TRC 95

ARKANSAS CENTER FOR TECHNOLOGY TRANSFER

Final Report

Bar Code Reading of Moving Trucks

Center for Robotics and Automation

Transportation Research Project No. 95

November 1990

UNIVERSITY OF ARKANSAS
College of Engineering

UA Engineering Research Center
West 20th Street
Fayetteville, Arkansas 72701
1-800-334-3571 (in-state)
(501) 575-6696

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
FHWA/AR-90/005		

4. Title and Subtitle	5. Report Date
Bar Code Reading of Moving Trucks	November 1990
	6. Performing Organization Code
	8. Performing Organization Report No.
7. Author(s)	
Marcus C. Langston	
9. Performing Organization Name and Address	10. Work Unit No. (TRAIS)
Center for Robotics and Automation University of Arkasnas Fayetteville, AR 72701	11. Contract or Grant No. HPR-3(014)
	13. Type of Report and Period Covered
12. Sponsoring Agency Name and Address	Final Project Report Dec. 1, 1986 thru June 30, 1989
Arkansas State Highway & Transportation Dept. P. O. Box 2261 Little Rock, AR 72203	14. Sponsoring Agency Code TRC-95

15. Supplementary Notes

Prepared in cooperation with the U.S. Department of Transportation, Federal Highway Administration and the Arkansas State Highway and Transportation Department.

16. Abstract

Barcoding is a cost effective, practical solution to accomplish automatic vehicle identification (AVI) at weigh stations. Hardware cost for each AVI station is estimated to be less than $9000. Large scale unit cost to bar code each vehicle is estimated to be $0.80.

Technology may also be utilized by surrounding states to reduce the amount of time, personnel and record keeping necessary to track trucks traversing their states. Information shared through computer links will allow the states to increase the efficiency of each weigh station's operation and increase the revenue by decreasing the amount of "wave throughs" when the trucks back up along the highway.

Substantial money can be saved by the trucking industry due to the ease of truck registration using bar codes and having this registration information available to other states on the computer system. There is an added savings of time and fuel because trucks which are legal do not have to come to a complete stop at weigh stations only to have to restart because they are within the state's regulations. The environment benefits due to the reduction in air pollution because the trucks do not waste fuel waiting in line.

17. Key Words	18. Distribution Statement
Bar Code Scanning AVI	

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

BAR CODE READING OF MOVING TRUCKS

Final Report

Transportation Research Project No. 95

November 1990

by

University of Arkansas at Fayetteville
Center for Robotics and Automation

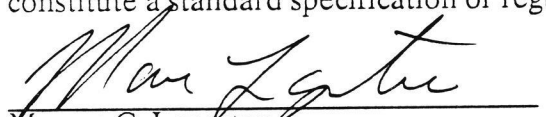
and the

Arkansas State Highway and Transportation Department

In Cooperation

with the

Federal Highway Administration



Marcus C. Langston
Principal Investigator

GAINS, FINDINGS AND CONCLUSIONS

The primary gains, findings, and conclusions of this study are as follows:

I. Gains and Findings

1. An "off the shelf" laser bar code reader was developed which successfully read 99.63% of the time under simulated weigh station operating conditions.

2. Laminating materials were identified and tested which extended the useful life of standard bar codes to over five years (the anticipated life of a commercial vehicle).

3. Successful installation of a prototype system at Lehi weigh station in West Memphis, Arkansas. Trained weigh station personnel in the operation of the system.

4. Developed a specialized communication system for data transfer between the laser scanner and host computer utilizing the RS-422 communications interface.

5. Incorporated communication software between the development team at the U of A and the weigh station personnel. Utilized direct communication link between the U of A and Lehi weight station.

6. Completion of software to support system integration between electronic weigh-in-motion scales, bar code scanner, and microcomputer data base.

II. Conclusions

1. The researchers responsible for this report have no reservations as to the technological feasibility of automatic vehicle identification (AVI) using bar code technology.

2. Barcoding is a cost effective, practical solution to accomplish automatic vehicle identification at weigh stations. The AVI system developed employs "off the shelf" existing technology. The hardware cost for each AVI station is estimated to be less than $9000. The unit cost to bar code each vehicle is estimated to be $0.80 (when conducted on a large scale).

3. This technology may also be utilized by surrounding states to reduce the amount of time, personnel and record keeping necessary to track trucks traversing their states. this ability to share information through computer links will allow the states to increase the efficiency of each weigh station's operation and increase the revenue by decreasing the amount of "wave throughs" when the trucks back up along the highway.

4. A substantial amount of money can be saved by the trucking industry due to the ease of truck registration using bar codes and having this registration information available to other states on the computer system. There is also the added savings of time and fuel because trucks which are legal do not have to come to a complete stop at weigh stations only to have to restart because they are within the state's regulations. The environment benefits due to the reduction in air pollution because the trucks do not waste fuel waiting in line.

