 264-7146
2	Flanged Bearings	T	5T/1	Bearings Inc.	
10	7/16-14 UNC-2A x 1 1/4 " HEX	T	NA	Rainbow	
100	7/16" Lock Washers	T	NA	Rainbow	
100	7/16" Flat Washers	T	NA	Rainbow	
1	24 V dc HPU	H	SRV3-25-93	Hapeco	(713) 466-1986
8	Marine Batteries	T	DC27115	GNB, INC.	(713) 462-2375
1	Exchange Hyd Brake Coupler	T	NA	HuskieTrailer	(713) 449-6246
1	Set Blue-Line Prints	T	NA	Blue Prints Plus	(713) 468-7025
8	3/8-16 UNC-24 x 3&3/4	T	NA	The Nut Place	(713) 462-3147
1	Sheave Shaft	T	AH-N227D052-P1	Electro-Mechanical	(713) 896-4373
1	Sheave Retainer	T	AH-N227D037-P1	Electro-Mechanical	(713) 896-4373
1	1/2 - 13UNC-2A x 4 Hex Bolts	T	NA	The Nut Place	(713) 462-3147
1	1/2 - 13UNC-2B Hex Nut Nyloc	T	NA	The Nut Place	(713) 462-3147
1	3/8 - 16UNC-2B Hex Nut Nyloc	T	NA	The Nut Place	(713) 462-3147
1	Set Latches & Retainers	T	Various N222D...	Merriworth Trading	(713) 466-7998
4	750-16 14-Ply Tire Upgrade	C	750-16	Discount Tires	(713) 558-0591
1	Solar Power Supply ± 12 v dc	C	62F1217	Newark	(713) 894-9334
1	Multi-Outlet Surge Suppressor	C	89F5056	Newark	(713) 894-9334
20	Dorn Tube Size 1 Straight	C	M19622/1-001	Dorn	(617) 662-9300
6	Dorn Tube Size 1 Y	C	M19622/4-01	Dorn	(617) 662-9300
18	Packing Glands Size 1B1	C	M19622/16-0005	Dorn	(617) 662-9300
8	Packing Glands Size 1C	C	M19622/16-0006	Dorn	(617) 662-9300
1	P.O. Revision #20978	T	Change gear box	T.L. Walker	(713) 460-4000
1	Painting Supplies	C	NA	That Hardware Place	
1	Battery Box	T	NA	Rice Metal	(713) 462-1978
2	Enlarge Drill Holes	T	NA	Merriworth Trading	(713) 466-7998
12	XOR GATE	C	74LS86	Ace Electronics	
20	1 UHF Cap	C	NA	Ace Electronics	
1000	0.1 UHF Cap	C	NA	Ace Electronics	
25	Feet Battery Cable- Red	C	WHT-594	Hi-Lo Auto Supply	(713) 469-9506
24	Battery Terminals	C	EXI-B102B	Hi-Lo Auto Supply	(713) 469-9506
18	Hyd Hose Adaptor	H	6MB X 6 MJ	PowerTrac	(713) 864-4673
7	Hyd Hose Adaptor	H	4MB X 6 MJ	PowerTrac	(713) 864-4673
8	Hyd Hose Adaptor	H	8MB X 6 MJ	PowerTrac	(713) 864-4673
8	Hyd Hose Adaptor	H	6MJ X 6FJ X 90	PowerTrac	(713) 864-4673
1	Hyd Hose Adaptor	H	6MP X 6MJ	PowerTrac	(713) 864-4673
7	Hyd Hose Adaptor	H	4MB X 6FJ X 90	PowerTrac	(713) 864-4673
1	Hyd Hose Adaptor	H	6MJ X 6MJ X 6FJ X T	PowerTrac	(713) 864-4673
1	Hyd Hose Adaptor	H	6FP X 4 FP	PowerTrac	(713) 864-4673
1	Hyd Hose Adaptor	H	0-2000-psi GAUGE	PowerTrac	(713) 864-4673
6	1" x 10" x 142.4" Flat Bar	T	NA	Sterling Steel	(713) 690-0347
1	10" Channel X 5"	T	NA	Sterling Steel	(713) 690-0347
10	Gal. Hyd Oil	H	Chevron AW-150 Grade 32	Moffit Oil	(713) 376-1595
60'	12 Conductor 18 AWG Cable	C		Basic Wire and Cable	(713) 228-0400
200'	22 AWG Shielded Quad	C		Basic Wire and Cable	(713) 228-0400

CODES: T = TRAILER. A = ARM, H = HYDRAULICS, C = CONTROLS, F = FRAME. 0 = OPERATIONAL, DR = DRAWINGS					
QTY	DESCRIPTION	CODE	VENDOR OR MFG & PART #	VENDOR	PHONE
6	10-Degree Tilt Switches	C	CMBR90	Comus	(201) 667-6200
1	Paint Battery Box	T	NA	Car Shop	
4	FT. Chain - Spectrum 4 or Equal	A	Spectrum 4	Bishop	(713) 674-2266
2	5/16" Latching Clevis Hook	A	S3144-1225021	Bishop	(713) 674-2266
1	Linear Actuator Plate Mod	A	DH-N227D042-G1 REV A	Custom Heliarc	
1	Hydraulic "T" Fitting	H	6MJ X 6MJ X 6FJXBRANCH	PowerTrac	(713) 864-4673
6	Sheave Block	A	NA	Sonbeck	(713) 355-5003
1	50-Pin Shrouded Header	C	46F600	Newark	(713) 688-6954
4	Eject Levers for Above	C	46F626	Newark	(713) 688-6954
8	Quad 2 Input Nor Gate	C	SN74LS02N	Newark	(713) 688-6954
4	Power Op-Amp	C	ULN3751ZV	Newark	(713) 688-6954
4	Quad-Power Driver	C	UDN2944W	Newark	(713) 688-6954
16	Dual-Row Terminal Strips	C	90F8135	Newark	(713) 688-6954
8	14-Pin Dip Sockets	C	76F10-43	Newark	(713) 688-6954
10	&k-Mounted Computer System	C	RME 145 & RMK 110	Recortec	(800) 729-7654
1	Topside Interface PCBS	C	NA	Hedcorp	(713) 696.3451
5	Ball-Drive Actuator	A	85151	Motions Systems Gnp	(908) 222.1800
2	Universal Joint	A	McMaster-Carr 6452K7	McMaster-Carr	(708) 833-0300
1	UHMWP Rect. Bar 3/8" x 4" x 5"	A	McMaster-Carr 8702K78	McMaster-Carr	(708) 833-0300
2	2 1/2 Bronze Bushing	A	T.L. Walker	T.L. Walker	(713) 460-4000
1	Quick-Release Pins	A	98320A138	McMaster-Carr	(708) 833-0300
4	Extension Springs	A	E0750-125-2000	Associated Spring	(800) 442-5108
10	Asst. Nuts & Bolts	A		Rainbow	
1	Asst. Nuts & Bolts	A		Huffco	
1	Blue-Line Prints	DR	NA	Blue Prints Plus	(713) 468.7026
33	12,000 # Axle Kits	T	TS-6079	Husky Trailer	
1	#10 Dico Surge Couple	T		Husky Trailer	
1	Tandem Hyd Line Kit	T		Husky Trailer	
1	Light Kit	T		Husky Trailer	
1	License Plate Light & Bracket	T		Husky Trailer	
1	10 x 72 Fenders	T		Husky Trailer	
2	16" Wheels	T		Husky Trailer	
4	5000 # TW Jacks	T		Husky Trailer	
1	5000 # Casters	T		Husky Trailer	
1	2 5/16 x 1 Hitch Ball	T		Husky Trailer	
1	24" Unicycle	T		Memorial Schwinn	(713) 465-0096
1	Airless Tube	T		Memorial Schwinn	(713) 465-0096
1	Tube Installation	T		Memorial Schwinn	(713) 465-0096
1	Spelter Socket 1/2" dia	A	G-416-1039655	Intercoastal Supply	
1	17/8- 9 x 2 1/4 Bolt	A		Intercoastal Supply	
4	1 1/8 Washer	A		Intercoastal Supply	
1	Revision to P.O. 20938	F	Hogan Fluid Power	Hogan Fluid Power	(713) 741-0435
4	Top Wind Jack Handle	T	TWH	Trailer Wheel & Frame	(713) 578-8888
1	Linear Actuator Plate	T	DH-N227D042-G1	Merriworth Trading	(713) 466-7998
1	Spacer	T	BH-N227D046-P1	Merriworth Trading	(713) 466-7998
2	Pull-Bar Pins	T	BH-N227D044-P1	Merriworth Trading	(713) 466-7998
1	Trailer Pull Bar	T	CH-N227D043-G1	Merriworth Trading	(713) 466-7998
2	Odometer Mounting Lug	T	BH-N227D-080-P1	Custom Heliarc	
1	Odometer Pivot Arm	T	BH-N227D-081-G1	kustom Heliarc	
2	Encoder Anti-Rotation Bar	T	BH-N227D-082-P2	Custom Heliarc	
2	Encoder Anti-Rotation Arm	T	BH-N227D-083-P1	Custom Heliarc	
1	Encoder Coupling	T	BH-N227D-084-P1	Custom Heliarc	
8	Hardened Thrust Washers	T	CAT#18866	Bearing Inc.	
4	Tires	T	NA	Discount Tire	(713) 558.0591
4	Valve Stems	T	NA	Discount Tire	(713) 558-0591
4	Recycle Fees	T	NA	Discount Tire	(713) 558-0591
1	12 Terminal Barrier Strips	C	28F710	Newark	(713) 688-6954
1	8 Terminal Barrier Strips	C	28F706	Newark	(713) 688-6954
4	20 Terminal Barrier Strips	c	28F876	Newark	(713) 688-6954

CODES: T = TRAILER, A = ARM, H = HYDRAULICS, C = CONTROLS, F = FRAME, 0 = OPERATIONAL, DR = DRAWINGS					
QTY	DESCRIPTION	CODE	VENDOR OR MFG & PART #	VENDOR	PHONE
1	12 Terminal Marker Strips	C	29F814	Newark	(713) 688-6954
1	8 Terminal Marker Strips	C	29F810	Newark	(713) 688-6954
4	20 Terminal Marker Strips	C	29F822	Newark	(713) 688-6954
25	Terminal Jumpers	C	28F833	Newark	(713) 688-6954
1	Solid-State Relay	C	89F953	Newark	(713) 688-6954
3	Pots 5K	C	87F3561	Newark	(713) 688-6954
2	Panel-Mounted LEDs, 24V	C	87F928	Newark	(713) 688-6954
1	Inverter Power Supply 24-115	C	87F5329	Newark	(713) 688-6954
1	24-V Battery Chargers	C	7322K49	McMaster-Carr	833-0300
2	Battery Disconnect Switches	C	7110K38	McMaster-Carr	(708) 833-0300
2	Arm Thrust Washer	A	BH-N227D049-P1	Electro-Mechanical Srvcs	(713) 741-0435
1	2.5" Bore Cylinder, 22" S, 1.25"Rrod	H	DB-250-2200-125 CCAD	Hogan Fluid Power	(713) 741-0435
1	2.5" Bore Cylinder, 14" S, 1.25" Rod	H	DB-250-1400-125 CCAD	Hogan Fluid Power	(713) 741-0435
1	3.5" Bore Cylinder, 16" S, 1.5" Rod	H	DB-350-1600-150 CCBO	Hogan Fluid Power	(713) 741-0435
1	3.5" Bore Cylinder, 12" S, 1.5" Rod	H	DB-350-1200-150 CCBO	Hogan Fluid Power	(713) 741-0435
1	4" Bore Cylinder, 12" S, 2" Rod	H	DB-400-1200-200 CCBO	Hogan Fluid Power	(713) 741-0435
1	Hydraulic Manifold w/Valves	H	Various See MR	Automation Engineering	(713) 460-2317
6	Counter-Balance Valves	H	CP-440-1-1-6S-F-A-20-4.5-015	Automation Engineering	(713) 460-2317
1	Dual-Axis Joystick	C	MAURY JSP2303-2AS103	Ralph's Electronics	(713) 645-2211
1	Single-Axis Joystick	C	MAURY J2301-1AS103	Ralph's Electronics	(713) 645-2211
2	Multi-Function I/O Board	C	77633-01 National Instrument	National Instruments	(800) 433-3488
1	Set Slip Rings	A	180-0002-08 Focal Tech.	Focal Technologies	(902) 468-2263
1	Gear Motor, 1.5 hp, 24V dc	A	S72	T.L. Walker	RETURNED
1	24V dc Motor Controller	C	HBP-2	Dart Controls, Inc.	(317) 873-5211
1	Dry Altimeter	C	ISU-101-IS-ER-LI-H60-STR	Contaq Technologies	(802) 453-3332
2	Accustar Clinometer Vert. Mount	C	Lucas 02338-03	Mike Traylor Products, Inc.	(800) 580-8271
1	Accustar Clinometer Horz. Mount	C	Lucas 02706-01	Mike Traylor Products, Inc.	(800) 580-8271
1	Float Switch N.O.	C	50195K73	McMaster-Carr	(708) 833-0300
2	Float Switch N.C.	C	50195K74	McMaster-Carr	(708) 833-0300
1	2" Bore Bearing	H	TS39573	T.L. Walker	(713) 460-4000
1	2" Bearing Cup	H	TS39521	T.L. Walker	(713) 460-4000
6	Tilt Sensors & Clips	C	CA22-0, 1A, CMBR90	Comus International	(201) 667-6200
2	Rotary Encoders, 64 cpr	C	89F1680	Newark	(713) 688-6954
3	Toggle Switches SPDT	C	14F3132	Newark	(713) 688-6954
1	Locking Toggle SPDT	C	61F972	Newark	(713) 688-6954
1	24V 10-amp Power Supply	C	LFS-45A-24 Lambda	Lambda	(800) 526-2325
4	Digital Panel Meters	C	DPM 332-ND	Digikey	(800) 344-4539
1	Data Sonics Altimeter	C	PSA-900	C.A. Richards	(713) 531-7417
200'	Armoured Cable 7 Conductor	C	A2007	Vector Cable	(713) 274-7738
1	Telescoping Boom, 60'	C	QUOTE 93-011	Coastal Hydraulic Cranes	(713) 448-8998
1	Watertight Aluminum Enclosure	C	SIF3563	Newark	(713) 688-6954
8	Half-H Bridge Driver, 6 amps	C	UDN2955W-2	Newark	(713) 688-6954
4	12-Pin Terminal Strip	C	K182-ND	Newark	(713) 688-6954
8	Tilt Motor PCBs	C	NA	Hedcorp	(713) 688-6954
1	Hand-Box Enclosure, 5 x 7 x 2	C	90F943	Newark	(713) 688-6954
1	5 x 7 Enclosure Cover	C	90F860	Newark	(713) 688-6954
2	Modem Connectors, 50 Pin	C	90F4539	Newark	(713) 688-6954
100	Discrete Wire Crimp Terminals	C	90F4132	Newark	(713) 688-6954
1	Hook-Up Wire Kit	C	36F182	Newark	(713) 688-6954
1	Blue-Line Drawings & Reduction	DR	NA	Blue Prints Plus	(713) 468-7026
1	Fabricate Parts as per Drawings	T	NA	Merriworth Trading	(713) 466-7998
2	Blue-Line Prints	DR	NA	Blue Prints Plus	(713) 468-7026
2	Hamilton Caster (12 x 3)	T	RA-12MH	Aim	(713) 528-2000
3	Hamiltion Caster (12 x 4)	T	RMD-124-SDT	Aim	(713) 528-2000
2	Linear Actuator Clevis Bracket	A	BH-N227D035-G1	Custom Heliarc	
2	Linear Actuator Clevis Pivot	A	BH-N227D038-G1	Custom Heliarc	
1	Altimeter Pivot Shaft	A	BH-N227D031-G1	Custom Heliarc	
1	Cable Clamp	A	BH-N227D029-P1	Custom Heliarc	
1	Elbow-Pin Pivot	A	BH-N227D036-P1	Custom Heliarc	

CODES: T = TRAILER, A = ARM, H = HYDRAULICS, C = CONTROLS, F = FRAME, 0 = OPERATIONAL, DR = DRAWINGS

QTY	DESCRIPTION	CODE	VENDOR OR MFG & PART #	VENDOR	PHONE
1	Retainer Turn-Down Plate	A	BH-N227D039-P1	Custom Heliarc	
2	Elbow Pivot-Pin Retainer Plate	A	BH-N227D033-P1	Custom Heliarc	
1	Sonar Altimeter Housing	A	CH-N227D032-G1	Custom Heliarc	
1	Dry-Landing Altimeter	A	DH-N227D030-G1	Custom Heliarc	
1	Winch Drum	A	DH-N227D026-G1	Custom Heliarc	
1	Winch Drum Hub	A	CH-N227D028-P1	Custom Heliarc	
2	Hydraulic Adaptor Fitting	H	6MB X 6MJ	PowerTrac	(713) 864-4673
4	Hydraulic Adaptor Fitting	H	6MB X 6MJ90	PowerTrac	(713) 864-4673
1	Set Hydraulic Hoses	A	Attached List	PowerTrac	(713) 864-4673
14.5 ft	3/8" Hose, 2000 psi	A	NA	PowerTrac	(713) 864-4673
11.5 ft	3/8" Hose, 2000 psi	A	NA	PowerTrac	(713) 864-4673
3	24V dc Contractor, 100 amps	T	50F3665	Newark	(713) 6886954
1	Muffin Fan, 115 ac	T	81F8103	Newark	(713) 688-6954
1	Fan Finger Guard	T	81F2738	Newark	(713) 688-6954
1	Right-Angle Plug&Cord	T	81F8125	Newark	(713) 688-6954
2	Clamp Half	T	SKWB070193-1	SFR Gray Steel	
2	Stiffener Plates	T	SKWB070193-2	SFR Gray Steel	
1	Sonar Retainer	A	AH-N227D089- P1	Electro-Mechanical	
24	#4 x 3/8 Crimp Ring Lugs	C	NA	Wholesale Electric	
1	3/4 - 10 Die	T	NA	Bass Tool	
1	Die Handle	T	NA	Bass Tool	
4	Handles	T	SKBB070893	A-1 Welding	
1	1" x 10" Flat Bar, 142" Long	T	NA	Sterling Steel	(713) 690-0347
1	Gallon Bright Yellow Paint	T	AK6400	American Coating	
1	Qt. Red Paint	T	AK6230	American Coating	
6	Dorn Tube, Size 41"	C	M19622/1-005	B&D Electric	
6	Dorn Glands, Size 4f	C	M19622/19-0007	B&D Electric	
6	Dorn Tube, Size 1	C	M19622/1-001	B&D Electric	
6	Dorn Glands, Size 1C	C	M19622/16-006	B&D Electric	
1	Lot Sheet Metal Parts	C	NA	Custom Fab	(713) 686-5691
2	Multi-function I/O Board	C	776333-01	National Instruments	
4	Power Op-Amps	C	LM12CLK-ND	Digikey	(800) 3444539
2	0.1-ohm, 5-Watt Precision Resistor	C	SC53-0.1	Digikey	(800) 344-4539
1	1000 pF Cap. 10 pack	C	P4152	Digikey	(800) 344-4539
4	10 uF Cap-Tantalum	c	P2102	Digikey	(800) 344-4539
2	5V Regulators, 150 Ma	C	LM78L054CZ	Digikey	(800) 3444539
4	10-Amp Diodes	C	PBYR1045PH-ND	Digikey	(800) 3444539
1	SFC 3104 100 kHz Clk	C	SFC3104	Digikey	(800) 344-4539
2	7 x 19" Rack Panels	C	91F391	Newark	(713) 688-6954
2	Digital Panel Meters	C	DPM-332-ND	Digikey	(800) 344-4539
5	Relays, 24V dc SPDT	C	90F3056	Newark	(713) 688-6954
1	Relay spst, 24V dc Panel Mount	C	89F3188	Newark	(713) 688-6954
1	Solid-State Relay	C	89F953	Newark	(713) 688-6954
1	Box-Insulated Female Disconnects	C	42F3731	Newark	(713) 688-6954
1	Box-Insulated Male Disconnects	C	42F3734	Newark	(713) 688-6954
1	Toggle Switch	C	14F3132	Newark	(713) 688-6954
2	Float Switch N.O.	C	50195K73	McMaster-Carr	(708) 833-0300
2	Float Switch N.C.	C	50195K74	McMaster-Carr	(708) 833-0300
1	Box Screws	T	NA	That Hardware Place	
1	Can Red Spray Paint	T	NA	That Hardware Place	
2	Rotary Encoder	C	Newark	Newark	(713) 688-6954
1	Vertical Inclinometer	C	Lucas	Lucas-Schaevitz	(800) 745-8008
1	1-Ton Transport Truck	O	NA	Texas Hot Shot	(713) 996-1533
1	Truck-to-Transport Bridge Insp.	O	NA	Texas Hot Shot	(713) 996-1533
1	Trailer License & Registration	O	NA	Carl S. Smith, tax assessor	
1	Truck-to-Transport Bridge Insp.	O	NA	Texas Hot Shot	(713) 996-1533
1	Bill Bath Expenses L	T	NA	Various	
2	Liquid Level Control	C	2A551	W.W. Grainger	(713) 462-1551
1	Beldon Cable	C	9964	Newark	(713) 688-6954