TABLE OF CONTENTS

Appendix

LIST OF TABLES AND FIGURES

Tables:

Figures:

INTRODUCTION

Background

In 1985 the Arkansas State Highway and Transportation Department along with other state and federal agencies began funding project TRC-95: "Bar Code Reading of Trucks in Weigh Stations." The goal of this project was to determine the feasibility of using bar codes to track trucks as they pass through the weigh stations.

When a commercial vehicle pulls into a weigh station, the personnel at the station visually check the side of the vehicle for a current sticker and halts the truck to weigh the vehicle on the static scales. This process is slow and error prone because every state requires their own stickers. If the vehicle does not have a current sticker or is over weight, the personnel at the station then tells the driver over a loud speaker to pull over into a parking area and come into the weigh station. The personnel at the weigh station now also have the added responsibility of checking the paper work on the vehicle while weighing other vehicles and checking for stickers.

Since every truck must stop to be weighed, a que of trucks builds up. When the que backs up past the entrance of the weigh station and onto the highway, the trucks are allowed to by pass the weigh station altogether. Drivers who are operating overweight vehicles, whose paper work is not up to date or who do not have a current sticker on their vehicle, will try to get past the weigh station at this time. This problem not only causes a safety hazard, but also means lost revenue for the state in the form of fines not paid by the illegal drivers.

This type of system also costs the owner/operators of commercial vehicles many hundreds of dollars a year. Most trucking firms have two to three full time employees whose only job is to keep up with the paper work required by each state their vehicles pass through and to keep the stickers up to date on the vehicles. Also, because the vehicles have to stop at the weigh station, fuel is wasted slowing from highway speed, waiting in que, and returning to highway speed.

The current system is unsafe, inefficient, and wasteful in personnel and fuel for both the state and the owner/operators of commercial vehicles. A system incorporating AVI technology using bar codes and a weigh-in-motion scale would improve the current system and benefit both the state and owner/operators.

Research Results

A four year research project conducted by the University of Arkansas at Fayetteville and the Arkansas State Highway and Transportation Department determined that the use of bar codes is a feasible means of automatically identifying commercial vehicles. During field testing, this system proved capable of providing a first time read rate of over 97% of the vehicles which passed by the scanner.

Results from Phase I through Phase III produced the following results:

1. Maximization of system hardware achieving a 97% first-pas read rate.
2. Research produced a bar code emblem which will last the anticipated life of the truck.
3. Developed a communications system linking the University of Arkansas development team and the weigh station personnel. The communication system allowed trouble shooting and support of the weigh station system via phone lines.
4. Compiled a computer data base and developed the software for bar code scanning and information storage.

This report details the results of research in the above areas, conducted between December 1, 1986 and June 30, 1989. The research effort was a joint program conducted by the Arkansas State Highway and Transportation Department, the Center for Robotics and Automation, University of Arkansas, Fayetteville in cooperation with the Federal Highway Administration.

SUMMARY OF TECHNICAL FINDINGS

In order to accomplish the goals of this project several areas of research had to be conducted. Each area of research is discussed in detail below.

Emblem Materials

Investigation into materials which would be both durable and accurate under adverse conditions were required in order to get the quality of bar code necessary for this project. A lamina and retro-reflective material that would last the estimated life of the vehicle needed to be identified. Research included identifying plastic materials that could be used as protective sheeting for the bar codes. Tedlar and the Acryalr lamina were subjected to 2,400 hours of QUV testing (roughly the equivalent of 6½ years of outdoor exposure). The lamina showed no deterioration resulting from the weatherization testing. Tedlar was selected as the final lamina material.

Hardware

The selection of adequate hardware was most important to the success of this project. The whole concept is invalid without accurate scanning abilities and data collection capabilities.

Scanner

The bar code scanner used in the project was a Lazer-Data Model 624 A.V.I. laser scanner. This scanner was chosen because it is packaged in a weather proof housing designed for AVI use and has serial communication ability. The scanner comes with two serial communication ports. The first is a command port which allows the operator to communicate with the scanner to program it and perform equipment checks. The second port is used to send the ASCII characters representing the scanned bar code to the host computer. Because the scanner could be programmed from the serial port, reprogramming

of the scanner and diagnosis of scanner problems were performed from Fayetteville using Carbon Copy Plus software.

Modifications to the signal processing electronics and the housing unit of the laser bar code reader, coupled with new materials used for the bar code emblem, produced an accurate testing system. System testing was conducted in one of the Robotics Center's laboratories, which was modified to simulate a truck passing a weigh station. A rotating motion machine was designed and built to move codes past the scanner at speeds of up to 60 MPH (Figure 1). Banks of high intensity lights were installed to simulate sunlight. Reading at a distance of 84 inches (7 ft.) and at a speed of 35 MPH, under (simulated) bright sunlight, the system successfully read 99.63% of the time (30 no reads in 8,888 tries).

The scanner was designed to be used where a wide scan angle or long reading range was required. This scanner can work with a host computer system or operate in a stand alone mode where the scanner will operate lane gates or lights.

A 2mW helium-neon laser is used to generate the laser beam necessary to retrieve data form bar coded labels. The beam is reflected off of an eight sided rotating mirror that produces a 60° scan angle. The energy of the laser beam then exits the scanner and is reflected off the bar coded labels and back to the scanner. A photodiode picks up the reflected energy and the signal is then processed by analog circuitry and converted to a digital signal. The digital signal is then decoded by the digital portion of the scanner and converted to ASCII format. This ASCII string is then transmitted out the serial port to the host computer.

The bar code scanner is set up to read retro reflective vehicle labels. These labels are designed to return a high gain signal to the scanner. When the dot of the laser beam passes over the bar code, the reflective energy of the beam is measured by a photodiode. When the beam passes over a bar on the bar code, a low amount of the returned energy is measured. Where the beam passes over where there is no bar, a high amount of returned energy is measured. The returned energy is then processed by the scanner and transmitted to the host computer. (See Figure 2a and 2b).

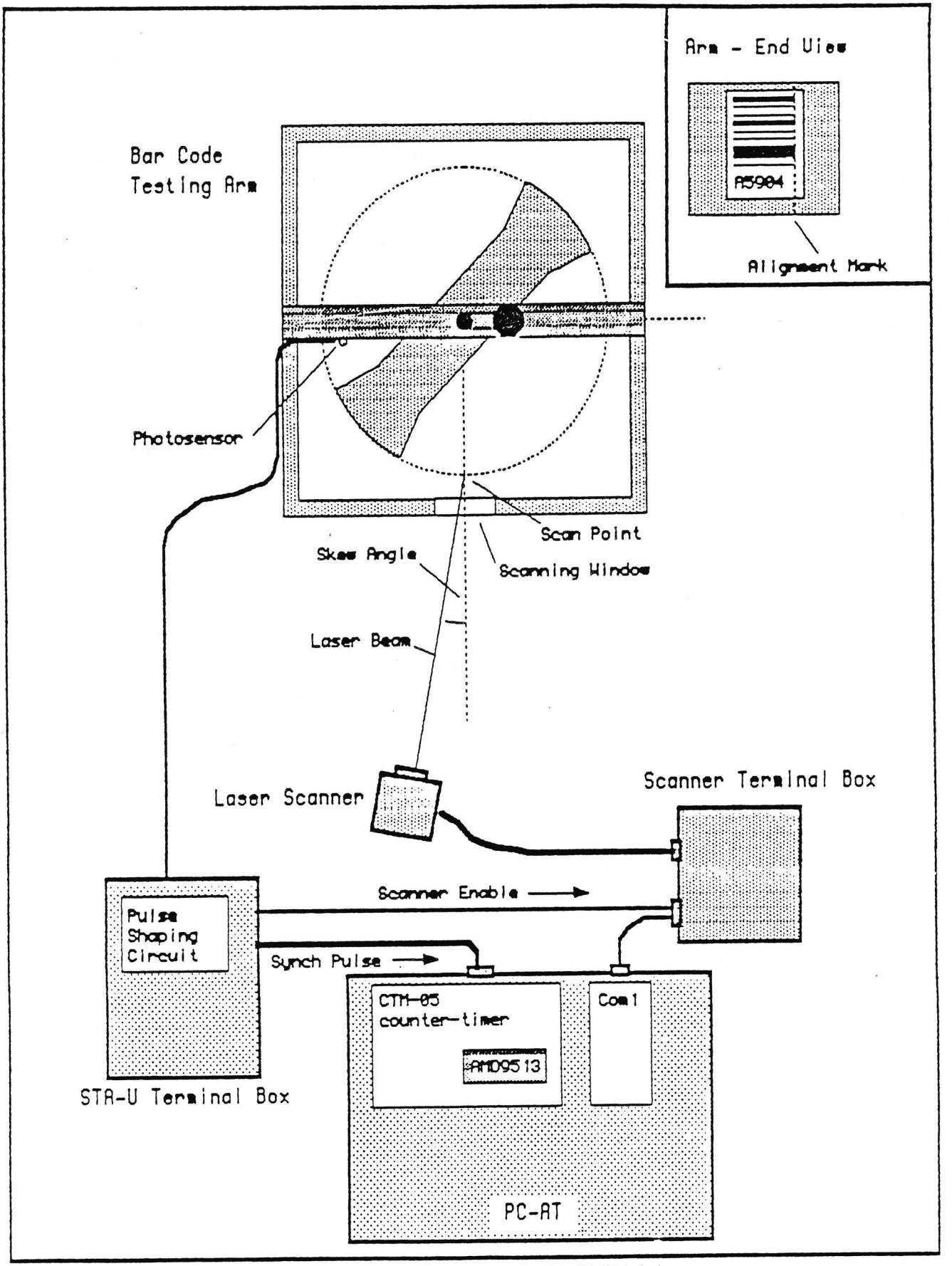
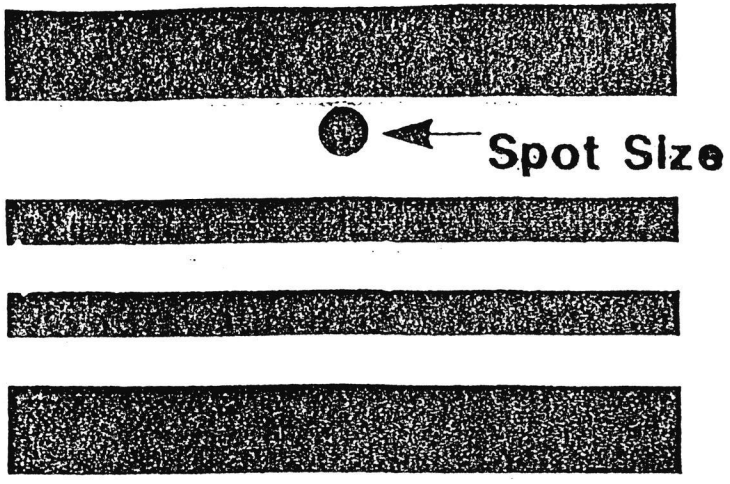