NA = not applicable

1 in = 25.4 mm
1 ft = 0.305 m
1 psi = 6.9 kPa
ton = 0.907 metric tons

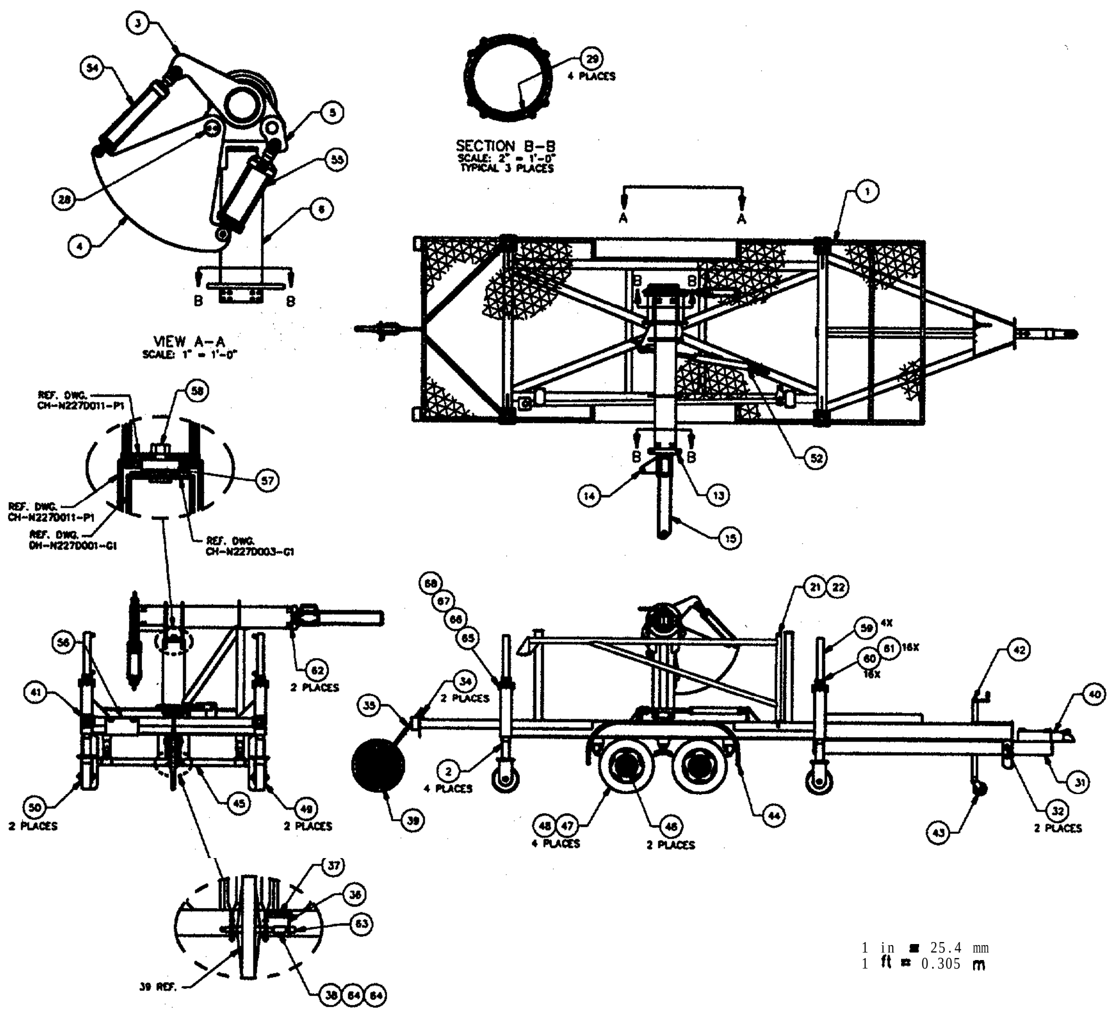


Figure 26. Mechanical drawings of Bridge Inspector and parts list.

PTY	REMARKS	DESCRIPTION	PART NO.	ITEM
1	STEEL	BRIDGE INSPECTOR TRAILER FRAME	DH-N227D001-G1	1
4	STEEL	STABILIZER LEG	CH-N227D002-G1	2
1	STEEL	ARM YAW TURN PLATE	CH-N227D005-P1	3
1	STEEL	TURN-DOWN PLATE	DH-N227D006-P1	4
1	STEEL	ARM-MOUNT ANCHOR PLATE	BH-N227D007-P1	5
1	STEEL	ARM-MOUNT	DH-N227D012-G1	6
1	STEEL	LATCH	CH-N227D014-G1	7
1	STEEL	LATCH PIVOT SHAFT	BH-N227D015-P1	8
1	STEEL	LATCH PIVOT SHAFT CLAMP	BH-N227D016-P1	9
1	STEEL	LATCH SPRING RETAINER	BH-N227D017-P1	10
1	STEEL	LATCH PIVOT SHAFT LUG	BH-N227D018-P1	11
1	STEEL	SPRING RETAINER MOUNT	BH-N227D019-P1	12
1	STEEL	RETAINING COLLAR	CH-N227D020-P1	13
1	STEEL	ELBOW CYLINDER ROD END MOUNT	CH-N227D092-G1	14
1	STEEL	ARM	DH-N227D023-G1	15
1	STEEL	LATCH SHAFT RETAINER MOUNT	BH-N227D025-P1	16
1	STEEL	WINCH DRUM	DH-N226D026-G1	17
1	STEEL	ELBOW	DH-N227D027-G1	18
1	STEEL	WINCH DRUM HUB	CH-N227D028-P1	19
1	STEEL	CABLE CLAMP	BH-N227D029-P1	20
1	STEEL	DRY-LAND ALTIMETER ARM	DH-N227D030-G1	21
1	STEEL	DRY-LAND ALTIMETER PIVOT SHAFT	BH-N227D031-G1	22
1	STEEL	SONAR ALTIMETER HOUSING	CH-N227D032-G1	23
1	STEEL	ELBOW-PIVOT PIN RETAINER PLATE	BH-N227D033-P1	24
1	STEEL	ELBOW PIN	BH-N227D034-P1	25
1	STEEL	LINEAR ACTUATOR CLEVIS BRACKET	BH-N227D035-G1	26
1	STEEL	ARM PIN	BH-N227D036-P1	27
1	STEEL	RETAINER TURN-DOWN PLATE	BH-N227D039-P1	28
12	UHMWPE	BEARING PAD	BH-N227D040-P1	29
1	STEEL	SONAR ALTIMETER PLATE	DH-N227D042-G1	30
1	STEEL	TRAILER PULL BAR	CH-N227D043-G1	31
2	STEEL	PULL-BAR PIN	BH-N227D044-P1	32
1	STEEL	SPACER	BH-N227D046-P1	33
1	STEEL	ODOMETER MOUNTING LUG	BH-N227D080-P1	34
1	STEEL	ODOMETER PIVOT ARM	BH-N227D081-P1	35
1	STEEL	ODOMETER ENCODER MOUNT	BH-N227D082-P1	36
1	STEEL	ANTIROTATION MOUNT	BH-N227D083-P1	37
1	STEEL	ODOMETER - SENSOR COUPLING	BH-N227D084-P1	38
1	MATHEWS	24" UNICYCLE (ODOMETER)		39
1	DICO	HYD. BRAKE ACTUATOR #10 (2 5/16" BALL)	16075	40
1	HUSKY	LIGHTING KIT		41
1	HUSKY	TONGUE-MOUNTED JACK, 5000 lb.		42
1	HUSKY	JACK CASTER, 5000 lb.		43
2	HUSKY	FENDER, 10" X 72"		44
1	HUSKY	TANDEM AXLE KIT, 12,000 lb.		45
1	TWF	12" HYD. BRAKES		46
1	HUSKY	TRAILER WHEEL, 16" X 6"		47
1		TIRES, 16" X 7.50		48
2	HAMILTON	STEEL CASTER (OVERALL HEIGHT 15 1/2")	R-MD-124SDT	49
2	HAMILTON	STEEL CASTER (OVERALL HEIGHT 14")	R-A-12DH	50
1	CROSS	HYD. CYL. 2 1/2" BORE, 22" STROKE	DB250-2200-125 PCA0	51
1	CROSS	HYD. CYL. 2 1/2" BORE, 14" STROKE	DB250-1400-125 PCA0	52
1	CROSS	HYD. CYL. 3 1/2" BORE, 16" STROKE	DB350-1600-150 RCB0	53
1	CROSS	HYD. CYL. 3 1/2" BORE, 12" STROKE	DB350-1200-150 RCB0	54
1	CROSS	HYD. CYL. 5" BORE, 12" STROKE	DB500-1200-200 RCB0	55
1	HUSKY	LICENSE PLATE LIGHT & BRACKET		56
1	TIMKEN	BEARING, 2" BORE	39573 - 39521	57
1	CAD PLT'D	SHOULDER BOLT, 1 1/4" X 1 7/8"		58
4	-	3" BORE HYDRAULIC CYLINDER	-	59
16	A-307	CAPSCREW 1/2"-13 UNC-2A X 2" LG.		60
16	-	LOCK WASHER, 1/2"	-	61
2	UHMWPE	ARM THRUST WASHER	BH-N227D049-P1	62
1		ODOMETER ENCODER		63
2		3/32" ROLL PIN		64
4	ASTM A36	LEVELING CYLINDER MTG. PLATE	CH-N227D093-G1	65
16	A-307	CAPSCREW 3/4"-10UNC-2A X 2 1/2" LG.	-	66
16	A-307	HEX NUT 3/4"-10 UNC-2B		67
16	A-367	LOCKWASHER, 3/4"		68

1 in = 25.4 mm
1 lb = 0.454 kg

NOTE: Detailed mechanical drawings are on file at:
Turner-Fairbank Highway Research Center
HNR-10
6300 Georgetown Pike
McLean, VA 22 10 l-2296

Telephone: (703) 285-2474

Refer to archive file for contract
DTFH6 l-9 1 -C-O0060 with Sonsub, Inc.

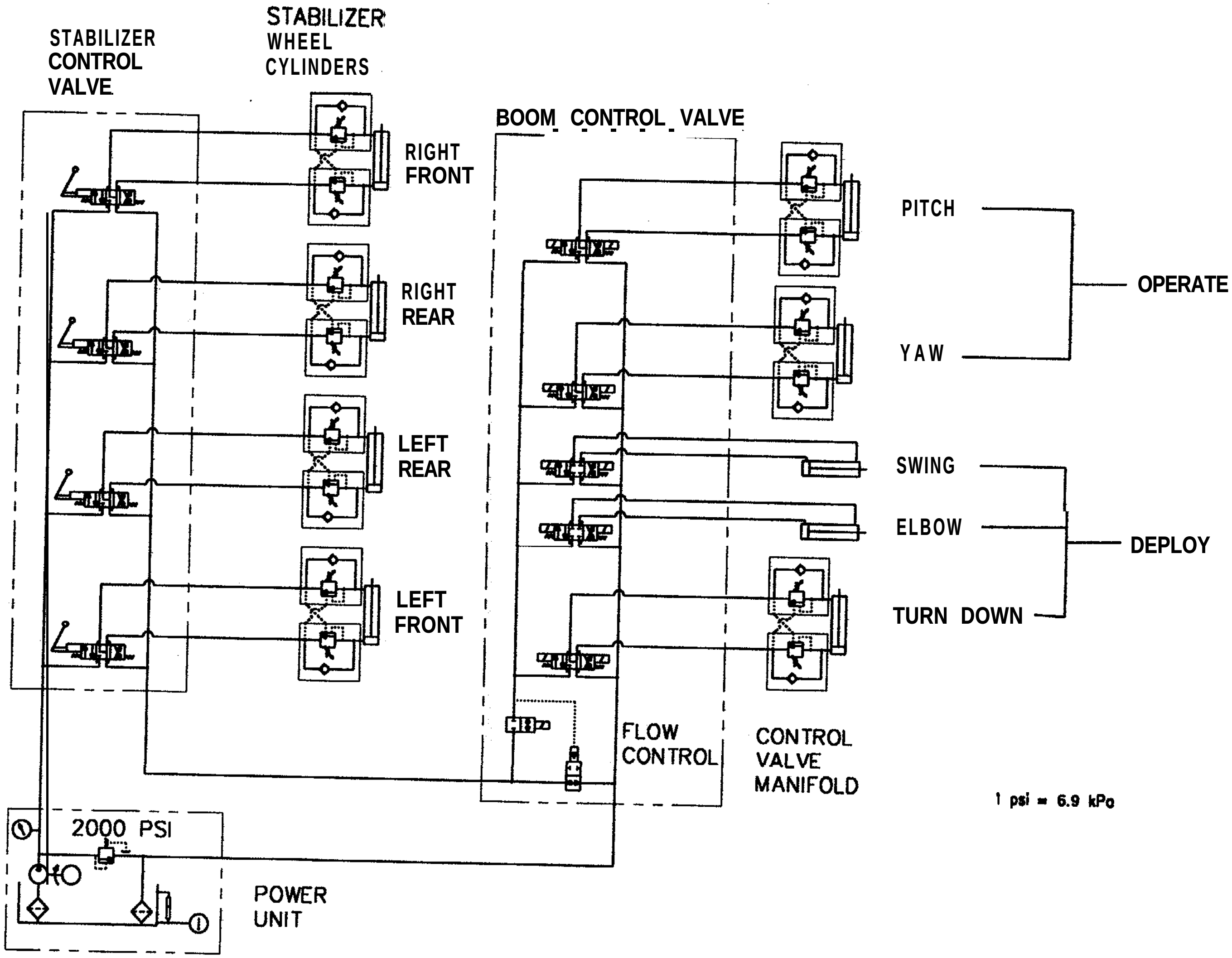


Figure 27. Bridge inspector hydraulic circuit.

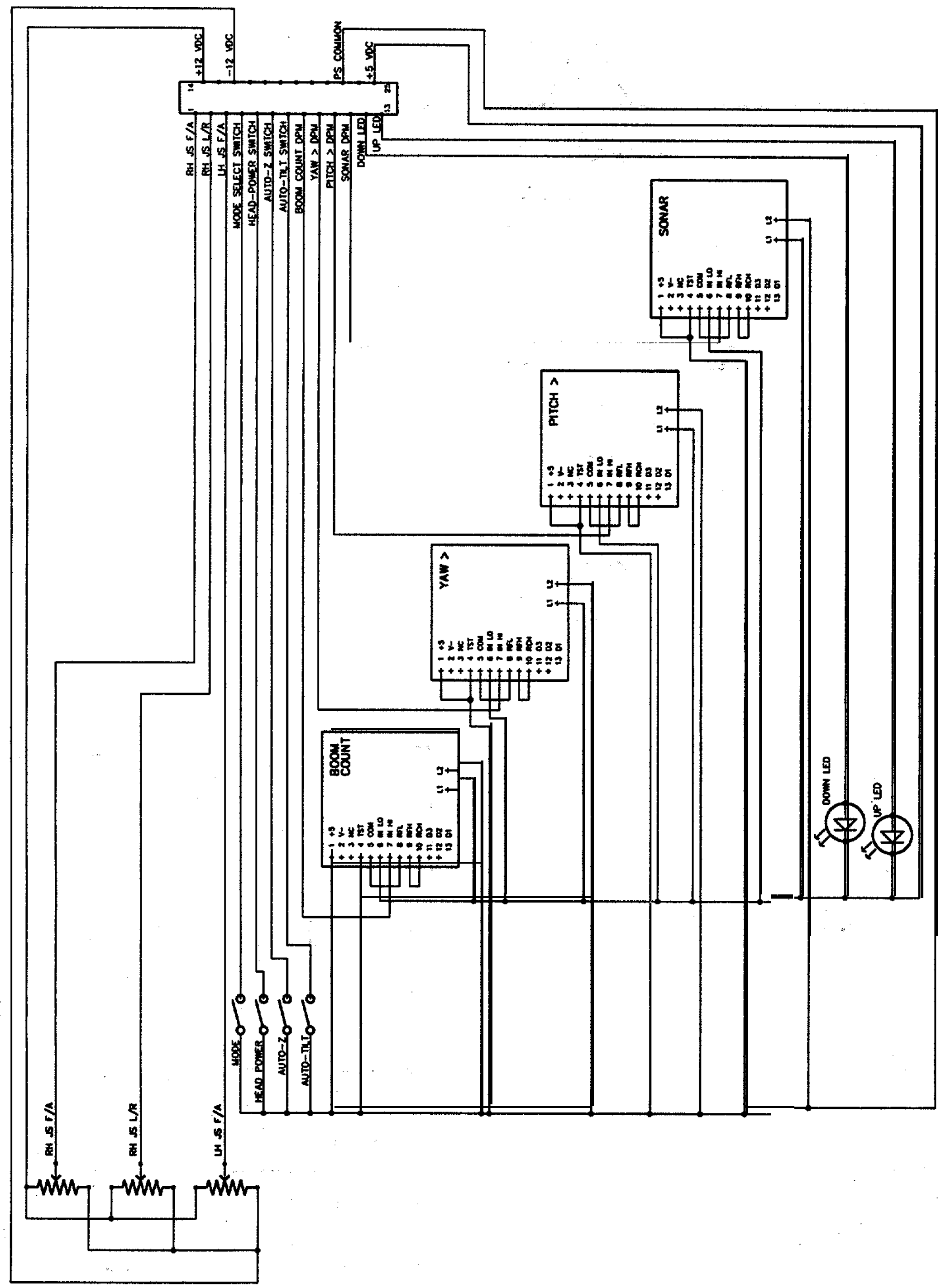
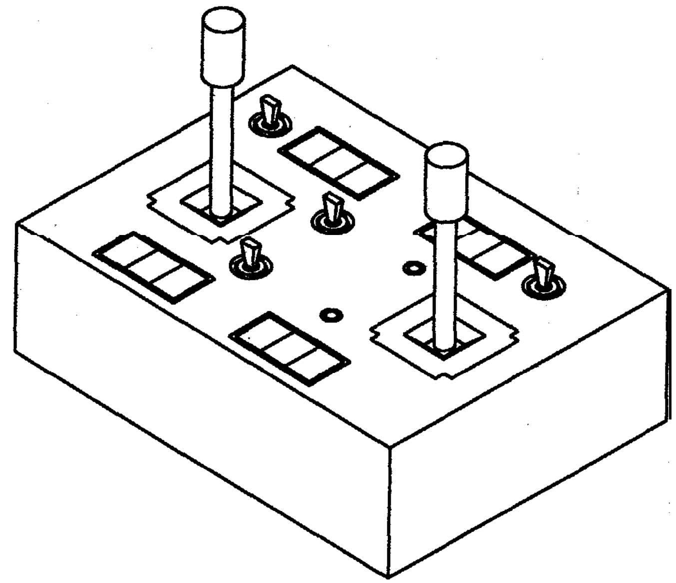


Figure 28. Consolette wiring schematic.