Arm - End View
A5984
Alignment Mark
Bar Code
Testing Arm
Photosensor
Scan Point
Skew Angle
Scanning Window
Laser Beam
Laser Scanner
Scanner Terminal Box
Scanner Enable
Pulse
Shaping
Circuit
Synch Pulse
CTM-05
counter-timer
AM9513
Com1
STA-U Terminal Box
PC-AT

Figure 1



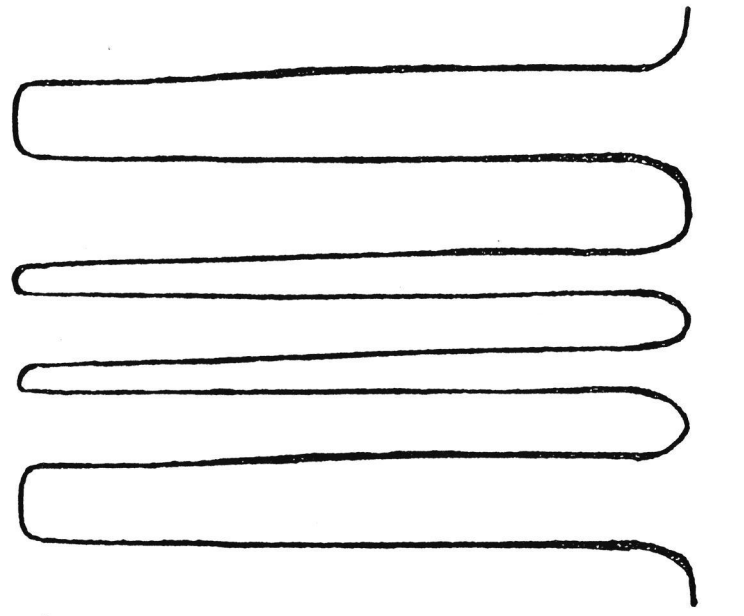


Figure 2a

Size of spot of focused light
is appropriate for a good read.