1. CUSTOM CONTROL BOX
2. JOYSTICK JSP-2302
3. DIGITAL PANEL METERS
4. SWITCHES TT13A-6T
5. LED INDICATORS



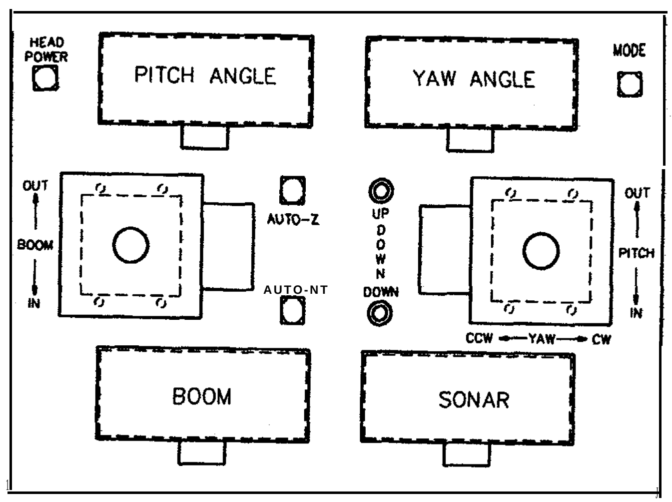


Figure 29. Hand control box layout.

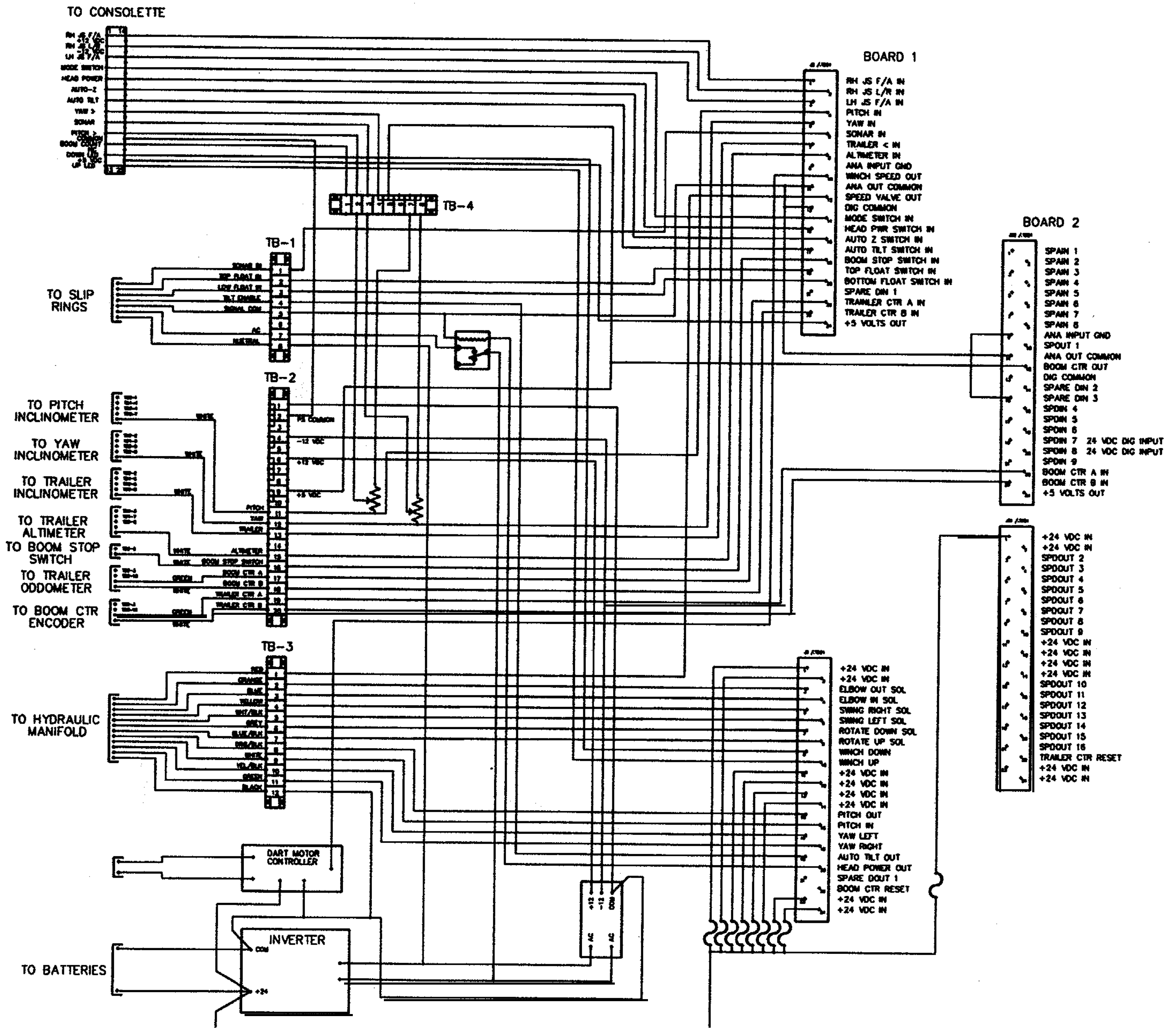


Figure 30. Console wiring diagram.

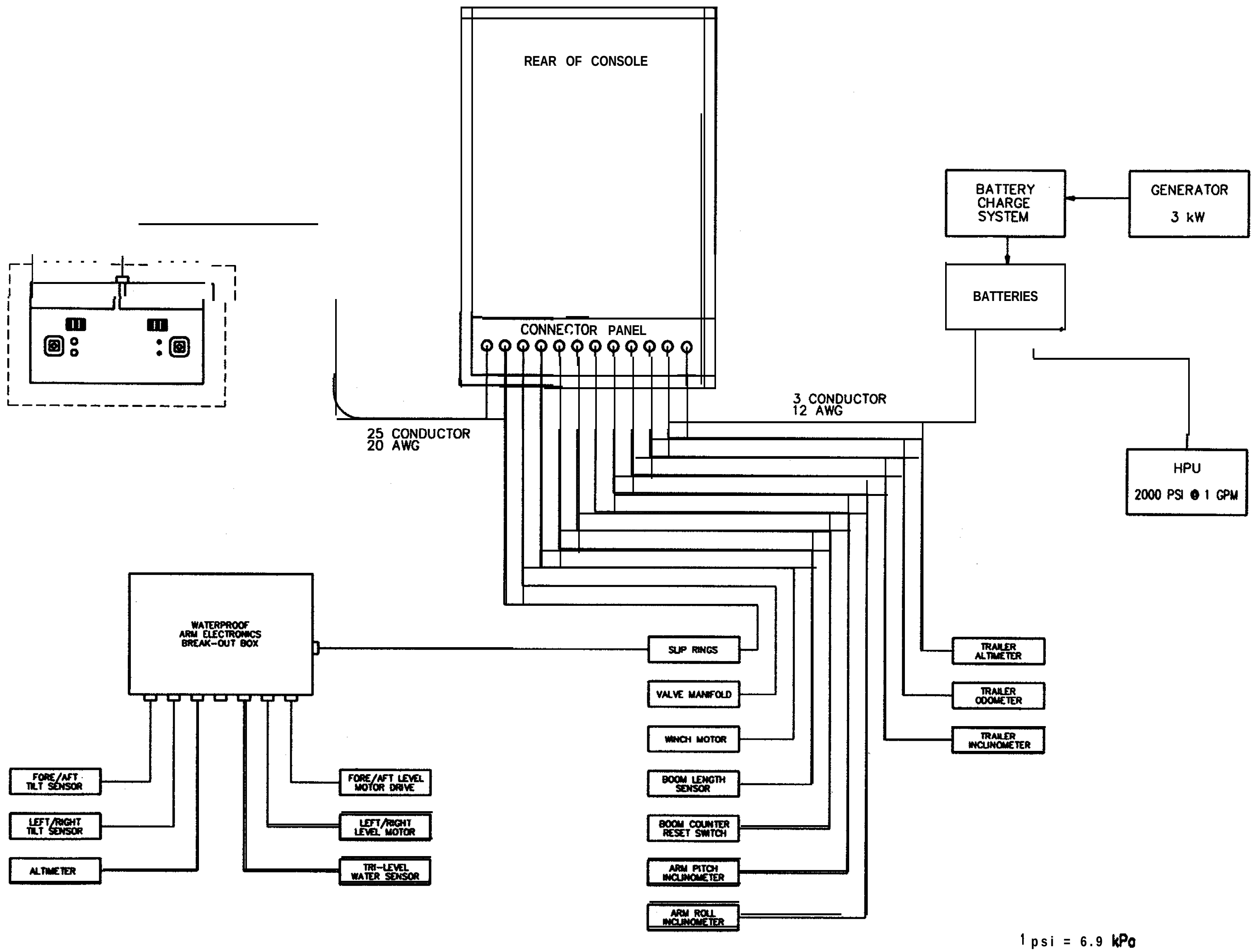


Figure 3 1. Bridge inspector general arrangement diagram.

APPENDIX E. BRIDGE PROGRAM LISTINGS

```
DEFINT A-Z
Xa! = Yaw Angle, Ya! = Pitch Angle, BL! = Boom Length
'Sa! = Sonar Reading, Od! = Odometer, Ia! = Trailer Inclinometer
'La! = Laser Reading, Xf! = Yaw in Feet, Yf! = Pitch in Feet
'Zf! = Height Boom Pivot to Water, Hb! = Bridge Height Relative to Start
'Xb! = X' Relative to Start, Yb! = Yf! - Yos!, Yos! = Hor Dist Deck Edge to Center Pin
'Zb! = Zf! - Zos!, Zos! = Boom Pivot to Deck, Lb! = La! - Los!
'Los! = Vert Dist Deck to Laser, Zt! = Sa! + Sf!
'Xtt! = Trailer X = Station + Ta, Xbt! = Xtt! + Xb!
'Ed! = World Deck Elevation = Hb! + Elevation, Eb! = Bed Elev in World = Elevation - Zt
'Ls! = Surface Elev in World = Elevation - Lb!

DECLARE SUB BRIDGEGRAPH (BGCount)
DECLARE SUB BRIDGELOG ()
DECLARE SUB CALCULATE (Station!, Elevation, BGCount)
DECLARE SUB DIAGNOSTICS ()
DECLARE SUB Err.Exit (procname$, err.num%)
DECLARE SUB FILEOPEN (EXT$, Valid%, ErrorNumber)
DECLARE SUB HELP ()
DECLARE SUB INIT ()
DECLARE SUB MENU1 ()
DECLARE SUB MENU1FUNCTIONS (KeyPress$)
DECLARE SUB MENU2 ()
DECLARE SUB MENU2FUNCTIONS (KeyPress$)
DECLARE SUB PARAMETERS ()
DECLARE SUB PILEGRAPH (PilePass, PCount)
DECLARE SUB PILELOG ()
DECLARE SUB READINPUTS ()
DECLARE SUB SCREENWRITE (Station!, Elevation)
DECLARE SUB SCRNPRNT ()
DECLARE SUB COUNTRESET (CountKey!)
DECLARE SUB WRITEOUTPUTS ()

COMMON SHARED TGLen, TGHite, PHite, ErrorNumber, MenuKey
COMMON SHARED Keyl, Key2, Key3, Key9, TrailNum, Trail
COMMON SHARED BridgeFile$, PileFile$, Stream%, Direction%
COMMON SHARED BName$, CName$, SName$, TName$, PName$
COMMON SHARED TaOffset!, OffSet!, BoomOffset!, Yos!, Zos!, Los!
COMMON SHARED Xa!, Ya!, BL!, Sa!, Od!, Ia!, La!, Xf?, Yf?, Zf!, Hb!
COMMON SHARED Xb!, Yb!, Zb!, Lb!, Zt!, Xtt!, Xbt!, Ed!, Ls!, Eb!
COMMON SHARED Yaw0#, YawSpan#, PitchO#, PitchSpan#, BL0#, BLSpan#
COMMON SHARED TAngle0#, TAngleSpan#, Odd0#, OddSpan#, Sonar0#, SonarSpan#
COMMON SHARED Laser0#, LaserSpan#

'$INCLUDE: 'NIDAQ.INC'
'
' "Include" file for Microsoft QuickBASIC programs
' Copyright (C) National Instruments 1992.
' nidaq.inc

DECLARE FUNCTION A2000.Calibrate% CDECL ALIAS "_A2000_Calibrate" (BYVAL a%, BYVAL b%, BYVAL
c%, BYVAL d%, BYVAL e#)
DECLARE FUNCTION A2000.Config% CDECL ALIAS "_A2000_Config" (BYVAL a%, BYVAL b%, BYVAL c%,
BYVAL d%)
```

DECLARE FUNCTION A2 1 50.Calibrate% CDECL ALIAS "_A21 SO-Calibrate" (BWAL a%, BWAL b%, BWAL c%)
DECLARE FUNCTION AI.Check% CDECL ALIAS "-AI-Check" (BWAL a%, SEG b%, SEG c%)
DECLARE FUNCTION AI.Clear% CDECL ALIAS "_AI_Clear" (BWAL a%)
DECLARE FUNCTION AI.Configure% CDECL ALIAS "-AI-Configure" (BWAL a%, BWAL b%, BYVAL c%, BWAL d%, BWAL e%, BWAL f%)
DECLARE FUNCTION AI.Mux.Config% CDECL ALIAS "_AI_Mux_Config" (BWAL a%, BYVAL b%)
DECLARE FUNCTION AI.Read% CDECL ALIAS "_AI_Read" (BWAL a%, BWAL b%, B WAL c%, SEG d%)
DECLARE FUNCTION AI.Setup% CDECL ALIAS "-AI-Setup" (BWAL a%, BWAL b%, BWAL c%)
DECLARE FUNCTION AI.VRead% CDECL ALIAS "_AI_VRead" (BWAL a%, BWAL b%, BWAL c%, SEG d#)
DECLARE FUNCTION AI.VScale% CDECLALIAS "_AI_VScale" (BWAL a%, BWAL b%, BWAL c%, BWAL d#, BWAL e#, BWAL f%, SEG g#)
DECLARE FUNCTION Align.DMA.Buffer% CDECL ALIAS "_qb_alignDMAbuffer" (BWAL a%, BYVAL b%, c%(), BWAL d%, BWAL e&, SEG f%)
DECLARE FUNCTION AO.Calibrate% CDECL ALIAS "_AO_Calibrate" (B WAL a%, B WAL b%, B WAL c%)
DECLARE FUNCTION AO.Config% CDECL ALIAS "_AO_Config" (BWAL a%, BWAL b%, BWAL c%, BWAL d#, BYVAL e%)
DECLARE FUNCTION AO.Configure% CDECL ALIAS "_AO_Configure" (B WAL a%, B WAL b%, B WAL c%, BYVAL d%, B WAL e#, B WAL f%)
DECLARE FUNCTION AO.Update% CDECL ALIAS "_AO_Update" (BWAL a%)
DECLARE FUNCTION AO.VScale% CDECL ALIAS "_AO_VScale" (BWAL a%, BWAL b%, B WAL c#, SEG d%)
DECLARE FUNCTION AO.VWrite% CDECL ALIAS "_AO_VWrite" (BWAL a%, BWAL b%, BWAL c#)
DECLARE FUNCTION AO. Write% CDECL ALIAS "_AO_Write" (B WAL a%, B WAL b%, B WAL c%)
DECLARE FUNCTION CTR.Config% CDECL ALIAS "_CTR_Config" (B WAL a%, B WAL b%, B WAL c%, B WAL d%, BWAL e%, B WAL f%)
DECLARE FUNCTION CTR.Evcount% CDECL ALIAS "_CTR_Evcount" (B WAL a%, B WAL b%, BYVAL c%, B WAL d%)
DECLARE FUNCTION CTR.Evread% CDECL ALIAS "_CTR_Evread" (BWAL a%, BWAL b%, SEG c%, SEG d%)
DECLARE FUNCTION CTR.FOUT.Config% CDECL ALIAS "_CTR_FOUT_Config" (B WAL a%, B WAL b%, B WAL c%, BYVAL d%, B WAL e%)
DECLARE FUNCTION CTR.Period% CDECL ALIAS "_CTR_Period" (BWAL a%, BWAL b%, B WAL c%)
DECLARE FUNCTION CTR.Pulse% CDECL ALIAS "_CTR_Pulse" (B WAL a%, B WAL b%, B WAL c%, BWAL d%, BWAL e%)
DECLARE FUNCTION CTR.Rate% CDECL ALIAS "_CTR_Rate" (BYVAL a#, B WAL b#, SEG c%, SEG d%, SEG e%)
DECLARE FUNCTION CTR.Reset% CDECL ALIAS "_CTR_Reset" (BWAL a%, BWAL b%, BWAL c%)
DECLARE FUNCTION CTR.Restart% CDECL ALIAS "_CTR_Restart" (BWAL a%, BWAL b%)
DECLARE FUNCTION CTR.Simul.Op% CDECL ALIAS "_qb_ctrsimulop" (B WAL a%, B WAL b%, c%(), BYVAL d%)
DECLARE FUNCTION CTR.Square% CDECL ALIAS "_CTR_Square" (BYVAL a%, BWAL b%, BYVAL c%, BYVAL d%, B WAL e%)
DECLARE FUNCTION CTR.State% CDECL ALIAS "_CTR_State" (B WAL a%, B WAL b%, SEG c%)
DECLARE FUNCTION CTR.Stop% CDECL ALIAS "_CTR_Stop" (BWAL a%, BWAL b%)
DECLARE FUNCTION DAQ.Check% CDECL ALIAS "_DAQ_Check" (BWAL a%, SEG b%, SEG c%)
DECLARE FUNCTION DAQ.Clear% CDECL ALIAS "_DAQ_Clear" (BYVAL a%)
DECLARE FUNCTION DAQ.Config% CDECL ALIAS "_DAQ_Config" (BYVAL a%, BWAL b%, BYVAL c%)
DECLARE FUNCTION DAQ.DB.Config% CDECL ALIAS "_DAQ_DB_Config" (BWAL a%, BWAL b%)
DECLARE FUNCTION DAQ.DB.HalfReady% CDECL ALIAS "_DAQ_DB_HalfReady" (BWAL a%, SEG b%, SEG c%)
DECLARE FUNCTION DAQ.DB.StrTransfer% CDECLALIAS "_qb_daqdbstrxfr" (BYVAL a%, b$, SEG c%, SEG d%)
DECLARE FUNCTION DAQ.DB.Transfer% CDECL ALIAS "_qb_daqdbxfr" (a%, b%(), c%, d%)
DECLARE FUNCTION DAQ.Monitor% CDECL ALIAS "_qb_daqmonitor" (a%, b%, c%, d%, e%(), f%, g%)

DECLARE FUNCTION DAQ.Op% CDECL ALIAS "_qb_daqop" (a%, b%, c%, d%(), e%, f#)

DECLARE FUNCTION DAQ.Rate% CDECL ALIAS "_DAQ_Rate" (BWAL a#, BWAL b%, SEG c%, SEG d%)

DECLARE FUNCTION DAQ.Start% CDECL ALIAS "_qb_daqstart" (a%, b%, c%, d%(), e%, f%, g%)

DECLARE FUNCTION DAQ.StopTrigger.Config% CDECL ALIAS "_DAQ_StopTrigger_Config" (BWAL a%, BWAL b%, BWAL c&)

DECLARE FUNCTION DAQ.to.Disk% CDECL ALIAS "_qb_daqtodisk" (BWAL a%, BWAL b%, BWAL c%, d$, BWAL e&, BWAL f#, BWAL g%)

DECLARE FUNCTION DAQ.Trigger.Config% CDECL ALIAS "_DAQ_Trigger_Config" (BWAL a%, BWAL b%, BWAL c%)

DECLARE FUNCTION DAQ.VScale% CDECL ALIAS "_qb_daqvscale" (BWAL a%, BWAL b%, BWAL c%, B WAL d#, B WAL e#, B WAL f%, g%(), h#())