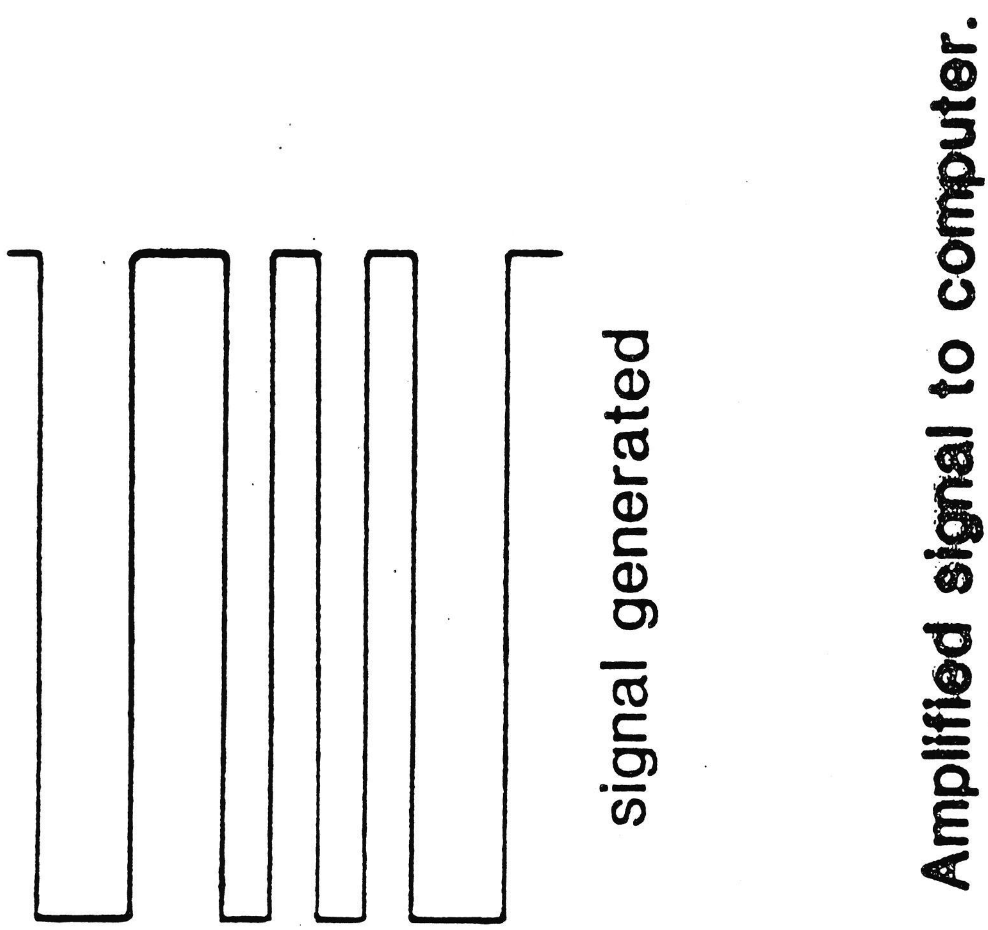


Figure 2b

Microcomputer System

A microcomputer system for retrieving and storing information was specified and purchased. The system is capable of storing the information necessary to process 467,000 vehicles, before having to access a larger host data base. A personal computer was chosen as the base for the system because of the low cost and the amount of development software available. Another reason for using a personal computer was the ability to upgrade the hardware as faster and more powerful computers become available. When the original system was installed in Springdale, software upgrades and monitoring were easily accomplished because of the closeness of the weigh station to the University of Arkansas Campus at Fayetteville. In Phase II of the project, the decision was made to move the system to West Memphis because of the large volume of truck traffic; Lehi Weigh Station at West Memphis is the busiest weigh station in the state. The system was still able to handle the upgrades and changes resulting from the dramatic increase in the input data of the new location.

Additional Hardware Requirements

The scanner has an RS-232 serial communications port. The distance involved between the scanner and host computer, required additional communication equipment was required for the system to operate.

The distance from the host computer to the scanner is about 1000 feet, the RS-232 is unable to transmit this distance, so RE-422 was chosen as the communication protocol. Also the RS-422 has better noise immunity and the protocol allows the cable to run at least 1000 feet. The scanner itself was unable to support RS-422 so a circuit was designed to convert the RS-232 signal to RS-422. This circuit was installed on the scanner side of the communication cable. A similar circuit which converted RS-422 back to RS-232 was installed on the PC side of the cable. The scanner converter is powered by the scanner while the circuit at the PC side of the system uses the PC power supply.

During Phase II the following list of minimum requirements for hardware was determined:

1. An IBM PC-AT compatible micro-computer with VGA monitor, and 80-MB hard disk, 1.2-MB floppy, 1.44 MB floppy, 2 serial ports, and a parallel port.
2. An internal 2400 baud modem.
3. Bar code scanner with serial communications.

Software

The use of "off the shelf" software was utilized whenever possible and then customized for the bar code applications. To tie the hardware together into a working scanning system, a number of programs had to be written to interact with the data base and scanner. The following is a list of the minimum software requirements:

1. Communications software to communicate with the scanner.
2. Data input screen using a windowing system.
3. A real time data retrieval code that displays bar code information on a screen.
4. A menu to tie the system together.
5. Report generating software.

Communication Software

The software used to communicate with the scanner is an interrupt driven function written in assembler. A 1K circular buffer stored the information sent from the scanner. This information is the character representation of the bar code plus a carriage return as a post amble character. Tests performed in the lab determined that there were no overflow problems with the 1K buffer running at 9600 baud or with processing information from vehicles with speeds up to 30 MPH. The source code for these functions is provided in Appendix I.

Data Input Software

The data input program was written using C Windows Toolkit by Magna Carta Software. This software helped the programmer set up windowing and data entry screens. It provided functions to allow only certain types of data to be entered into the data entry

fields and functions for data entry fields editing. The Toolkit was used in the data entry program, the menus, and the report generating programs.

Bar Code Retrieval Software

The current version of the bar code retrieval software was designed to display the bar code information to the screen. This was done because the system was not to be connected to the lane control lights or the weigh-in-motion scale.

Determining Real Time - The plans for the weigh station required that the over head lane control lights be a minimum of 200 feet from the weigh in motion scale. The scanner was placed at the same location as the weigh in motion in scale (Figure 3). This meant that the program would have at least the time required for the vehicle to travel 200 feet to determine if the vehicle was legal or not. Table 1 list shows how many seconds a vehicle takes to travel 200 feet at different speeds.

MPH	SECONDS
25	5.45
30	4.54
35	3.90
40	3.41
45	3.03
50	2.73
55	2.48
60	2.27

Table 1.