DECLARE FUNCTION DIG.Block.Check% CDECL ALIAS "_DIG_Block_Check" (BWAL a%, BWAL b%, SEG c%)

DECLARE FUNCTION DIG.Block.Clear% CDECL ALIAS "_DIG_Block_Clear" (BWAL a%, BWAL b%)

DECLARE FUNCTION DIG.Block.In% CDECL ALIAS "_qb_digblkin" (BYVAL a%, BWAL b%, c%(), BWAL d%)

DECLARE FUNCTION DIG.Block.Out% CDECL ALIAS "_qb_digblkout" (BWAL a%, BWAL b%, c%(), BWAL d%)

DECLARE FUNCTION DIG.Block.PG.Config% CDECL ALIAS "_DIG_Block_PG_Config" (BWAL a%, BWAL b%, BWAL c%, BYVAL d%, BYVAL e%, BYVAL f%, BWAL g%)

DECLARE FUNCTION DIG.DB.Config% CDECL ALIAS "_DIG_DB_Config" (BYVAL a%, BYVAL b%, BWAL c%, BYVAL d%, BYVAL e%)

DECLARE FUNCTION DIG.DB.HalfReady% CDECL ALIAS "_DIG_DB_HalfReady" (BWAL a%, BWAL b%, SEG c%)

DECLARE FUNCTION DIG.DB.StrTransfer% CDECL ALIAS "_qb_digdbstrxfr" (B WAL a%, B WAL b%, c$, BWAL d%)

DECLARE FUNCTION DIG.DB.Transfer% CDECL ALIAS "_qb_digdbxfr" (BWAL a%, BWAL b%, c%(), B WAL d%)

DECLARE FUNCTION DIG.Grp.Config% CDECL ALIAS "_DIG_Grp_Config" (BYVAL a%, BYVAL b%, BYVAL c%, B WAL d%, B WAL e%)

DECLARE FUNCTION DIG.Grp.Mode% CDECL ALIAS "_DIG_Grp_Mode" (B WAL a%, BYVAL b%, B WAL c%, BYVAL d%, BWAL e%, BYVAL f%, B WAL g%)

DECLARE FUNCTION DIG.Grp.Status% CDECL ALIAS "_DIG_Grp_Status" (BYVAL a%, BYVAL b%, SEG c%)

DECLARE FUNCTION DIG.In.Grp% CDECL ALIAS "_DIG_In_Grp" (BYVAL a%, BWAL b%, SEG c%)

DECLARE FUNCTION DIG.In.Line% CDECL ALIAS "_DIG_In_Line" (BWAL a%, BWAL b%, BWAL c%, SEG d%)

DECLARE FUNCTION DIG.In.Port% CDECL ALIAS "_DIG_In_Port" (BYVAL a%, BWAL b%, SEG c%)

DECLARE FUNCTION DIG.Line.Config% CDECL ALIAS "_DIG_Line_Config" (BWAL a%, BWAL b%, BYVAL c%, B WAL d%)

DECLARE FUNCTION DIG.Out.Grp% CDECL ALIAS "_DIG_Out_Grp" (BYVAL a%, BWAL b%, BWAL c%)

DECLARE FUNCTION DIG.Out.Line% CDECL ALIAS "_DIG_Out_Line" (BYVAL a%, BWAL b%, BWAL c%, BYVAL d%)

DECLARE FUNCTION DIG.Out.Port% CDECL ALIAS "-DIG-Out-Port" (BYVAL a%, BWAL b%, BYVAL c%)

DECLARE FUNCTION DIG.Prt.Config% CDECL ALIAS "_DIG_Prt_Config" (BYVAL a%, BWAL b%, BWAL c%, BYVAL d%)

DECLARE FUNCTION DIG.Prt.Status% CDECL ALIAS "_DIG_Prt_Status" (B WAL a%, BWAL b%, SEG c%)

DECLARE FUNCTION DIG.SCAN.Setup% CDECL ALIAS "_qb_DIG_SCAN_Setup" (BWAL a%, BWAL b%, BYVAL c%, d%(), BYVAL e%)

DECLARE FUNCTION DSP2200.Calibrate% CDECL ALIAS "_DSP2200_Calibrate" (BYVAL a%, BYVAL b%, BYVAL c%)

DECLARE FUNCTION DSP2200.Config% CDECL ALIAS "_DSP2200_Config" (BWAL a%, BWAL b%, B WAL c%, BYVAL d%)

DECLARE FUNCTION Get.DA.Brds.Info% CDECL ALIAS "_Get_DA_Brds_Info" (BYVAL a%, SEG b%, SEG c%, SEG d%, SEG e%, SEG f%, SEG g%, SEG h%, SEG i%)

DECLARE FUNCTION Get.NI.DAQ.Version% CDECL ALIAS "_Get_NI_DAQ_Version" (SEG b&)

DECLARE FUNCTION ICTR.Read% CDECL ALIAS "_ICTR_Read" (BWAL a%, BWAL b%, SEG c%)
DECLARE FUNCTION ICTR.Reset% CDECL ALIAS "_ICTR_Reset" (BWAL a%, BYVAL b%, BWAL c%)
DECLARE FUNCTION ICTR.Setup% CDECL ALIAS "_ICTR_Setup" (BWAL a%, BWAL b%, BWAL c%, B WAL d%, B WAL e%)
DECLARE FUNCTION Init.DA.Brds% CDECL ALIAS "_Init_DA_Brds" (BWAL a%, SEG b%)
DECLARE FUNCTION Lab.ISCAN.Check% CDECL ALIAS "_qb_labIscancheck" (BWAL a%, SEG b%, SEG c%, d%())
DECLARE FUNCTION Lab.ISCAN.Op% CDECL ALIAS "_qb_labIscanop" (a%, b%, c%, d%(), e%, f#, g#, h%())
DECLARE FUNCTION Lab.ISCAN.Start% CDECL ALIAS "_qb_labIscanstart" (a%, b%, c%, d%(), e%, f%, g%, h%)
DECLARE FUNCTION Lab.ISCAN.to.Disk% CDECL ALIAS "_qb_labIscantodisk" (BWAL a%, BWAL b%, BWAL c%, d$, BYVAL e&, BWAL f#, BWAL g#, BWAL h%)
DECLARE FUNCTION LPM 16.Calibrate% CDECL ALIAS "_LPM16_Calibrate" (BWAL a%)
DECLARE FUNCTION MAI.Arm% CDECL ALIAS "_MAI_Arm" (BWAL a%, BWAL b%)
DECLARE FUNCTION MAI.Clear% CDECL ALIAS "_MAI_Clear" (BWAL a%)
DECLARE FUNCTION MAI.Coupling% CDECL ALIAS "_qb_maiCoupling" (BYVAL a%, BYVAL b%, c%())
DECLARE FUNCTION MAI.Read% CDECL ALIAS "_qb_maiRead" (BWAL a%, b%())
DECLARE FUNCTION MAI.Scale% CDECL ALIAS "_qb_maiScale" (BWAL a%, BWAL b&, c%(), d#())
DECLARE FUNCTION MAI.Setup% CDECL ALIAS "_qb_maiSetup" (B WAL a%, BWAL b%, c%(), d%(), B WAL e%, BYVAL f%, B WAL g%)
DECLARE FUNCTION Master.Slave.Config% CDECL ALIAS "_qb_masterSlaveConfig" (BWAL a%, BWAL b%, c%())
DECLARE FUNCTION MDAQ.Check% CDECL ALIAS "_MDAQ_Check" (BWAL a%, BYVAL b%, SEG c%, SEG d&, SEG e&)
DECLARE FUNCTION MDAQ.Clear% CDECL ALIAS "_MDAQ_Clear" (BWAL a%)
DECLARE FUNCTION MDAQ.Ext.Setup% CDECL ALIAS "_MDAQ_Ext_Setup" (B WAL a%, B WAL b&, BWAL c%, BYVAL d&, BWAL e&, BWAL f&)
DECLARE FUNCTION MDAQ.Get% CDECL ALIAS "_qb_mdaqGet" (a%, b%, c%, d&, e&, f&, g&, h%(), i&, j&, k&, 1%)
DECLARE FUNCTION MDAQ.ScanRate% CDECL ALIAS "_MDAQ_ScanRate" (B WAL a%, B WAL b%, BWAL c%)
DECLARE FUNCTION MDAQ.Setup% CDECL ALIAS "_qb_mdaqSetup" (a%, b&, c%, d&, e&, f%())
DECLARE FUNCTION MDAQ.Start% CDECL ALIAS "_MDAQ_Start" (B WAL a%, BWAL b&)
DECLARE FUNCTION MDAQ.Stop% CDECL ALIAS "_MDAQ_Stop" (BWAL a%)
DECLARE FUNCTION MDAQ.StrGet% CDECL ALIAS "_qb_mdaqStrGet" (BYVAL a%, BYVAL b%, BWAL c%, BWAL d&, BWAL e&, BWAL f&, BYVAL g&, h$, SEG i&, SEG j&, SEG k&, SEG 1%)
DECLARE FUNCTION MDAQ.Trig.Delay% CDECL ALIAS "_MDAQ_Trig_Delay" (BWAL a%, BYVAL b%, B WAL c%)
DECLARE FUNCTION MDAQ.Trig.Select% CDECL ALIAS "_MDAQ_Trig_Select" (BWAL a%, BWAL b%, B WAL c%, BYVAL d%, BWAL e%, BYVAL f%, BYVAL g%)
DECLARE FUNCTION MIO.Calibrate% CDECL ALIAS "_MIO_Calibrate" (BYVAL a%, BYVAL b%, BYVAL c%, BWAL d%, BYVAL e%, BWAL f%, BWAL g%, BYVAL h#, BYVAL i%)
DECLARE FUNCTION MIO.Config% CDECL ALIAS "_MIO_Config" (BYVAL a%, BWAL b%, BYVAL c%)
DECLARE FUNCTION REG.Level.Read% CDECL ALIAS "_REG_Level_Read" (BWAL a%, BYVAL b%, SEG c&)
DECLARE FUNCTION REG.Level.Write% CDECL ALIAS "_REG_Level_Write" (BYVAL a%, BYVAL b%, BYVAL c&, BYVAL d&, SEG e&)
DECLARE FUNCTION RTSI.Clear% CDECL ALIAS "_RTSI_Clear" (BYVAL a%)
DECLARE FUNCTION RTSI.Clock% CDECL ALIAS "_RTSI_Clock" (BWAL a%, BYVAL b%, BYVAL c%)
DECLARE FUNCTION RTSI.Conn% CDECL ALIAS "_RTSI_Conn" (BWAL a%, BWAL b%, BYVAL c%, B WAL d%)
DECLARE FUNCTION RTSI.DisConn% CDECL ALIAS "_RTSI_DisConn" (BYVAL a%, BWAL b%, BYVAL c%)
DECLARE FUNCTION SCAN.Demux% CDECL ALIAS "_qb_scandemux" (a%(), BYVAL b%, BYVAL c%, B WAL d%)

```
DECLARE FUNCTION SCAN.Op% CDECL ALIAS "_qb_scanop" (a%, b%, c%(), d%(), e%(), f%, g#, h#)
DECLARE FUNCTION SCAN.Setup% CDECL ALIAS "_qb_scansetup" (BYVAL a%, BYVAL b%, c%(), d%())
DECLARE FUNCTION SCAN.Start% CDECL ALIAS "_qb_scanstart" (a%, b%(), c%, d%, e%, f%, g%)
DECLARE FUNCTION SCAN.to.Disk% CDECL ALIAS "_qb_scantodisk" (BYVAL a%, BWAL b%, c%(), d%(),
e$, BWAL f&, BWALg#, BWAL h#, BWAL i%)
DECLARE FUNCTION SCXI.Calibrate.Setup% CDECL ALIAS "_SCXI_Calibrate_Setup" (BYVAL a%, BWAL
b%, BWAL c%)
DECLARE FUNCTION SCXI.Change.Chan% CDECL ALIAS "_SCXI_Change_Chan" (BYVAL a%, BYVAL b%,
BWAL c%)
DECLARE FUNCTION SCXI.Get.Chassis.Info% CDECL ALIAS "_SCXI_Get_Chassis_Info" (BWAL a%, SEG b%,
SEG c%, SEG d%, SEC e%, SEG f%)
DECLARE FUNCTION SCXI.Get.Module.Info% CDECL ALIAS "_SCXI_Get_Module_Info" (BYVAL a%, BWAL
b%, SEG c&, SEG d%, SEG e%)
DECLARE FUNCTION SCXI.Load.Config% CDECL ALIAS "_SCXI_Load_Config" (BYVAL a%)
DECLARE FUNCTION SCXI.MuxCtr.Setup% CDECL ALIAS "_SCXI_MuxCtr_Setup" (BYVAL a%, BWAL b%,
B WAL c%, BYVAL d%)
DECLARE FUNCTION SCXI.Reset% CDECL ALIAS "_SCXI_Reset" (BWAL a%, BYVAL b%)
DECLARE FUNCTION SCXI.SCAN.Setup% CDECL ALIAS "_qb_SCXIScanSetup" (BYVAL a%, BWAL b%,
c%(), d%(), e%(), B WAL f%, B WAL g%)
DECLARE FUNCTION SCXI.Set.Config% CDECL ALIAS "_qb_SCXISetConfig" (BYVAL a%, BYVAL b%,
B WAL c%, BWAL d%, BWAL e%, BWAL f%, g&(), h%(), i%())
DECLARE FUNCTION SCXI.Set.Module.Gain% CDECL ALIAS "_SCXI_Set_Module_Gain" (BYVAL a%, BYVAL
b%, B WAL c%)
DECLARE FUNCTION SCXI.Single.Chan.Setup% CDECL ALIAS "_SCXI_Single_Chan_Setup" (B WAL a%,
B WAL b%, B WAL c%, B WAL d%)
DECLARE FUNCTION SCXI.Track.Hold.Control% CDECL ALIAS "_SCXI_Track_Hold_Control" (BWAL a%,
BWAL b%, BWAL c%, BWAL d%)
DECLARE FUNCTION SCXI.Track.Hold.Setup% CDECL ALIAS "_SCXI_Track_Hold_Setup" (BYVAL a%,
BYVAL b%, BYVAL c%, BYVAL d%, BYVAL e%, BYVAL f%, BYVAL g%)
DECLARE FUNCTION Set.DAQ.Mode% CDECL ALIAS "_Set_DAQ_Mode" (BYVAL a%, BWAL b%)
DECLARE FUNCTION Timeout.Config% CDECL ALIAS "_Timeout_Config" (BWAL a%, BYVAL b&)
DECLARE FUNCTION Trigger.Window.Config% CDECL ALIAS "_Trigger_Window_Config" (BWAL a%,
BWAL b%, BYVAL c%)
DECLARE FUNCTION WFM.Chan.Control% CDECL ALIAS "_WFM_Chan_Control" (BWAL a%, BYVAL b%,
B WAL c%)
DECLARE FUNCTION WFM.Check% CDECL ALIAS "_WFM_Check" (BWAL a%, BYVAL b%, SEG c%, SEG
d&, SEG e&)
DECLARE FUNCTION' WFM.ClockRate% CDECL ALIAS "_WFM_ClockRate" (BYVAL a%, BWAL b%, BYVAL
c%, B WAL d%, B WAL e&, B WAL f%)
DECLARE FUNCTION WFM.DB.Config% CDECL ALIAS "_qb_WFM_DB_Config" (BYVAL a%, BWAL b%,
c%(), BWAL d%, BWAL e%, BWAL f%)
DECLARE FUNCTION WFM.DB.HalfReady% CDECL ALIAS "_qb_WFM_DB_HalfReady" (BYVAL a%, BYVAL
b%, c%(), SEG d%)
DECLARE FUNCTION WFM.DB.StrTransfer% CDECL ALIAS "_qb_WFM_DB_StrTransfer" (BYVAL a%,
BYVAL b%, c%(), d$, BYVAL e&)
DECLARE FUNCTION WFM.DB.Transfer% CDECLALIAS "_qb_WFM_DB_Transfer" (a%, b%, c%(), d%(), e&)
DECLARE FUNCTION WFM.from.Disk% CDECL ALIAS "_qb_WFM_from_Disk" (BYVAL a%, BWAL b%,
c%(), d$, BYVAL e&, BYVAL f&, BYVAL g&, BYVAL h#)
DECLARE FUNCTION WFM.Group.Control% CDECL ALIAS "_WFM_Group_Control" (BYVAL a%, BYVAL b%,
BYVAL c%)
DECLARE FUNCTION WFM.Group.Setup% CDECLALIAS "_qb_WFM_Group_Setup" (BWAL a%, BWAL b%,
c%(), B WAL d%)
DECLARE FUNCTION WFM.Load% CDECL ALIAS "_qb_WFM_Load" (a%, b%, c%(), d%(), e&, f&, g%)
DECLARE FUNCTION WFM.Op% CDECL ALIAS "_qb_WFM_Op" (a%, b%, c%(), d%(), e&, f&, g#)
DECLARE FUNCTION WFM.Rate% CDECL ALIAS "_WFM_Rate" (BWAL a##, BYVAL b%, SEG c%, SEG d%)
```

```
DECLARE FUNCTION WFM.Scale% CDECL ALIAS "_qb_WFM_Scale" (BWAL a%, BWAL b%, BWAL c&, BWAL d#, e#(), f%())
DECLARE FUNCTION AI.Config% CDECL ALIAS "_AI_Config" (BWAL a%, BYVAL b%, BWAL c%, BWAL d%)
DECLARE FUNCTION AI.Scale% CDECL ALIAS "-AI-Scale" (BWAL a%, BWAL b%, BYVAL c%, SEG d#)
DECLARE FUNCTION CTR.Clock% CDECL ALIAS "_CTR_Clock" (BWAL a%, BWAL b%, BWAL c%, BWAL d%)
DECLARE FUNCTION CTR.Simul.Stop% CDECL ALIAS "_CTR_Simul_Stop" (BWAL a%, BWAL b%, BWAL c%, BYVAL d%)
DECLARE FUNCTION DAQ.Scale% CDECL ALIAS "_qb_daqscale" (BWAL a%, BWAL b%, BWAL c%, d%(), e#())
DECLARE FUNCTION Get.A2000.Info% CDECL ALIAS "_Get_A2000_Info" (BWAL a%, SEG b%, SEG c%, SEG d%, SEG e%, SEG f%)
DECLARE FUNCTION Get.DIO24.Info% CDECL ALIAS "_Get_DIO24_Info" (BYVAL a%, SEG b%, SEG c%)
DECLARE FUNCTION Get.DIO32F.Info% CDECL ALIAS "_Get_DIO32F_Info" (BWAL a%, SEG b%, SEG c%, SEG d%, SEG e%, SEG f%)
DECLARE FUNCTION Get.LabBrd.Info% CDECL ALIAS "_Get_LabBrd_Info" (BWAL a%, SEG b%, SEG c%, SEG d%, SEG e%)
DECLARE FUNCTION Get.MIO16.Info% CDECL ALIAS "_Get_MIO1 6_Info" (BWAL a%, SEG b%, SEG c%, SEG d%, SEG e%, SEG f%, SEG g%)
DECLARE FUNCTION Lab.SCAN.Check% CDECL ALIAS "_qb_labscancheck" (BWAL a%, SEG b%, SEG c%, d%())
DECLARE FUNCTION Lab.SCAN.Op% CDECL ALIAS " qb_labscanop" (a%, b%, c%, d%(), e%, f#, g%())
DECLARE FUNCTION Lab.SCAN.Start% CDECL ALIAS' qb_labscanstart" (a%, b%, c%, d%(), e%, f%, g%)
DECLARE FUNCTION Lab.SCAN.to.Disk% CDECL ALIAS "_qb_labscantodisk" (BWAL a%, BWAL b%, BWAL c%, d$, BWAL e&, BWAL f#, BWAL g%)
DECLARE FUNCTION MIO16.F5.Calibrate% CDECL ALIAS "_MIO16_F5_Calibrate" (BWAL a%, BWAL b%, BWAL c%, BWAL d%, BWAL e%, BWAL f%, B YVAL g%, BYVAL h#, BWAL i%)
DECLARE FUNCTION MIO 16.F5.Config% CDECL ALIAS "_MIO 16_F5_Config" (B WAL a%, B WAL b%)
DECLARE FUNCTION WF.Check% CDECL ALIAS "_WF_Check" (B WAL a%, B WAL b%, SEG c%, SEG d%, SEG e%)
DECLARE FUNCTION WF.Clear% CDECL ALIAS "_WF_Clear" (BWAL a%)
DECLARE FUNCTION WF.DB.Config% CDECL ALIAS "_WF_DB_Config" (BWAL a%, BYVAL b%, BYVAL c%, BWAL d%)
DECLARE FUNCTION WF.DB.StrTransfer% CDECL ALIAS "_qb_wfdbstrxfr" (BWAL a%, BWAL b%, c$, BWAL d%)
DECLARE FUNCTION WF.DB.Transfer% CDECL ALIAS "_qb_wfdbxfr" (a%, b%, c%(), d%)
DECLARE FUNCTION WF.from.Disk% CDECL ALIAS "_qb_wffromdisk" (BWAL a%, BYVAL b%, c$, BYVAL d%, BWAL e#, BWAL f&, BWAL g&)
DECLARE FUNCTION WF.Load% CDECL ALIAS "_qb_wfload" (a%, b%, c%(), d%, e%)
DECLARE FUNCTION WF.Op% CDECL ALIAS "_qb_wfop" (a%, b%, c%(), d%(), e%, I%, g%, h%, i#)
DECLARE FUNCTION WF.Pause% CDECL ALIAS "_WF_Pause" (BYVAL a%)
DECLARE FUNCTION WF.Rate% CDECL ALIAS "_WF_Rate" (BWAL a#, BYVAL b%, SEG c%, SEG d%)
DECLARE FUNCTION WF.Resume% CDECL ALIAS "_WF_Resume" (B WAL a%)
DECLARE FUNCTION WF.Scale% CDECL ALIAS "_qb_wfscale" (BYVAL a%, BYVAL b%, BYVAL c%, BYVAL d#, e#(), f%())
DECLARE FUNCTION WF.Start% CDECL ALIAS "_WF_Start" (BYVAL a%, BWAL b%, BYVAL c%)
DECLARE FUNCTION WFM.UpdateRate% CDECL ALIAS "_WFM_UpdateRate" (BWAL a%, BWAL b%, BWAL c%, BWAL d&, BWAL e%)

'$INCLUDE: 'AKEY 1.BI'

NULL$ = CHR$(0)
```

```
 'Arrow key definitions
DELETE% = NULL$ + CHR$(83)
HOME$ = NULL$ + CHR$( 7 1)
PGUP$ = NULL$ + CHR$(73): PGDN$ = NULL$ + CHR$(8 1)
LARROW$ = NULL$ + CHR$(75): RARROW$ = NULL$ + CHR$(77)
UPARROW$ = NULL$ + CHR$(72): DNARROW$ = NULL$ + CHR$(80)

CONST LABPC.PLUS.BRD.CODE% = 28
CONST PI = 3.141593
CONST DegtoRad = PI / 180

heap.size& = SETMEM(-2700)          'Set aside memory for buffers

DIM SHARED Vin#( 15)
DIM SHARED Vout#(3)
DIM SHARED Din( 15)
DIM SHARED Dout(3 1)
DIM SHARED CTR&(6)
DIM SHARED Graph!(650, 3)
DIM SHARED Alt!(650, 1)
DIM SHARED HeadImage(3, 3)
DIM SHARED Trails!(4000, 1)
DIM SHARED TrailImage(3, 3)
DIM SHARED f$( 13)

f$(0) = CHR$(0)
f$(1) = CHR$(0) + CHR$(59)
f$(2) = CHR$(0) + CHR$(60)
f$(3) = CHR$(0) + CHR$(61)
f$(4) = CHR$(0) + CHR$(62)
f$(5) = CHR$(0) + CHR$(63)
f$(6) = CHR$(0) + CHR$(64)
f$(7) = CHR$(0) + CHR$(65)
f$(8) = CHR$(0) + CHR$(66)
f$(9) = CHR$(0) + CHR$(67)
f$( 10) = CHR$(0) + CHR$(68)
f$(11) = CHR$(0) + CHR$(133)
f$(12) = CHR$(0) + CHR$(134)
f$(13) = CHR$(0) + CHR$(104)

OPEN "OFFSETS.DAT" FOR INPUT AS #3
 INPUT #3, Zos!
 INPUT #3, Yos!
 INPUT #3, Los!
CLOSE #3

OPEN "Info.dat" FOR INPUT AS #3
 INPUT #3, BName$
 INPUT #3, CName$
 INPUT #3, SName$
 INPUT #3, TName$
CLOSE #3

CLS
PRINT "    DEFAULTS"
```

```
PRINT “BRIDGE:  "; BName$
PRINT “City or County  "; CName$
PRINT “STATE:  "; SName$
PRINT “Bridge Type:  "; TName$
PRINT
PRINT

INPUT “Bridge Name: ", B 1 name$:
INPUT “County or City: ", C 1 name$
INPUT “State:", S 1 name%
INPUT ‘Type of Bridge: ", T 1 name$

IF B 1name$ o "" THEN BName$ = B I name%
IF C lname$ o "" THEN CName$ = C 1 name$
IF S 1 name$ <> "" THEN SName$ = S 1 name$
IF T1name$ <> "" THEN TName$ = T1name$

Tnum = FREEFILE
OPEN "Info.dat" FOR OUTPUT AS Tnum
 PRINT #Tnum, BName$
 PRINT #Tnum, CName$
 PRINT #Tnum, SName$
 PRINT #Tnum, TName$
CLOSE #Tnum

Yaw0# = .4
YawSpan# = 16.666667#
PitchO# = .6
PitchSpan# = 16.8#
BL0# = 15.5#
BLSpan# = .8#
TAngle0# = 0
TAngleSpan# = 16.66667#
'Odd0# = 0
OddSpan# = .9578#
'Sonar0# = 0
SonarSpan# = 2.644# * 9.84
'Laser0# = 0
LaserSpan# = 1#

err.num = Init.DA.Brds( 1, brd.code%)     ‘Initialize board 1
IF (err.num) THEN
 LOCATE 1, 1: PRINT “Error from Init.DA.Brds #1 is "; err.num%
 PRINT ‘Terminating the program”
 SLEEP 2
 STOP
END IF

err.num = Init.DA.Brds(2, brd.code%)     ‘Initialize board 2
IF (err.num) THEN
 LOCATE I, 1: PRINT “Error from Init.DA.Brds #2 is "; err.num%
 PRINT “Terminating the program”
 SLEEP 2
 STOP
END IF
```

```
            'board 1
err.num = AI.Configure( 1, 0, 1, 5, 0, 0)   'setup +/- 5 volt input
Err.Exit "AI.Configure", err.num
err.num = AO.Config( 1, 0, 1, 0, 0)         'set chan 0 out to 0- 10 volt range
Err.Exit "AO1.Config", err.num
err.num = AO.Config( I, 1, 1 , 0, 0)        'set chan 1 out to 0- 10 volt range
Err.Exit "AO2.Config", err.num
err.num = DIG.Prt.Config( 1 , 1 , 0, 1)     'configure port b as output
Err.Exit "DIG.Prt1.Config", err.num
err.num = DIG.Prt.Config( 1, 2, 0, 1)       'configure port c as output
Err.Exit "DIG.Prt2.Config", err.num
err.num = ICTR.Setup( 1, 1 , 0, 0, 1)       'configure Counter 1 HERE
Err.Exit "ICTR.Setup 1", err.num
err.num = ICTR.Read( 1 , 0, count 1)
err.num = ICTR.Setup( 1, 2, 0, 0, 1)        'configure Counter 2
Err.Exit "ICTR.Setup2", err.num

            board 2
err.num = AI.Configure(2, 0, 1, 10, 1, 0)   'setup O-10 volt input
Err.Exit "AI.Configure", err.num
err.num = AO.Config(2, 0, 1, 0, 0)          'set chan 0 out to O-IO volt range
Err.Exit"AO 1.Config", err.num
err.num = AO.Config(2, 1, 1 , 0, 0)         'set chan 1 out to 0- IO volt range
Err.Exit "AO2.Config", err.num
err.num = DIG.Prt.Config(2, 1, 0, 1)        'configure port b as output
Err.Exit "DIG.Prt1.Config", err.num
err.num = DIG.Prt.Config(2, 2, 0, 1)        'configure port c as output
Err.Exit "DIG.Prt2.Config", err.num
err.num = ICTR.Setup(2, 1, 0, 0, 1)         'configure Counter I
Err.Exit "ICTR.Setup0", err.num
err.num = ICTR.Setup(2, 2, 0, 0, 1)         'configure Counter 2
Err.Exit "ICTR.Setup0", err.num

TGLen = 1
TGHite = 1
PHite = 10
Direction$ = "+"
Stream$ = "U"
MenuKey = 1

ON ERROR GOTO HANDLE
COUNTRESET 1
COUNTRESET 2
  SCREEN 12, 3
  WIDTH 80, 60
  'PARAMETERS
INIT

RESTART:
 DO
   READINPUTS
   CALL CALCULATE(Station!, Elevation, BGCount)
   IF Key3 OR Key9 THEN CALL BRIDGEGRAPH(BGCount)
   CALL PILEGRAPH( PilePass, PCount)
   Dout(3 1) = 0
```

```
  WRITEOUTPUTS
  Key3 = 0
  KeyPress$ = INKEY $
  SELECT CASE KeyPress$
   CASE LARROW$
    IF TGLen > 1 THEN TGLen = TGLen - 1
    CALL   BRIDGEGRAPH(BGCount)
   CASE  RARROW$
    IF TGLen < 12 THEN TGLen = TGLen + 1