SYSTEM OPERATION

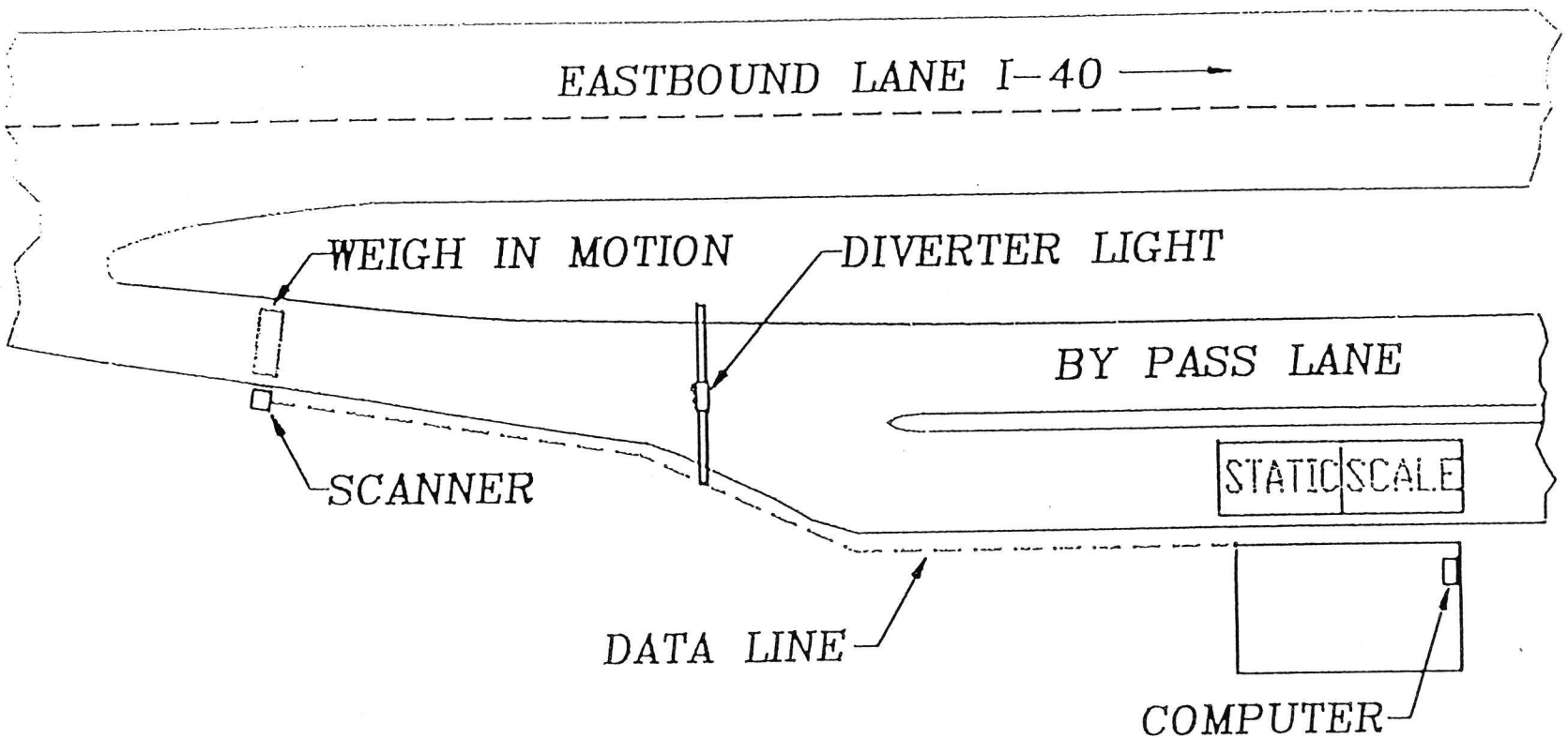


Figure 3

The total system is limited by the weigh in motion scale. The scale cannot read reliably over 40 MPH necessitating the vehicles to slow down to at least 40 MPH or to enter the weigh station. This means the program has at least 3.41 seconds to do five major operations:

1. Scan and transmit bar code to the micro-computer.
2. Search database for bar code and retrieve the information.
3. Based on weight and information the database determines the vehicle's legality.
4. Determine if vehicle is legal based on information in database and weight.
5. Activate the lane control lights to tell driver either to bypass or pull over at weigh station.

Weight determination and the bar code scann was done in parallel on separate processors. The weigh-in-motion system and bar code scanner then send the information to the micro-computer via serial lines. The time required to do this is negligible compared to the disk access time. Defining the vehicles legality is a simple and quick comparison operation done by memory with out any I/O. Activation of the lane control lights was done with digital I/O boards and required little time. Searching of the database for the bar code required the longest time, about 2.27 seconds at 40 mph or two thirds of the time.

Btrive and B-Trees - To make this operation as fast as possible and to speed up development time, the decision was made to purchase a database record manager instead of writing one from scratch. Btrieve by Novell was chosen because of ease of use and low cost.