    CALL BRIDGEGRAPH(BGCount)
   CASE UPARROW$
    IF TGHite < 5 THEN TGHite = TGHite + 1
    CALL BRIDGEGRAPH(BGCount)
   CASE DNARROW$
    IF TGHite > 1 THEN TGHite = TGHite - 1
    CALL BRIDGEGRAPH(BGCount)
   CASE PGUP$
    IF PHite < 10 THEN PHite = PHite + 1
    PilePass = 0
    PCount = 0
    CALL  PILEGRAPH(PilePass, PCount)
   CASE PGDN$
    IF PHite > 4 THEN PHite = PHite - 1
    PilePass = 0
    PCount = 0
    CALL  PILEGRAPH(PilePass, PCount)
   CASE f$( 13)
    SCRNPRNT
   CASE ELSE
    IF MenuKey = 1 THEN MENU1 FUNCTIONS KeyPress$ ELSE MENU2FUNCTIONS KeyPress$
  END SELECT
  CALL  SCREENWRITE(Station!, Elevation)
 LOOP
END

HANDLE:
 ErrorNumber = ERR
 SELECT CASE ErrorNumber
  CASE 7
   LOCATE 58, 1: PRINT "NOT ENOUGH MEMORY"
   SLEEP 2
   RESUME
  CASE 53           'File not found - ok to open
   RESUME NEXT
  CASE 55           'File exists - Overwrite
   RESUME  NEXT
  CASE 61
   LOCATE 57, 20: PRINT "** ERROR - Disk Is Full"
   LOCATE 58, 20: PRINT "Insert New disk & Press C to continue, or A to Abort: "
   DO
    Char$ = UCASE$( INPUT$( 1))
    IF Char$ = "C" THEN
     RESUME         ' Resume where you left off
    ELSEIF Char$ = "A" THEN
     LOCATE 57, 20: PRINT "                              "
```

```
      LOCATE 58.20: PRINT "
      CLOSE
      Key1 =0
      Key2 = 0
      ErrorNumber = 0
      RESUME NEXT
     END IF
   LOOP
  CASE 64            'Invalid filename
   RESUME NEXT
  CASE 71            ' Disk not ready
   LOCATE 57, 20: PRINT "** ERROR - Disk drive not ready'
   LOCATE 58, 20: PRINT "Insert disk & Press C to continue, or R to restart: "
   Do
    Char$ = UCASE$( INPUT$( 1))
    IF Char$ = "C" THEN
      RESUME
    ELSEIF Char$ = "R" THEN
      LOCATE 57, 20: PRINT "                              "
      LOCATE 58, 20: PRINT "                              "
      RESUME RESTART
    END IF
   LOOP
  CASE 76
   RESUME NEXT
  CASE ELSE
   LOCATE 57.20: PRINT "ERROR NUMBER "; ErrorNumber
   SLEEP 2
 END SELECT
RESUME NEXT

SUB BRIDGEGRAPH (BGCount)

STATIC TravelTemp!, ElevTemp#
'Bridge Graph
 LINE (80, 59)-(630, 180), 11, BF
 LINE (80, 59)-(630, 180), 8, B
 COLOR 7
 LINE (80, 100)-(630, 100)

 VIEW SCREEN (80, 60)-(630, 180)
 WINDOW
'horizontal 0 line
 COLOR 6
 LOCATE 10, 2: PRINT "ELEV"
 COLOR 13
 LOCATE 8, 7: PRINT 10 * TGHite
 LOCATE 13, 8: PRINT "0-"
 'COLOR 13
 LOCATE 18, 2: PRINT "DRY'
 LOCATE 19, 2: PRINT "ECHO"
 'COLOR 13
 LOCATE 23, 7: PRINT USING "###"; 20 * TGHite
 LOCATE 24,10: PRINT 0
 LOCATE 24, 20: PRINT 100 * TGLen
```

```
 LOCATE 24, 33: PRINT 200 * TGLen
 LOCATE 24, 45: PRINT 300 * TGLen
 LOCATE 24, 58: PRINT 400 * TGLen
 LOCATE 24, 7 1: PRINT 500 * TGLen
'Vertical lines
 COLOR 7
 LINE (180, 60)-(180, 180)
 LINE (280, 60)-(280, 180)
 LINE (380, 60)-(380, 180)
 LINE (480, 60)-(480, 180)
 LINE (580, 60)-(580, 180)

 VIEW (80, 60)-(630, 180)
 WINDOW (0, 10 * TGHite)-(550 * TGLen, -20 * TGHite)
 COLOR 6
'Draw Graph
 FOR X = 0 TO BGCount
  LINE (Graph!(X, 0), Graph!(X, 1))-(Graph!(X, 2), Graph!(X, 3))
 NEXT X
'Draw Contaq Lines
 COLOR 13
 FOR X = 0 TO BGCount
  LINE (Graph!(X, 0), Alt!(X, 0))-(Graph!(X, 2), Alt!(X, 1))
 NEXT X

 VIEW
 WINDOW
 Key9 = 0
END SUB

SUB BRIDGELOG
 '         Xa   Ya   BL   Sa   Od   Ia   La   Xf   Yf   Zf   Hb   Xb   Yb   Zb   Lb   Zt   Xtt   Xbt   Ed   Ls
Eb
 PRINT #1, USING "+##.# +##.# ##.# +##.# ####.# +##.## +##.# +##.# +##.# +##.# +##.# ####.# +##.# +##.#
+##.# +###.# #####.# #####.# ####.# ####.# ####.#"; Xa!; Ya!; BL!; Sa!; Od!; Ia!; La!; Xf!; Yf!; Zf?; Hb!; Xb!; Yb!;
Zb!; Lb!; Zt!; Xtt!; Xbt!; _
Ed!; Ls!; Eb!
END SUB

SUB CALCULATE (Station!, Elevation, BGCount)
STATIC ReadAngle, Travel1 Temp!, Travel2Temp!, AngleAcum#, LaserTemp#
STATIC XDispTemp!, YDispTemp!
 En = ErrorNumber
 IF Key3 = 1 THEN
  ReadAngle = 0
  Travel 1 Temp! = 0
  Travel2Temp! = 0
  AngleAcum# = 0
  LaserTemp# = 0
  Hb! = 0
  ERASE Graph!
  ERASE Alt!
 END IF
'Read Laser from Corn Port
 PortFile = FREEFILE
```

```
'OPEN "COM2:9600,N,8,1,CS0,DS0" FOR RANDOM AS #PortFile
 'PRINT #PortFile, "SO 1920"
 'PRINT #PortFile, "Z"
 'Print #PortFile

' PRINT #PortFile, CHR$( 17)
 'Print #PortFile, "O
' INPUT #PortFile, Data%
' INPUT #PortFile, Data$
' PRINT #PortFile, CHR$( 19)
' La! = ((VAL(LEFT$(Data$, 8)) * .35 / 2.54) / 12) + 1
'CLOSE #PortFile
'Calculate Analog Input Scaling
Ia! = (-Vin#(6) + TAngle0#) * TAngleSpan#
Sa! = -Vin#( 5) * SonarSpan#
'La! = ABS(((Vin#(7) - 4.998) * 12) - 3.45) - 5.5
Lb! = La! - Los!
Ls! = Elevation - Lb!
'Calulate Analog Outputs
scaleout = 8
IF Din(O) = 0 THEN
 IF scaleout * (ABS(Vin#(0)) + ABS(Vin#( I)) + ABS(Vin#(2))) > 9.99 THEN
  Vout#( 1) = 9.99                            'Valve
 ELSE
  Vout#( 1) = scaleout * (ABS(Vin#(0) + Vin#( I) + Vin#(2) + .l))
 END IF
 Vout#(0) = 2.5                              'Dart
ELSE
 IF scaleout * (ABS(Vin#(O)) + ABS(Vin#( 1))) > 9.9 THEN
  Vout#( 1) = 9.99                            'Valve
 ELSE
  Vout#( 1) = scaleout * (ABS(Vin#(0)) + ABS(Vin#( 1)))         'Valve
 END IF

'WINCH OUTPUT BASE ON BOOM LENGTHS
 IF Din(4) = 0 THEN          'Din(4) is Boom Stop Switch case is off
  IF Din(7) = 0 THEN         'Din(7) is auto z switch case is off
   SELECT CASE BL!
    CASE 59 TO 59.9
     SELECT CASE (Vin#(2) * 1.25) + 2.5
      CASE IS >= 2.5
       Vout#(0) = (Vin#(2) * 1.25) + 2.5            'Dart
      CASE IS < 2.5
       Vout#(0) = ((Vin#(2) * 1.25) / 2) + 2.5
     END SELECT
    CASE IS >= 60
     IF ((Vin#(2) * 1.25) + 2.5) <= 2.5 THEN              'Dart
      Vout#(0) = 2.5
     ELSE
      Vout#(0) = (Vin#(2) * 1.25) + 2.5            'Dart
     END IF
    CASE ELSE
     Vout#(0) = (Vin#(2) * 1.25) + 2.5             'Dart
   END SELECT
  ELSE
```

```
'Auto Z Switch On - insert auto z case here
      IF Din(5) = 1 THEN
       Vout#(0) = 4
      ELSEIF Din(6) = 1 THEN
       Vout#(0) = . I
      ELSE Vout#(0) = 2.5
      END IF
     END IF
  ELSE                  'Din(4) - Boom Stop = 1
     SELECT CASE (Vin#(2) * 1.25) + 2.5
      CASE IS < 0
       Vout#(0) = 0
      CASE IS >= 2.5
       Vout#(0) = 2.5
      CASE IS < 2.5
       Vout#(0) = (Vin#(2) * 1.25) + 2.5                    'Dart
     END SELECT
   END IF
  END IF

BOOM  COUNTER  OUTPUT
  IF BL! > 0 AND BL! < 1000 THEN
   Vout#(2) = (BL! * .01005) + .022 - .0161
  END IF

'Calculate Digital Outputs Based on Digital inputs and analog inputs
  IF Din(O) = 1 THEN      'operate mode
   Dout(4) = 0
   Dout(5) = 0
   Dout(8) = 0
   Dout(9) = 0
   Dout(10) = 0
   Dout(11) = 0

   IF Vin#(0) < -.2 THEN
    Dout(0) = 0
    Dout(1) = 1
   ELSEIF Vin#(0) > .2 THEN
    Dout(0) = 1
    Dout(1) =0
   ELSE
    Dout(0) = 0
    Dout(1) = 0
   END IF

   IF Vin#( 1) < -.2 THEN
    Dout(2) = 1
    Dout(3) = 0
   ELSEIF Vin#( 1) > .2 THEN
    Dout(2) = 0
    Dout(3) = 1
   ELSE
    Dout(2) = 0
    Dout(3) = 0
   END IF
```

```
    IF Vin#(2) < -.2 THEN
     Dout(6) = 1
     Dout(7) = 0
    ELSEIF Vin#(2) > .2 THEN
     Dout(6) = 0
     Dout(7) = 1
    ELSE
     Dout(6) = 0
     Dout(7) = 0
    END IF

  ELSE
   Dout(0) = 0
   Dout(1) =0
   Dout(2) = 0
   Dout(3) = 0
   Dout(6) = 0
   Dout(7) = 0

   IF Vin#(2) < -.2 THEN
    Dout(4) = 1
    Dout(5) = 0
   ELSEIF Vin#(2) > .2 THEN
    Dout(4) = 0
    Dout(5) = 1
   ELSE
    Dout(4) = 0
    Dout(5) = 0
   END IF

   IF Vin#(1) < -.2 THEN
    Dout(8) = 0
    Dout(9) = 1
   ELSEIF Vin#( 1) > .2 THEN
    Dout(8) = 1
    Dout(9) = 0
   ELSE
    Dout(8) = 0
    Dout(9) = 0
   END IF

   IF Vin#(0) < -.2 THEN
    Dout(10) = 1
    Dout(ll)=O
   ELSEIF Vin#(0) > .2 THEN
    Dout( 10) = 0
    Dout(11) =1
   ELSE
    Dout( 10) = 0
    Dout(ll)=O
   END IF
END IF

IF Din(3) THEN Dout( 16) = 1 ELSE Dout( 16) = 0
IF Din(4) THEN COUNTRESET 2
```

```
'Other calculations

 CTR&(0) = CTR&(2) - CTR&(1)
 CTR&(3) = CTR&(5) - CTR&(4)
 Od! = (((CTR&(0) / 10) - TaOffset!) * OddSpan#)
 BL! = ((CTR&(3) - BoomOffset!) / 10 * BLSpan#) + BL0#
 Ya! = -((Vin#(3) * PitchSpan#) - Pitch0#)
 Xa! = -((Vin#(4) * YawSpan#) - Yaw0#)
 PangRad# = Ya! * DegtoRad
 YangRad# = Xa! * DegtoRad
 Zf! = (COS(ATN(SQR((TAN(PangRad#) ^ 2) + (TAN(YangRad#) ^ 2)))) * BL!) ' * 1.05
 Xf! = TAN(YangRad#) * Zf!
 Yf! = TAN(PangRad#) * Zf?
 Xb! = Xf! + Od!
 Yb! = Yf! - Yos!
 Zb! = Zf! - Zos!
 Zt! = Zb! + Sa!
 Eb! = Elevation - Zt!
 Xtt! = Station! + Od!
 Xbt! = Xtt! + Xf!
 Delta 1 Od! = Od! - Travel 1 Temp!
 Delta2Od! = Od! - Travel2Temp!

 IF Delta1 Od! >= .1 THEN
  ReadAngle = ReadAngle + 1
  AngleAcum# = AngleAcum# + Ia!
  Travel 1 Temp! = Od!
 END IF
 IF Delta2Od! >= 1 THEN
  TrailerAvg# = AngleAcum# / ReadAngle
  DeltaElev# = Delta2Od! * TAN(.017453292# * TrailerAvg#)
  Elev! = Hb!
  Hb! = Hb! + DeltaElev#
  Ed! = Hb! + Elevation
  IF Key1 = 1 THEN
   BRIDGELOG
   BGCount = BGCount + 1
   Graph!(BGCount, 0) = Travel2Temp!
   Graph!(BGCount, 2) = Od!
   Graph!(BGCount, 1) = Elev!
   Graph!(BGCount, 3) = Hb!
   Alt!(BGCount, 0) = LaserTemp#
   Alt!(BGCount, 1) = -La! + Hb!
   Key9 = 1
  END IF

  IF Key2 THEN
   ERASE Trails!
   PILELOG
  END IF

  Travel2Temp! = Od!
  ReadAngle = 0
  AngleAcum# = 0
  LaserTemp# = -La! + Hb!
```

```
 END IF
   TotalDisp! = SQR((Xf! ^ 2) + (Yf! ^ 2)) - SQR((XDispTemp! ^ 2) + (YDispTemp! ^ 2))
 IF ABS(TotalDisp!) >= .5 THEN
  IF Key2 THEN
   Trails!(TrailNum, 0) = Xf!
   Trails!(TrailNum, 1) = Yf!
   IF TrailNum < 4000 THEN TrailNum = TrailNum + 1
   XDispTemp! = Xf!
   YDispTemp! = Yf!
   PRINT #2,USING " +##.#    +##.#    ##.#    +#####.#    +##.#    +##.#"; Ya!; Xa!; BL!; -Xf! +
Station! + Od!; Yf! + Offset!; Zt!
  END IF
 END IF
END SUB

SUB COUNTRESET (CountKey!)

 IF CountKey! = 1 THEN
  err.num = ICTR.Setup( 1, 1, 0, 0, 1)      'configure Counter 1
  Err.Exit "ICTR.Setup11", err.num
  err.num = ICTR.Setup( 1, 2, 0, 0, 1)      'configure Counter 2
  Err.Exit "ICTR.Setup12", err.num
  Dout(15) = 1
  WRITEOUTPUTS
  Dout(15) = 0
  WRITEOUTPUTS
  Dout( 15) = 1
  WRITEOUTPUTS
  Dout( 15) = 0
  WRITEOUTPUTS
  READINPUTS
  CALL CALCULATE(Station!, Elevation, BGCount)
  IF (CTR&(0) / 10) <> 0 THEN
   TaOffset! = (CTR&(0) / 10)
  ELSE
   TaOffset! = 0
  END IF
  Key3 = 1
 END IF

 IF CountKey! = 2 THEN
   err.num = ICTR.Setup(2, 1, 0, 0, 1)      'configure Counter 1
   Err.Exit "ICTR.Setup21", err.num
   err.num = ICTR.Setup(2, 2, 0, 0, 1)      'configure Counter 2
   Err.Exit "ICTR.Setup22", err.num
   Dout(31) = 1
   WRITEOUTPUTS
   Dout(31) = 0
   WRITEOUTPUTS
   Dout(31) = 1
   WRITEOUTPUTS
   Dout(31) =0
   WRITEOUTPUTS
   'CTR&(3) = CTR&(5) - CTR&(4)
   'BL! = ((CTR&(3) / 10 * BLSpan#)) + BL0#
```

```
    BoomOffset! = CTR&(3)

    'CALL CALCULATE(Station!, Elevation, BGCount)
    'IF (CTR&(3) / 10) <> 0 THEN
    ' BoomOffset! = (CTR&(3) / 10)
    'ELSE
    ' BoomOffset! = 0
    'END IF
  END IF
END SUB

SUB DIAGNOSTICS
CLS
DO WHILE KeyPress$ <> f$(6)
 KeyPress$ = INKEY$
 IF KeyPress$ = CHR$(27) THEN EXIT DO
 IF KeyPress$ = f$( 11) THEN COUNTRESET 2
 IF KeyPress$ = f$( 10) THEN COUNTRESET 1
 IF KeyPress$ = f$( 13) THEN SCRNPRNT

 READINPUTS
 CALL CALCULATE(Station!, Elevation, BGCount)
 WRITEOUTPUTS

 COLOR 12
 LOCATE 3, 30: PRINT "Analog Inputs"
 LOCATE 14, 30: PRINT "Analog Outputs"
 LOCATE 19, 30: PRINT "Counters"
 LOCATE 25, 30: PRINT "Digital Inputs"
 LOCATE 36, 30: PRINT "Digital Outputs"

 LOCATE 4, 5: PRINT "+/- 5 Volt Inputs"
 LOCATE 4, 46: PRINT "0-10 Volt Inputs"
'Assignments Board 1
 COLOR 13
 LOCATE 5, 14: PRINT 'Right Joystick Fore-Aft"
 LOCATE 6, 14: PRINT "Right Joystick Left-Right"
 LOCATE 7, 14: PRINT "Left Joystick Fore-Aft"
 LOCATE 8, 14: PRINT "Pitch Inclinometer"
 LOCATE 9, 14: PRINT "Yaw Inclinometer"
 LOCATE 10, 14: PRINT "DataSonics Sonar"
 LOCATE 11, 14: PRINT 'Trailer Inclinometer"
 LOCATE 12, 14: PRINT "Contaq Echo Sounder"
 LOCATE 16, 14: PRINT "Proportional Valve"
 LOCATE 17, 14: PRINT "Winch Up-Down Speed"
 LOCATE 2 1, 14: PRINT "Total Travel"
 LOCATE 22, 14: PRINT "Trailer Reverse Travel"
 LOCATE 23, 14: PRINT "Trailer Forward Travel"
 LOCATE 27, 14: PRINT "Operate/Deploy Switch"
 LOCATE 28, 14: PRINT "Auto Tilt Switch"
 LOCATE 29, 14: PRINT "Spare Din 1"
 LOCATE 30, 14: PRINT "Head Power Switch"
 LOCATE 3 1, 14: PRINT "Boom Stop Switch"
 LOCATE 32, 14: PRINT "Top Float Switch"
 LOCATE 33, 14: PRINT "Bottom Float Switch"
```

```
LOCATE 34, 14: PRINT "Auto-2 Switch"
LOCATE 38, 14: PRINT "Pitch Out"
LOCATE 39, 14: PRINT "Pitch In"
LOCATE 40, 14: PRINT "Yaw Right"
LOCATE 41, 14: PRINT "Yaw Left"
LOCATE 42, 14: PRINT "Rotate Down"
LOCATE 43, 14: PRINT "Rotate Up"
LOCATE 44, 14: PRINT "Winch Down"
LOCATE 45, 14: PRINT "Winch Up"
LOCATE 46, 14: PRINT "Swing Left"
LOCATE 47, 14: PRINT "Swing Right"
LOCATE 48, 14: PRINT "Elbow In"
LOCATE 49, 14: PRINT "Elbow Out"
LOCATE 50, 14: PRINT "Head Power Out"
LOCATE 5 1, 14: PRINT "Auto Tilt Out"
LOCATE 52, 14: PRINT "Spare Dout 1"
LOCATE 53, 14: PRINT 'Boom Ctr Reset"

COLOR 15: LGCATE 1, 10: PRINT "Board # 1 Functions"
'Analog Inputs Board 1
FOR X = 0 TO 7
  LOCATE X + 5, 1: PRINT USING "\ \## ##.###"; "Vin"; X; Vin#(X)
  'PRINT USING "##.###"; Vin#(X)
NEXT X

'Analog outputs
LOCATE 16, 1: PRINT USING "\   \##.###"; "Vout 1 "; Vout#( 1)
LOCATE 17, 1: PRINT USING "\   \##.###"; "Vout 0 "; Vout#(0)

'Counters
LOCATE 21, 1: PRINT USING "\   \+#####"; "Ctr 0"; CTR&(0)
LOCATE 22, 1: PRINT USING "\   \#####"; "Ctr 1"; CTR&( 1)
LOCATE 23, 1: PRINT USING "\   \#####"; "Ctr 2"; CTR&(2)

'Digital Inputs
FOR X = 0 TO 7
  LOCATE X + 27, 1: PRINT "Din"; X; "   "; Din(X)
NEXT X

'Digital Outputs
FORX=OTO 15
  LOCATE X + 38, 1: PRINT USING "\ \ ##   #"; "Dout"; X; Dout(X)
NEXT X

'Assignments Board 2
COLOR 13
LGCATE 5, 55: PRINT "Spare Ain 1"
LOCATE 6, 55: PRINT "Spare Ain 2"
LOCATE 7, 55: PRINT "Spare Ain 3"
LOCATE 8, 55: PRINT "Spare Ain 4"
LOCATE 9, 55: PRINT "Spare Ain 5"
LOCATE 10, 55: PRINT "Spare Ain 6"
LOCATE 11, 55: PRINT "Spare Ain 7"
LOCATE 12, 55: PRINT "Spare Ain 8"
LOCATE 16, 55: PRINT 'Boom Count Out"
```

```
 LOCATE 17, 55: PRINT "Spare Aout 1"
 LOCATE 21, 55: PRINT 'Total Boom Length"
 LOCATE 22, 55: PRINT "Boom Down Counter"
 LOCATE 23, 55: PRINT "Boom Up Counter"
 LOCATE 27, 55: PRINT "Spare Din 2"
 LOCATE 28, 55: PRINT "Spare Din 3"
 LOCATE 29, 55: PRINT "Spare Din 4"
 LOCATE 30, 55: PRINT "Spare Din 5"
 LOCATE 3 1, 55: PRINT "Spare Din 6"
 LOCATE 32, 55: PRINT "Spare Din 7"
 LOCATE 33, 55: PRINT "Spare Din 8"
 LOCATE 34, 55: PRINT "Spare Din 9"
 LOCATE 38, 55: PRINT "Spare Dout 2"
 LOCATE 39, 55: PRINT "Spare Dout 3"
 LOCATE 40, 55: PRINT "Spare Dout 4"
 LOCATE 41, 55: PRINT "Spare Dout 5"
 LOCATE 42, 55: PRINT "Spare Dout 6"
 LOCATE 43, 55: PRINT "Spare Dout 7"
 LOCATE 44, 55: PRINT "Spare Dout 8"
 LOCATE 45, 55: PRINT "Spare Dout 9"
 LOCATE 46, 55: PRINT "Spare Dout 10"
 LOCATE 47, 55: PRINT "Spare Dout 11"
 LOCATE 48, 55: PRINT "Spare Dout 12"
 LOCATE 49, 55: PRINT "Spare Dout 13"
 LOCATE 50, 55: PRINT "Spare Dout 14"
 LOCATE 5 1, 55: PRINT "Spare Dout 15"
 LOCATE 52, 55: PRINT "Spare Dout 16"
 LOCATE 53, 55: PRINT "Trailer Ctr Reset"
 COLOR 14
 LOCATE 1, 45: PRINT "Board # 2 Functions"
'Analog Inputs Board 2
 FOR X = 8 TO 15
  LOCATE X - 3, 41: PRINT USING "\ \ ## ##.###"; "Vin"; X; Vin#(X)
 NEXT X
'Analog Outputs Board 2
 LOCATE 16, 4 1: PRINT USING '1   \ ##.###"; "Vout 2 "; Vout#(2)
 LOCATE 17, 41: PRINT USING "\   \ ##.###"; "Vout 3 "; Vout#(3)
'Counters Board 2
 LOCATE 21, 41: PRINT USING "\   \+#####"; "Ctr 3"; -CTR&(3)
 LOCATE 22, 41: PRINT USING '1   \#####"; "Ctr 4"; CTR&(4)
 LOCATE 23, 41: PRINT USING "\   \#####"; "Ctr 5"; CTR&(5)
'Digital Inputs
 FOR X = 8 TO 15
  LOCATE X + 19, 41: PRINT USING "\ \##   #"; "Din"; X; Din(X)
 NEXT X
'Digital Outputs Board 2
 FOR X = 16 TO 31
  LOCATE X + 22, 41: PRINT "Dout"; X; " "; Dout(X)
 NEXT X
LOOP
INIT
END SUB

SUB Err.Exit (procname$, err.num)
```

```
IF err.num <> 0 THEN
 'LOCATE 58, 1
 'COLOR 15
 'PRINT "A/D Board Error - Error in "; procname$; " is "; err.num
END IF

END SUB

SUB FILEOPEN (EXT$, Valid%, ErrorNumber)

 LOCATE 57, 20
 PRINT "                    "
 COLOR 15
 LOCATE 57, 20
 INPUT ; "ENTER LOG FILENAME: ", File%
 FileName$ = UCASE$(File$) + "." + EXT$
 ON ERROR GOT0 HANDLE          'Checks if file exists & is valid filename
 IF EXT$ = "BRI" THEN
  OPEN FileName$ FOR INPUT AS # 1
  BridgeFile% = FileName$
 END IF
 IF EXT$ = "PIL" THEN
  OPEN FileName$ FOR INPUT AS #2
  PileFile$ = FileName$
 END IF

 SELECT CASE ErrorNumber
  CASE 0                  'File already exits
   LOCATE 57, 20: PRINT 'File Already Exists              "
   LOCATE 58, 20: PRINT "Over Write Yes or No?"
   Char$ = UCASE$( INPUT$( 1))
   LOCATE 57, 20: PRINT "                           "
   LOCATE 58, 20: PRINT "                           "
   IF UCASE$(Char$) = 'Y" THEN
    IF EXT$ = "BRI" THEN
     CLOSE #1
     OPEN FileName$ FOR OUTPUT AS # 1
    ELSEIF EXT$ = "PIL" THEN
     CLOSE #2
     OPEN FileName$ FOR OUTPUT AS #2
    END IF
    Valid% = 1
   END IF
   IF UCASE$(Char$) = "N" THEN
    CLOSE #1
    CALL FILEOPEN(EXT$, Valid%, ErrorNumber)
   END IF
  CASE 53                 'File does not exist and filename valid
   IF EXT$ = "BRI" THEN
    OPEN FileName$ FOR OUTPUT AS # 1
   ELSEIF EXT$ = "PIL" THEN
    OPEN FileName$ FOR OUTPUT AS #2
   END IF
   Valid% = 1
   LOCATE 57, 20: PRINT "
```

```
    LOCATE 58, 20: PRINT "
   CASE 55
    IF EXT$ = "BRI" THEN CLOSE #1
    IF EXT$ = "PIL" THEN CLOSE #2
    ErrorNumber = 0
   CASE 64                  'Invalid filename
    LOCATE 57, 20
    PRINT "You Must Enter A Valid FileName"
    LOCATE 58, 20: PRINT "Press any key to continue or ESCAPE to quit"
    DO WHILE keyp$ = ""
     keyp$ = INKEY$
     IF keyp$ = CHR$(27) THEN
      LOCATE 57, 20: PRINT "                                    "
      LOCATE 58, 20: PRINT "                                         "
      ErrorNumber = 0
      EXIT SUB
     END IF
    LOOP
    LOCATE 57, 20: PRINT "                                    "
    LOCATE 58, 20: PRINT "                                  "
    ErrorNumber = 0
    CALL FILEOPEN(EXT$, Valid%, ErrorNumber)
   CASE 76
    LOCATE 57, 20: PRINT "Path Not Found - Hit Any Key to Continue"
    DO WHILE keyp$ = ""
     keyp$ = INKEY$
    LOOP
    LOCATE 57.20: PRINT "                               "
    ErrorNumber = 0
    CALL FILEOPEN(EXT$, Valid%, ErrorNumber)
  END SELECT
  LOCATE 57, 20
  PRINT "                          "
END SUB

SUB INIT
  PilePass = 0
  PCount = 0
  CLOSE
  CLS

  SCREEN 12, 3                 'MAKEHEAD IMAGE
  WIDTH 80, 60
  CIRCLE (3, 3), 3, 14
  GET (0, 0)-(6, 6), HeadImage
  LINE (10, 10)-(16, 16), 15, BF
  GET ( 10, 1 0)-( 13, 13), TrailImage
  CLS
  SCREEN 12, 3
  WIDTH 80, 60
'Outside border
  LINE (O, O)-(639,445). 1, B
  LINE (3, 3)-(636, 442), 1, B
  PAINT (1, 1), 1, 1
'JS bottom border
```

```
  LINE (0, 50)-(639, 53), 1, BF
'T border
  LINE (0, 230)-(639, 233), 1, BF
  LINE (200, 232)-(203, 444), 1, BF 'vertical line
'Bridge Section
 'COLOR 12: LOCATE 27, 2: PRINT "TRAVERSE MODE"
 COLOR 14:
 LOCATE 26, 2: PRINT "DIRECTION: "
 LOCATE 28, 2: PRINT "SIDE: "
 LOCATE 26, 20: PRINT "STATION = "
 LOCATE 28, 20: PRINT "ALTITUDE = "
 LOCATE 26, 40: PRINT "TRAVEL = "
 LOCATE 28, 40: PRINT "ALTIMETER = "
 LOCATE 26, 6 1: PRINT "TRAILER < "
 LOCATE 28, 6 1: PRINT "ELEVATION= "
'Pile Section
 COLOR 12
 LOCATE 3 1, 28: PRINT "PILE MODE DISPLAY'
 LOCATE 46, 32: PRINT "OFFSETS"
 LOCATE 50, 32: PRINT "DECK TO"
 COLOR 14
 LOCATE 33, 28: PRINT "PITCH <"
 LOCATE 34, 28: PRINT 'YAW <"
 LOCATE 36, 28: PRINT "PITCH Y"
 LOCATE 37, 28: PRINT 'YAW X"
 LOCATE 39, 28: PRINT "YDisp"
 LOCATE 40, 28: PRINT "XDisp"
 LOCATE 42, 28: PRINT "BOOM LEN"
 LOCATE 44, 28: PRINT "SONAR"
 LOCATE 47, 28: PRINT " Yos Zos Los"
 LOCATE 52.28: PRINT "SURFACE"
 LOCATE 54, 28: PRINT "BED"
'Menu Section
 COLOR 14
 LOCATE 33, 2: PRINT "F1 : "
 LOCATE 35 2: PRINT "F2: "
 LOCATE 37: 21 PRINT "F3: "
 LOCATE 39, 2: PRINT "F4: "
 LOCATE 4 1, 2: PRINT "F5: "
 LOCATE 43, 2: PRINT "F6: "
 LOCATE 45, 2: PRINT "F7: "
 LOCATE 47, 2: PRINT "F8: "
 LOCATE 49, 2: PRINT "F9: "
 LOCATE 5 1, 2: PRINT "F1 0:"
 LOCATE 53, 2: PRINT "Fl 1:"
 LOCATE 55, 2: PRINT "F12:"
 IF MenuKey = 1 THEN MENU1
 IF MenuKey = 2 THEN MENU2
'Joystick Section
 COLOR 14
 LOCATE 2, 2: PRINT "PITCH SIGNAL: "; "   Volts"
 LOCATE 2, 32: PRINT "YAW SIGNAL: "; "   Volts"
 LOCATE 2, 59: PRINT "Z SIGNAL: "; "   Volts"
 LOCATE 4, 2: PRINT "HEAD POWER "
 LOCATE 4.32: PRINT "AUTO Z "
```

```
 LOCATE 4, 59: PRINT "AUTO LEVEL "
 LOCATE 6, 2: PRINT "MODE SELECT "
 LOCATE 6, 32: PRINT "SPEED"
 LOCATE 6, 59: PRINT "WINCH"

 CALL SCREENWRITE(Station!, Elevation)
 CALL   BRIDGEGRAPH(BGCount)
 CALL PILEGRAPH(PilePass, PCount)
'Message  area
 COLOR 12
 LOCATE 57, 2: PRINT "Message Area:"
END SUB

SUB MENU1
 COLOR 12
 LOCATE 31, 2: PRINT "    MENU 1       "
 LOCATE 32, 2: PRINT "                 "

 IF Key1 = 0 THEN
   COLOR 10
   LOCATE 33.6: PRINT "START BRIDGE LOG"
 ELSE
   COLOR 12
   LOCATE 33, 6: PRINT "STOP BRIDGE LOG"
 END IF

 IF Key2 = 0 THEN
   COLOR 10
   LOCATE 35, 6: PRINT "START PILE LOG"
 ELSE
   COLOR 12
   LOCATE 35, 6: PRINT "STOP PILE LOG"
 END IF

 COLOR 13
 LOCATE 37, 6: PRINT "EDIT FIELD NOTES "  'F3
 LOCATE 39, 6: PRINT "TRAILS          "  'F4
 LOCATE 41, 6: PRINT "MANUAL LOG      "  'F5
 LOCATE 43, 6: PRINT "DIAGNOSTIC PAGE "  'F6
 LOCATE 45, 6: PRINT "PARAMETERS PAGE "  'F7
 LOCATE 47, 6: PRINT "MENU 2          "  'F8
 LOCATE 49, 6: PRINT "                "  'F9
 LOCATE 5 1, 6: PRINT "TRAILER RESET   "  'F 10
 LOCATE 53, 6: PRINT "BOOM RESET      "  'Fll
 LOCATE 55, 6: PRINT "EXIT            "  'F12

 IF Trail THEN
  LOCATE 39, 14
  COLOR 15
  PRINT "ON "
 ELSE
  LOCATE 39, 14
  COLOR 15
  PRINT "OFF"
 END IF
```

```
END SUB

SUB MENU 1 FUNCTIONS (KeyPress$)
 SELECT CASE KeyPress$
  CASE f$(l)
   IF Keyl = 0 THEN
    EXT$ = "BRI"
    Valid% = 0
    CALL FILEOPEN(EXT$, Valid%, ErrorNumber)
    IF Valid% = 1 THEN
     INPUT "Direction Of Travel + or - ", d$
     SELECT CASE d$
      CASE ""
       Direction$ = "+"
      CASE "+"
       Direction$ = d$
      CASE "-"
       Direction$ = d$
     END SELECT
     LOCATE 58: PRINT "                              "
     LOCATE 58:
     INPUT "Stream Side Up or Down ", S$
     LOCATE 58: PRINT "                              "
     SELECT CASE UCASE$(S$)
      CASE "U"
       Stream$ = "U"
      CASE "D"
       Stream$ = "D"
     END SELECT
     COLOR 4
     LOCATE 33, 6: PRINT "                 "
     LOCATE 33, 6: PRINT "STOP BRIDGE LOG "
     PRINT # 1, "FILENAME: "; BridgeFile$, "DATE: "; DATE$, "TIME: "; TIME$
     PRINT # 1, "BRIDGE: "; BName$, "CO/CITY: "; CName$, "STATE: "; SName$
     PRINT # 1, "US or DS: "; Stream$, "TYPE OF DECK: "; TName$
     PRINT # 1, "OFFSETS: ", "Yos: "; Yos!, "Zos: "; Zos!, "Los: "; Los!
     PRINT # 1, "STARTING STATION: "; Station!, "DIRECTION: "; Direction$, "STARTING DECK
ELEVATION: "; Elevation
     PRINT #1,
     PRINT #1, " Xa "; " Ya "; " BL "; " Sa "; " Od "; " Ia "; " La "; " Xf "; " Yf "; " Zf "; " Hb ";
     PRINT # 1, " Xb "; " Yb "; " Zb "; " Lb "; " Zt "; " Xtt "; " Xbt "; " Ed "; " Ls "; " Eb "
     PRINT #1,
     Keyl = 1
    END IF
   ELSE
    CLOSE # 1
    COLOR 10
    LOCATE 33, 6: PRINT "START BRIDGE LOG"
    Keyl = 0
   END IF
  CASE f$(2)
   IF Key2 = 0 THEN
    EXT$ = "PIL"
    Valid% = 0
    CALL FILEOPEN(EXT$, Valid%, ErrorNumber)
```

```
IF Valid% = 1 THEN
 INPUT "Direction Of Travel + or - ", d$
 SELECT CASE d$
  CASE ""
   Direction% = "+"
  CASE "+"
   Direction$ = d$
  CASE "-"
   Direction$ = d$
 END SELECT
 LOCATE 58: PRINT "                              "
 LOCATE 58
 INPUT "Stream Side Up or Down ", S$
 LOCATE 58: PRINT "                              "
 SELECT CASE UCASE$(S$)
  CASE "U"
   Stream% = "U"
  CASE "D"
   Stream$ = "D"
 END SELECT
 LOCATE 58
 INPUT "Pile Number: ", PName$
 LOCATE 58: PRINT "                              "
 COLOR 4
 LOCATE 35, 6: PRINT "                "
 LOCATE 3 5, 6: PRINT "STOP PILE LOG "
 PRINT #2, "FILENAME: "; PileFile$, "DATE: "; DATE$, "TIME: "; TIME%
 PRINT #2, "BRIDGE: "; BName$, "CO/CITY: "; CName$, "STATE: "; SName$
 PRINT #2, "US or DS: "; Stream%, 'TYPE OF DECK: "; TName$, "PILE NUMBER: "; PName$
 PRINT #2, "OFFSETS: ", "Yos: "; Yos!, "Zos: "; Zos!, "Los: "; Los!
 PRINT #2, "STARTING STATION: "; Station!, "DIRECTION: "; Direction$, "STARTING DECK
ELEVATION: "; Elevation
 PRINT #2,
 PRINT #2, " Xa "; " Ya "; " BL "; " Sa "; " Od "; " Ia "; " La "; " Xf "; " Yf "; " Zf "; " Hb ";
 PRINT #2, " Xb "; " Yb "; " Zb "; " Lb "; " Zt "; " Xtt "; " Xbt "; " Ed "; " Ls "; " Eb "
 PRINT #2,
 Key2 = 1
END IF
ELSE
 CLOSE #2
 COLOR 10
 LOCATE 35, 6: PRINT "START PILE LOG"
 Key2 = 0
END IF
CASE f$(3)
CLS
SCREEN 0
'Tnum = LEN(FileName$)
'IF Tnum > 0 THEN
' FI$ = LEFT$(FileName$, Tnum - 4)
' FI$ = FI$ + ".TXT"
'END IF
SHELL "EDIT "

INIT
```

```
  CASE f$(4)
   IF Trail THEN Trail = 0 ELSE Trail = 1
    IF Trail THEN
     LOCATE 39, 14
     COLOR 15
     PRINT "ON "
    ELSE
     LOCATE 39, 14
     COLOR 15
     PRINT "OFF"
     PilePass = 0
     PCount = 0
    END IF
  CASE f$(5)
   IF Key1 THEN BRIDGELOG
   IF Key2 THEN PILELOG
  CASE f$(6)
   DJAGNOSTICS
  CASE f$(7)
   PARAMETERS
  CASE f$(8)
   MenuKey = 2
   MENU2
  CASE f$(9)

  CASE f$( 10)
   COUNTRESET 1
  CASE f$(11)
   COUNTRESET 2
  CASE f$( 12)
   CLOSE
   'CLEAR       'RESETS OUTPUTS TO 0
   Vout#(0) = 2.5
   WRITEOUTPUTS
   END
 END SELECT
END SUB

SUB MENU2
 COLOR 12
 LOCATE 3 1, 2: PRINT "     MENU2          "
 LOCATE 32, 2: PRINT "                    "

 COLOR 13
 LOCATE 33, 6: PRINT "ENTER STATION    " 'Fl
 LOCATE 35, 6: PRINT "ENTER ALTITUDE   " 'F2
 LOCATE 37, 6: PRINT "ENTER BOOM OFFSETS" 'F3
 LOCATE 39, 6: PRINT "                 " 'F4
 LOCATE 4 1, 6: PRINT "                " 'F5
 LOCATE 43, 6: PRINT "DIAGNOSTIC PAGE " 'F6
 LOCATE 45, 6: PRINT "PARAMETERS PAGE " 'F7
 LOCATE 47, 6: PRINT "MENU 1           " 'F8
 LOCATE 49, 6: PRINT "                 " 'F9
 LOCATE 5 1, 6: PRINT "                " 'F10
 LOCATE 53, 6: PRINT "                 " 'Fll
```

```
 LOCATE 55, 6: PRINT "              " 'F12

END SUB

SUB MENU2FUNCTIONS (KeyPress$)
 SELECT CASE KeyPress$
  CASE f$(1)
   LOCATE 58
   INPUT "Start Station # ", Station!
   LOCATE 58, 25
   INPUT "Direction Of Travel From Station + or - ", d$
   SELECT CASE d$
    CASE ""
     Direction$ = "+"
    CASE "+"
     Direction$ = d$
    CASE "-"
     Direction$ = d$
   END SELECT
   LOCATE 58: PRINT "
  CASE f$(2)
   LOCATE 58
   INPUT "Starting Elevation", Elevation
   LOCATE 58: PRINT "              "
  CASE f$(3)
   DO
    LOCATE 57, 20: PRINT "Select to Change: 1) Yos  2) Zos 3) Los 4) Cancel "
    Choice = VAL(INKEY$)
    SELECT CASE Choice
     CASE 1
      LOCATE 58, 1: INPUT "ENTER Distance from Deck Edge to Center Pin (Yos): ", Yos!
      EXIT DO
     CASE 2
      LOCATE 58: PRINT "                                        ";
      EXIT DO
      LOCATE 58, 1: INPUT "ENTER Distance from Boom Pivot to Deck (Zos): ", Zos!
     CASE 3
      LOCATE 58: PRINT "                                        ";
      EXIT DO
      LOCATE 58.1: INPUT "ENTER Distance from Laser to Deck (Los): ", Los!
     CASE 4
      EXIT DO
    END SELECT
   LOOP
   LOCATE 57, 20: PRINT "                                   "
   PRINT "                                        ";
   DatNum = FREEFILE
   OPEN "OFFSETS.DAT" FOR OUTPUT AS #DatNum
   PRINT #DatNum, Zos!
   PRINT #DatNum, Yos!
   PRINT #DatNum, Los!
   CLOSE #DatNum
  CASE f$(4)
  CASE f%(5)
```

```
CASE f$(6)
 DIAGNOSTICS
CASE f$(7)
 PARAMETERS
CASE f$(8)
 MenuKey = 1
 KeyPress$ = ""
 MENU1
CASE f$(9)
```

```
  CASE f$( 10)
   'COUNTRESET 1
  CASE f$(11)
   'COUNTRESET 2
  CASE f$( 12)
 END SELECT


END SUB

SUB PARAMETERS
 CLS
'Outside border
 LINE (0, 0)-(639, 445), 1, B
 LINE (3, 3)-(636, 442), 1, B
 PAINT(1, 1), 1, 1
'Horizontal borders
 LINE (0, 18)-(639, 2 1), 1, BF
 LINE (0, 85)-(639, 88), 1, BF
 LINE (360, 120)-(639, 123), 1, BF

'Vertical borders
 LINE (360, 88)-(363, 442), 1, BF
 LINE (490, 120)-(493, 442), 1, BF


 DO WHILE KeyPress$ <> f$(7)
  KeyPress% = INKEY$
  IF KeyPress$ = CHR$(27) THEN EXIT DO
  IF KeyPress$ = f$( 11) THEN COUNTRESET 2
  IF KeyPress$ = f$( 10) THEN COUNTRESET 1
  IF KeyPress$ = f$( 13) THEN SCRNPRNT
  READINPUTS
  CALL CALCULATE(Station!, Elevation, BGCount)
  WRITEOUTPUTS
  LOCATE 2, 32: PRINT "PARAMETERS PAGE"
  COLOR 13
  LOCATE 4, 25: PRINT "Parameters As Written to Files"
  LOCATE 6, 4: COLOR 12
  PRINT "  Xa "; " Ya "; " BL "; " Sa "; " Od "; " Ia "; " La "; " Xf "; " Yf "; " Zf "; " Hb "
  LOCATE 7, 4: COLOR 15
  '        Xa   Ya   BL   Sa   Od   Ia   La   Xf   Yf   Zf   Hb
  PRINT USING " +##.# +##.# ##.# +##.# ####.# +##.## +##.# +##.# +##.# +##.# +##.#"; Xa!; Ya!; BL!; Sa!; Od!;
Ia!; La!; Xf!; Yf!; Zf!; Hb!

  LOCATE 9, 4: COLOR 12
  PRINT " Xb "; " Yb "; " Zb "; " Lb "; " Zt "; " Xtt "; " Xbt "; " Ed "; " Ls "; " Eb "
  LOCATE 10, 4: COLOR 15
  '        Xb   Yb   Zb   Lb   Zt   Xtt   Xbt   Ed   Ls   Eb
  PRINT USING "####.# +##.# +##.# +##.# +###.# #####.# #####.# ####.# ####.# ####.#"; Xb!; Yb!; Zb!; Lb!;
Zt!; Xtt!; Xbt!; Ed!; La!; Eb!

  LOCATE 13, 55: COLOR 13
  PRINT "SCALE FACTORS"
  LOCATE 15, 50: COLOR 14: PRINT "Offsets Scales"