Btrieve is based on the B-tree file indexing system developed by R. Bayer and E. McCreight in 1971. B-trees allow for low retrieval costs because the tree is automatically balanced during the insertion and deletion operations. To determine the costs of retrieval in the system, a few terms must first be defined:

Node: The node of a B-tree consists of keys, pointers to other nodes, and data pointers. The keys are used in comparison routines to determine if the requested key is in the node or if the search should continue to another node. If the requested key is in the node, the pointer to the data is used to retrieve the data. If the key is not in the node, the search continues to a node on the next level.

Order: The order of a B-tree, d, is the Node size. The Node will have between d and 2d keys and data pointers and between d + 1 and 2d + 1 node pointers.

Height: The height of a B-tree, h, is the logical vertical height of the tree. The height of the tree can change only if there is a split at the root during a insert operation or a concatenation during a delete operation. The height also shows the maximum number of levels the search has to traverse to find a key.

The retrieval costs for B-trees have been shown to grow as the logarithm of the number of keys, n. The formula to determine the worst case cost of retrieval is:

$$h \leq \log_d ((n+1)/2).$$

The following table (Table 2) shows the upper bound of the number of nodes retrieved in the worst case for various node sizes and file sizes:

		File Size (records)				
		10^3	10^4	10^5	10^6	10^7
Node size	10	3	4	5	6	7
	50	2	3	3	4	4
	100	2	2	3	3	4
	150	2	2	3	3	4

Table 2.

As can be seen from Table 2 the number of nodes to be retrieved to find an entry do not increase significantly as the file size increases. As the number of disk accesses are reduced the faster the information can be found in the database and loaded into memory.

BTRIEVE allows multiple files per database and multiple keys per file. A B-tree is maintained for every key in a file. This means data entry into the data base is slow because every time an entry is marked in a file, a B-tree must be balanced for each key. This tells BTRIEVE where the keys are in a file and what type of keys exist. BTRIEVE requires a data description file for every file to be used in the database. This file is in ASCII and tells BTRIEVE how many, the location and type of keys in the file. The data description file also tells BTRIEVE the record length of the data structure and the page buffer size for the file. An example of one of the data structures used in the data retrieval program and its data description can be found in Appendix II.

A copy of the TSR calling code is provided in Appendix I. The source code for the data structures, defines of Btrieve operations, and strings for error codes used in the data base programs can be found in the file btrieve.h included in Appendix III.

Retrieval Software - The data base is made up of a number of files. These files contain related information and are all maintained by the Btrieve Record Manager. The following is a list of these files and a short description of each:

Truck.db	Main database file. Contains information on each truck in the system.
Daily.db	This file logs the vehicle as it goes past the scanner. The bar code, time, date, and location of the scan is stored.
Bad_Read.db	This file holds the entries made by the operator at the weigh station. If a bar code passes the scanner and does not read, the operator would enter the bar code manually. the bar code, time, date, and location of the entry is stored.
Comment.db	This file can be used to store any information about the truck that is not storable anywhere else. This file was used to store the weather at the time the bar code was applied to the vehicle.
Flag.db	This file can be used to flag the bar code if the vehicle must be stopped, e.g. if it was stolen, need to get a message to the driver, etc.. It contains a start date, and end date, and a reason for the flag.
Owner.db	This file contains the owner information such as address, phone, name, etc..
Manufact.db	This file contains the names of the vehicle makes.
Color.db	This file contains color names.

The relationship between these files is shown in Figure 4 on the following page.