```

```
    LOCATE 18, 36: PRINT 'Yaw Angle"
    LOCATE 20, 34: PRINT 'Pitch Angle"
    LOCATE 22, 34: PRINT "Boom Length"
    LOCATE 24, 32: PRINT 'Trailer Angle"
    LOCATE 26, 36: PRINT "Odometer"
    LOCATE 28, 40: PRINT "Sonar"
    LOCATE 30, 40: PRINT "Laser"
    COLOR 15
    LOCATE 18, 50: PRINT USING "+#.#######"; Yaw0#
    LOCATE 18, 68: PRINT USING "+##.######"; YawSpan#
    LOCATE 20, 50: PRINT USING "+#.#######"; PitchO#
    LOCATE 20, 68: PRINT USING "+##.######"; PitchSpan#
    LOCATE 22, 50: PRINT USING "+##.######"; BL0#
    LOCATE 22, 68: PRINT USING "+#.#######"; BLSpan#
    LOCATE 24, 50: PRINT USING "+#.#######"; TAngleO#
    LOCATE 24, 68: PRINT USING "+##.######"; TAngleSpan#
    LOCATE 26, 50: PRINT USING "+#.#######"; OddO#
    LOCATE 26, 68: PRINT USING "+##.######"; OddSpan#
    LOCATE 28, 50: PRINT USING "+#.#######"; Sonar0#
    LOCATE 28, 68: PRINT USING "+##.######"; SonarSpan#
    LOCATE 30, 50: PRINT USING "+#.#######"; Laser0#
    LOCATE 30, 68: PRINT USING "+#.#######"; LaserSpan#


    'LOCATE 50, 4: PRINT "F7: EXIT", "F10: TRAILER RESET", "F11: BOOM RESET"
   LOOP
   INIT
 END SUB

 SUB PILEGRAPH (PilePass, PCount)

 STATIC XDispTemp%, YDispTemp%, Warning
 'PILE GRAPH
   IF PilePass < 1 THEN
    VIEW SCREEN (375, 256)-(630, 440)
    WINDOW
    LINE (380, 256)-(630, 440), 11, BF
    LINE (380, 256)-(630, 440), 8, B
    PilePass = PilePass + 1
 'horizontal lines
    COLOR 3
    LINE (375, 258)-(630, 258) 'max range line
    LINE (375, 298)-(630, 298)
    LINE (375, 338)-(630, 338)
    LINE (375, 378)-(630, 378)
    LINE (375, 418)-(630, 418)
    LINE (375, 438)-(630, 438)
    COLOR 13
    LOCATE 33, 44: PRINT " "
    LOCATE 33, 44: PRINT 3 * PHite
    LOCATE 38, 44: PRINT " "
    IF (PHite * 2) > 9 THEN xpos = 0 ELSE xpos = 1

    LOCATE 38, 44 + xpos: PRINT 2 * PHite
```

```
  LOCATE 43, 44: PRINT " "
  IF PHite > 9 THEN xpos = 0 ELSE xpos = 1
  LOCATE 43, 44 + xpos: PRINT 1 * PHite
  LOCATE 48, 45: PRINT 0
  LOCATE 53, 44: PRINT " "
  IF PHite > 9 THEN xpos = 0 ELSE xpos = 1
  LOCATE 53, 44 + xpos: PRINT 1 * PHite
'vertical lines
  COLOR 3
  LINE (385, 257)-(385, 438)
  LINE (425, 257)-(425, 438)
  LINE (465, 257)-(465, 438)
  LINE (505, 257)-(505, 438)
  LINE (545, 257)-(545, 438)
  LINE (585, 257)-(585, 438)
  LINE (625, 257)-(625, 438)
  COLOR 13
  LOCATE 32, 47: PRINT " ";
  LOCATE, 47: PRINT -3 * PHite;
  LOCATE , 53: PRINT " ";
  LOCATE, 53: PRINT -2 * PHite;
  LOCATE, 58: PRINT " ";
  LOCATE, 58: PRINT -1 * PHite;
  LOCATE, 63: PRINT 0;
  LOCATE , 68: PRINT " ";
  LOCATE , 68: PRINT 1 * PHite;
  LOCATE, 72: PRINT " ";
  LOCATE, 72: PRINT 2 * PHite;
  LOCATE, 77: PRINT " ";
  LOCATE, 77: PRINT 3 * PHite,
'circles
  LINE (502, 375)-(508, 381), 1
  LINE (502, 381)-(508, 375), 1
  COLOR 1                          'Center Circle
  LINE (402, 438)-(608, 438)             'Bottom line of Scale Circle
 END IF
 VIEW (385, 258)-(625, 438)
 WINDOW (30 * PHite, 30 * PHite)-(-30 * PHite, -15 * PHite)
 CIRCLE (0, 0), 300, 1, PI * 1.835, PI * 1.165          'SCALE CIRCLE
'trails
 IF Trail THEN
  FOR X = 0 TO TrailNum
   BoxX = Trails!(X, 0) * 10
   BoxY = Trails!(X, 1) * 10
   LINE (BoxX - 3, BoxY - 3)-(BoxX + 3, BoxY + 3), 15, BF
  NEXT X
 END IF
'boom Position
 SELECT CASE Xf! * 10
  CASE IS >= (PHite * 30) - PHite - 7
   XDisp% = (PHite * 30) - PHite - 7
   Extreme = 1
  CASE IS <= (-PHite * 30)
   XDisp% = (-PHite * 30)
   Extreme = 1
```

```
  CASE ELSE
   XDisp% = (Xf! * 10) - PHite + PHite / 3
 END SELECT

 SELECT CASE Yf! * 10
  CASE IS >= (PHite * 30) - PHite - 7
   YDisp% = (PHite * 30) - PHite - 7
   Extreme = 1
  CASE IS <= (-PHite * 15)
   YDisp% = (-PHite * 15)
   Extreme = 1
  CASE ELSE
   YDisp% = (Yf! * 10) - PHite + PHite / 3
 END SELECT

 IF PCount THEN
  PUT (XDispTemp%, YDispTemp%), HeadImage
 END IF

 PUT (XDisp%, YDisp%), HeadImage
 XDispTemp% = XDisp%
 YDispTemp% = YDisp%
 IF PCount < 1 THEN PCount = PCount + 1

 IF Extreme THEN
  LINE (-30 * PHite + 1, -15 * PHite + 1)-(30 * PHite - 1, 30 * PHite - 1), 4, B
  LINE (-30 * PHite, -15 * PHite)-(30 * PHite, 30 * PHite), 15, B
  Warning = 1
 END IF

 IF NOT Extreme AND Warning THEN
  LINE (-30 * PHite, - 15 * PHite)-(30 * PHite, 30 * PHite), 3, B
  Warning = 0
 END IF

 VIEW
 WINDOW
 LINE (636, 240)-(639, 260), 1, BF
END SUB

SUB PILELOG

'          Xa  Ya  BL  Sa  Od  Ia  La  Xf  Yf  Zf  Hb  Xb  Yb  Zb  Lb  Zt  Xtt  Xbt  Ed  Ls
Eb
 PRINT #2, USING "+##.# +##.# ##.# +##.# ####.# +##.## +##.# +##.# +##.# +##.# +##.# ####.# +##.# +##.#
+##.# +###.# #####.# #####.# ####.# ####.# ####.#"; Xa!; Ya!; BL!; Sa!; Od!; Ia!; La!; Xf!; Yf!; Zf!; Hb!; Xb!; Yb!;
Zb!; Lb!; Zt!; Xtt!; Xbt!; _
Ed!; Ls!; Eb!

END SUB

SUB READINPUTS

'Analog inputs Board 1
 err.num = AI.VRead( 1 , 0, 1, Vin#(0))    'RIGHT JOYSTICK FORE - AFT
```

```
 Err.Exit "AI.VRead7", err.num
 err.num = AI.VRead( 1, 1, 1, Vin#(1))    'RIGHT JOYSTICK LEFT - RIGHT
 Err.Exit "AI.VRead 1 ", err.num
 err.num = AI.VRead( 1, 2, 1, Vin#(2))    'LEFT JOYSTICK FORE - AFT
 Err.Exit "AI.VRead2", err.num
 err.num = AI.VRead( 1, 3, 1, Vin#(3))    'RIGHT JOYSTICK FORE - AFT
 Err.Exit "AI.VRead3", err.num
 err.num = AI.VRead( 1, 4, 1, Vin#(4))    'RIGHT JOYSTICK LEFT - RIGHT
 Err.Exit "AI.VRead4", err.num
 err.num = AI.VRead( 1, 5, 1, Vin#(5))    'LEFT JOYSTICK FORE - AFT
 Err.Exit "AI.VRead5", err.num
 err.num = AI.VRead( 1, 6, 1, Vin#(6))    'RIGHT JOYSTICK FORE - AFT
 Err.Exit "AI.VRead6", err.num
 err.num = AI.VRead( 1, 7, 1, Vin#(7))    'RIGHT JOYSTICK LEFT - RIGHT
 Err.Exit "AI.VRead7", err.num
'Analog inputs Board 2
 err.num = AI.VRead(2, 0, 1, Vin#(8))    'RIGHT JOYSTICK FORE - AFT
 Err.Exit "AI.VRead8", err.num
 err.num = AI.VRead(2, 1, 1, Vin#(9))
 Err.Exit "AI.VRead9", err.num
 err.num = AI.VRead(2, 2, 1, Vin#( 10))
 Err.Exit ''AI.VRead1 0", err.num
 err.num = AI.VRead(2, 3, 1, Vin#(1 1))
 Err.Exit "AI.VRead1 1 ", err.num
 err.num = AI.VRead(2, 4, 1, Vin#( 12))
 Err.Exit "AI.VRead12", err.num
 err.num = AI.VRead(2, 5, 1, Vin#( 13))
 Err.Exit "AI.VRead13", err.num
 err.num = AI.VRead(2, 6, 1, Vin#( 14))
 Err.Exit "AI.VRead14", err.num
 err.num = AI.VRead(2, 7, 1, Vin#( 15))
 Err.Exit "AI.VRead15", err.num
Digital inputs Board 1
 err.num = DIG.In.Line(1, 0, 0, Din(O))  'OPERATE - DEPLOY SWITCH - 0 = DEPLOY
 Err.Exit "DIG.In.Line0", err.num
 err.num = DIG.In.Line( 1, 0, 1, Din( 1))
 Err.Exit "DIG.In.Line 1", err.num
 err.num = DIG.In.Line( 1, 0, 2, Din(2))  'AUTO TILT SWITCH
 Err.Exit "DIG.In.Line2", err.num
 err.num = DIG.In.Line(1, 0, 3, Din(3))  'HEAD POWER ON - OFF SWITCH
 Err.Exit "DIG.In.Line3", err.num
 err.num = DIG.In.Line(1, 0, 4, Din(4))  'BOOM STOP PROXIMITY SWITCH
 Err.Exit "DIG.In.Line4", err.num
 err.num = DIG.In.Line( 1, 0, 5, Din(5))  'SPARE IN 2
 Err.Exit "DIG.In.Line5", err.num
 err.num = DIG.In.Line( 1, 0, 6, Din(6))  'SPARE IN 3
 Err.Exit "DIG.In.Line6", err.num
 err.num = DIG.In.Line( 1 , 0, 7, Din(7))  'AUTO Z SWITCH
 Err.Exit "DIG.In.Line7", err.num
'Digital inputs Board 2
 err.num = DIG.In.Line(2, 0, 0, Din(8))  'Spare in 4
 Err.Exit "DIG.In.Line8", err.num
 err.num = DIG.In.Line(2, 0, 1, Din(9))  'Spare in 5
 Err.Exit "DIG.In.Line9", err.num
 err.num = DIG.In.Line(2, 0, 2, Din( 10))  'Spare in 6
```

```
 Err.Exit "DIG.In.Line10", err.num
 err.num = DIG.In.Line(2, 0, 3, Din( 11))   'Spare in 7
 Err.Exit "DIG.In.Line1 1", err.num
 err.num = DIG.In.Line(2, 0, 4, Din( 12))   'Spare in 8
 Err.Exit "DIG.In.Line12", err.num
 err.num = DIG.In.Line(2, 0, 5, Din( 13))   'Spare in 9
 Err.Exit "DIG.In.Line13", err.num
 err.num = DIG.In.Line(2, 0, 6, Din( 14))   'Spare in 10
 Err.Exit "DIG.In.Line 14", err.num
 err.num = DIG.In.Line(2, 0, 7, Din( 15))   'SPARE IN 11
 Err.Exit "DIG.In.Line1 5", err.num
'Counter inputs Board I
 err.num = ICTR.Read(1, 1, Ctr 1)
 Err.Exit "ICTR.Read1 ", err.num
 err.num = ICTR.Read( 1, 2, Ctr2)
 Err.Exit "ICTR.Read2", err.num
 Ctr1& = Ctr1
 Ctr2& = Ctr2
 IF Ctr1 >= 0 THEN
  CTR&( 1) = ABS(Ctr1& - 65536)
  IF CTR&( 1) = 65536 THEN CTR&( 1) = 0
 ELSE
  CTR&( 1) = ABS((Ctr1& + 65536) - 65536)
 END IF

 IF Ctr2 > 0 THEN
  CTR&(2) = ABS(Ctr2& - 65536)
 ELSE
  CTR&(2) = ABS(Ctr2&)
 END IF
'Counter inputs Board 2
 err.num = ICTR.Read(2, I, Ctr4)
 Err.Exit "ICTR.Read4", err.num
 err.num = ICTR.Read(2, 2, Ctr5)
 Err.Exit "ICTR.Read5", err.num
 Ctr4& = Ctr4
 Ctr5& = Ctr5

 IF Ctr4 >= 0 THEN
  CTR&(4) = ABS(Ctr4& - 65536)
  IF CTR&(4) = 65536 THEN CTR&(4) = 0
 ELSE
  CTR&(4) = ABS((Ctr4& + 65536) - 65536)
 END IF

 IF Ctr5 > 0 THEN
  CTR&(5) = ABS(Ctr5& - 65536)
 ELSE
  CTR&(5) = ABS(Ctr5&)
 END IF
  CTR&(6) = CTR&(4) - CTR&(5)
END SUB

SUB SCREENWRITE (Station!, Elevation)
'Joystick section
```

```
 COLOR 15
 LOCATE 2, 15: PRINT USING "+#.##"; Vin#(0)   'rh js f/a
 LOCATE 2, 43: PRINT USING "+#.##"; Vin#( 1)   'rh js l/r
 LOCATE 2, 68: PRINT USING "+#.##"; Vin#(2)   'lh js f/a
 LOCATE 4, 14                'head power
 IF Din(3) = 0 THEN
  PRINT "OFF"
 ELSE
  PRINT "ON "
 END IF
 LOCATE 4, 40                'auto-z
 IF Din(7) = 0 THEN
  PRINT "OFF"
 ELSE
  PRINT "ON "
 END IF
 LOCATE 4, 72:               'auto-tilt
 lFDin(l)=OTHEN
  PRINT "OFF"
 ELSE
  PRINT "ON "
 END IF
 LOCATE 6, 15
 IF Din(O) = 0 THEN              'MODE SELECT
  PRINT "DEPLOY "
 ELSE
  PRINT "OPERATE"
 END IF
 LOCATE 6, 38: PRINT USING "### \\"; (Vout#(1) * 10); CHR$(37)      'SPEED VALVE OUTPUT
 LOCATE 6, 65: PRINT USING "+### \\"; (Vout#(0) - 2.5) * 58; CHR$(37) 'WINCH OUT
'Bridge section
 COLOR 15:
 LOCATE 26, 12: PRINT Direction$
 LOCATE 28, 8: IF Stream$ = "U" THEN PRINT "UpStream" ELSE PRINT "DownStream"
 LOCATE 26, 3 1: PRINT USING "#####\\"; Xtt!; ""
 LOCATE 28, 3 1: PRINT USING "#####\\"; Ed!; ""
 LOCATE 26, 49: PRINT USING "#####.#\\"; Od!        'CTR&(0) / 10; ""
 LOCATE 28, 52: PRINT USING "##.#\\"; La!; ""
 LOCATE 26, 7 1: PRINT USING "+##.##\\"; Ia!; CHR$(248)
 LOCATE 28, 72: PRINT USING "+##.#\\"; Hb!; ""     'must calculate elevation
'Pile Section
 COLOR 15
 LOCATE 33, 37: PRINT USING "+##.#\\"; Ya!; CHR$(248)
 LOCATE 34, 37: PRINT USING "+##.#\\"; Xa!; CHR$(248)
 LOCATE 36, 37: PRINT USING "+##.#\\"; Yf!; ""
 LOCATE 37, 37: PRINT USING "+##.#\\"; -Xf!; ""
 LOCATE 39, 37: PRINT USING "+##.#\\"; Yb!; ""
 LOCATE 40, 34: PRINT USING "+#####.#\\"; Xb!; ""
 LOCATE 42, 38: PRINT USING "##.#\\"; BL!; ""
 LOCATE 44, 38: PRINT USING "##.#\\"; Sa!; ""
 LOCATE 48, 28: PRINT Yos!; Zos!; Los!
 LOCATE 52, 38: PRINT USING "##.#\\"; Zb!; ""
 LOCATE 54, 37: PRINT USING "###.#\\"; Zt!; ""
END SUB
```

```
SUB SCRNPRNT
ESC$ = CHR$(27)
WIDTH LPRINT 255

LPRINT ESC$; "%-12345X";
LPRINT ESC$; "E";
LPRINT ESC$; "*t75R";
LPRINT ESC$; "&l1O";
LPRINT ESC$; "*p300x200Y";
LPRINT ESC$; "*r0F";
LPRINT ESC$; "*r1A";

FOR ROW = 479 TO 0 STEP -1
 LPRINT ESC$; "*b80W";
 FOR NEST = 79 TO 0 STEP -1
  a = 0
  FOR COL = 0 TO 7
   X = POINT(COL + NEST * 8, ROW)
   IF X <> 0 AND X <> 11 THEN a = a + (2 ^ COL)
  NEXT COL
  LPRINT CHR$(a);
 NEXT NEST
NEXT ROW
LPRINT ESC$; "*rC"
LPRINT ESC$; "E";
LPRINT ESC$; "%-12345X";

END SUB

SUB WRITEOUTPUTS
'BOARD 1
'Analog outputs
 err.num = AO.VWrite(1, 1, Vout#(1))          'PROPORTIONAL VALVE OUT
 Err.Exit "AO.VWrite1", err.num
 err.num = AO.VWrite(1, 0, Vout#(0))          'DART CONTROL OUT
 Err.Exit "AO.VWrite2", err.num

'Digital Outputs
 err.num = DIG.Out.Line(1, 1, 0, Dout(0))     'VALVE 1A
 Err.Exit "DIG.Out.Line0", err.num
 err.num = DIG.Out.Line(1, 1, 1, Dout(1))     'VALVE 1B
 Err.Exit "DIG.Out.Line1", err.num
 err.num = DIG.Out.Line(1, 1, 2, Dout(2))     'VALVE 2A
 Err.Exit "DIG.Out.Line2", err.num
 err.num = DIG.Out.Line(1, 1, 3, Dout(3))     'VALVE 2B
 Err.Exit "DIG.Out.Line3", err.num
 err.num = DIG.Out.Line(1, 1, 4, Dout(4))     'VALVE 3A
 Err.Exit "DIG.Out.Line4", err.num
 err.num = DIG.Out.Line(1, 1, 5, Dout(5))     'VALVE 3B
 Err.Exit "DIG.Out.Line5", err.num
 err.num = DIG.Out.Line(1, 1, 6, Dout(6))     'Spare Output
 Err.Exit "DIG.Out.Line6", err.num
 err.num = DIG.Out.Line(1, 1, 7, Dout(7))     'Spare Output
 Err.Exit "DIG.Out.Line7", err.num
 err.num = DIG.Out.Line(1, 2, 0, Dout(8))     'VALVE 4A
```

```
 Err.Exit "DIG.Out.Line8", err.num
 err.num = DIG.Out.Line( 1, 2, 1, Dout(9))    'VALVE 4B
 Err.Exit "DIG.Out.Line9", err.num
 err.num = DIG.Out.Line( 1, 2, 2, Dout( 10))   'VALVE 5A
 Err.Exit "DIG.Out.Line10", err.num
 err.num = DIG.Out.Line( 1, 2, 3, Dout( 11))   'VALVE 5B
 Err.Exit "DIG.Out.Line1 1", err.num
 err.num = DIG.Out.Line(1, 2, 4, Dout( 12))   'WINCH UP
 Err.Exit "DIG.Out.Line12", err.num
 err.num = DIG.Out.Line(1, 2, 5, Dout( 13))   'WINCH DOWN
 Err.Exit "DIG.Out.Line13", err.num
 err.num = DIG.Out.Line( 1, 2, 6, Dout( 14))   'Spare Output
 Err.Exit "DIG.Out.Line14", err.num
 err.num = DIG.Out.Line( 1, 2, 7, Dout( 15))   'Trailer Counter Reset
 Err.Exit "DIG.Out.Line15", err.num

'BOARD 2
'Analog Outputs
 err.num = AO.VWrite(2, 0, Vout#(2))    'Boom Counter Output
 Err.Exit "AO.VWrite3", err.num
 err.num = AO.VWrite(2, 1, Vout#(3))     'Spare Output
 Err.Exit "AO.VWrite4", err.num

'Digital Outputs
 err.num = DIG.Out.Line(2, 1, 0, Dout( 16))   'Spare Output
 Err.Exit "DIG.Out.Line0", err.num
 err.num = DIG.Out.Line(2, 1, 1, Dout( 17))   'Spare Output
 Err.Exit "DIG.Out.Line1", err.num
 err.num = DIG.Out.Line(2, 1, 2, Dout( 18))   'Spare Output
 Err.Exit "DIG.Out.Line2", err.num
 err.num = DIG.Out.Line(2, 1, 3, Dout( 19))   'Spare Output
 Err.Exit "DIG.Out.Line3", err.num
 err.num = DIG.Out.Line(2, 1, 4, Dout(20))   'Spare Output
 Err.Exit "DIG.Out.Line4", err.num
 err.num = DIG.Out.Line(2, 1, 5, Dout(2 1))   'Spare Output
 Err.Exit "DIG.Out.Line5", err.num
 err.num = DIG.Out.Line(2, 1, 6, Dout(22))   'Spare Output
 Err.Exit "DIG.Out.Line6", err.num
 err.num = DIG.Out.Line(2, 1, 7, Dout(23))   'Spare Output
 Err.Exit "DIG.Out.Line7", err.num
 err.num = DIG.Out.Line(2, 2, 0, Dout(24))   'Spare Output
 Err.Exit "DIG.Out.Line8", err.num
 err.num = DIG.Out.Line(2, 2, 1, Dout(25))   'Spare Output
 Err.Exit "DIG.Out.Line9", err.num
 err.num = DIG.Out.Line(2, 2, 2, Dout(26))   'Spare Output
 Err.Exit "DIG.Out.Line10", err.num
 err.num = DIG.Out.Line(2, 2, 3, Dout(27))   'Spare Output
 Err.Exit "DIG.Out.Line 11 ", err.num
 err.num = DIG.Out.Line(2, 2, 4, Dout(28))  'Spare Output
 Err.Exit "DIG.Out.Line12", err.num
 err.num = DIG.Out.Line(2, 2, 5, Dout(29))   'Spare Output
 Err.Exit "DIG.Out.Line13", err.num
 err.num = DIG.Out.Line(2, 2, 6, Dout(30))   'Spare Output
 Err.Exit "DIG.Out.Line14", err.num
 err.num = DIG.Out.Line(2, 2, 7, Dout(3 1))   'Boom Counter Reset
```

```
  Err.Exit "DIG.Out.Line1 5", err.num
END SUB
```