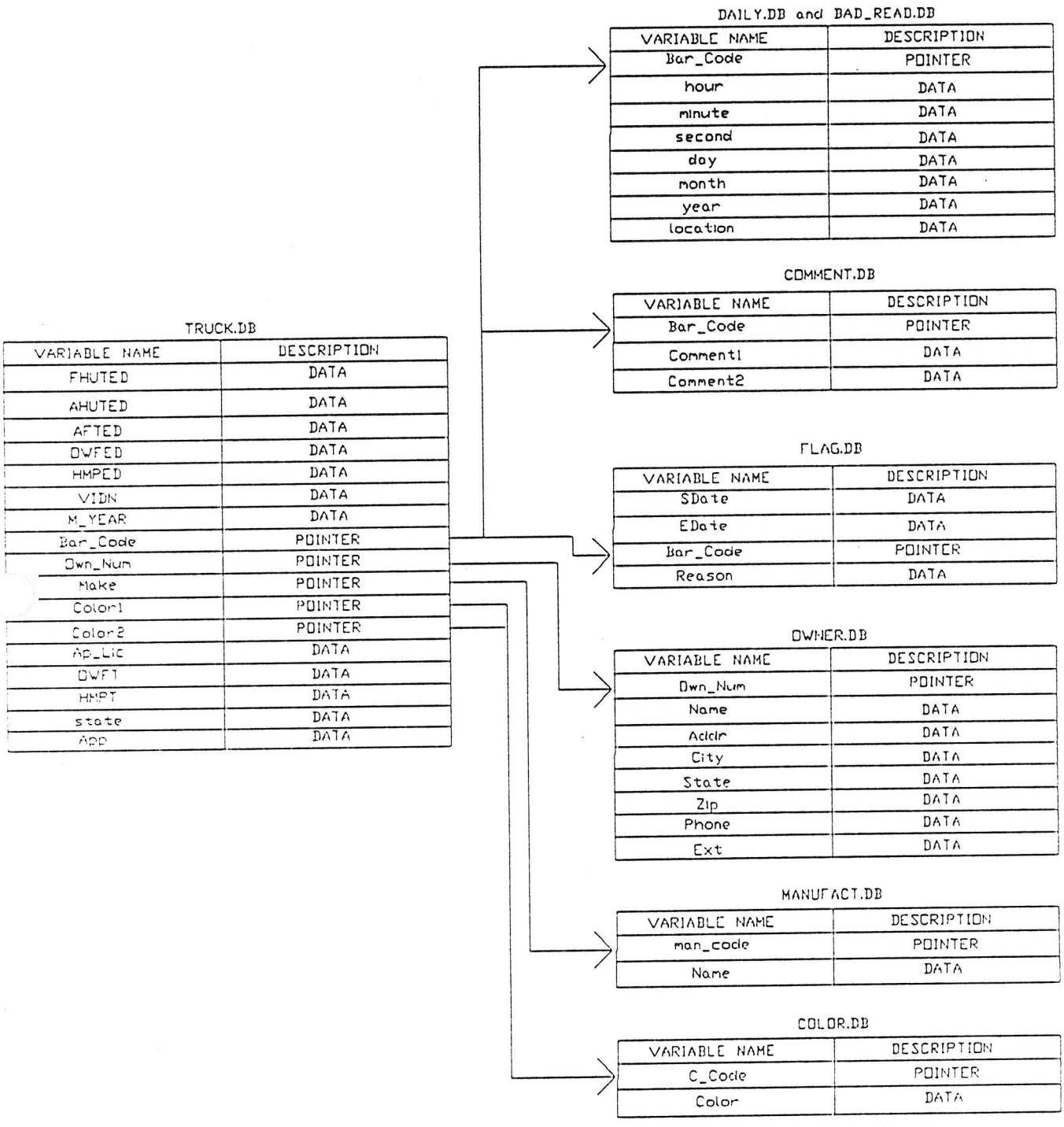

DAILY.DB and BAD_READ.DB
VARIABLE NAME
DESCRIPTION
Bar_Code
POINTER
hour
DATA
minute
DATA
second
DATA
day
DATA
month
DATA
year
DATA
location
DATA
COMMENT.DB
VARIABLE NAME
DESCRIPTION
Bar_Code
POINTER
Comment1
DATA
Comment2
DATA
TRUCK.DB
VARIABLE NAME
DESCRIPTION
FHUTED
DATA
AHUTED
DATA
AFTED
DATA
DWFED
DATA
HMPED
DATA
VIDN
DATA
M_YEAR
DATA
Bar_Code
POINTER
Own_Num
POINTER
Make
POINTER
Color1
POINTER
Color2
POINTER
AP_LIC
DATA
DWFT
DATA
HMPT
DATA
state
DATA
APP
DATA
FLAG.DB
VARIABLE NAME
DESCRIPTION
SDate
DATA
EDate
DATA
Bar_Code
POINTER
Reason
DATA
OWNER.DB
VARIABLE NAME
DESCRIPTION
Own_Num
POINTER
Name
DATA
Addr
DATA
City
DATA
State
DATA
Zip
DATA
Phone
DATA
Ext
DATA
MANUFACT.DB
VARIABLE NAME
DESCRIPTION
man_code
POINTER
Name
DATA
COLOR.DB
VARIABLE NAME
DESCRIPTION
C_Code
POINTER
Color
DATA

Figure 4

To retrieve the information in the database, the process followed these five steps:

1. Retrieve record from truck.db based on bar code received from scanner.
2. Retrieve record from owner.db based on the owner number in the record retrieved from truck.db.
3. Retrieve colors from color.db based on the color numbers in the record retrieved from truck.db.
4. Retrieve record from manufact.db based on the vehicle make number in the record retrieved from truck.db.
5. Finally, retrieved record from flag.db, if any, based on the bar code.

These steps load the required information into the related data structures in the memory. As discussed before, these are the steps that required the longest time to accomplish. To determine just how long these steps took verses the file size, the code which accomplished these operations was taken out of the code written for the project and a timing program was written around it. A copy of the flow chart and code for this program, loadcode.c, can be found in Appendix IV.

The program loadcode.c produced a data file every time 1000 new entries were made to the database and the timings were taken. Every data file has the following format:

```
"Number of Entries",40000
"Number of Reads",1000
"AB186",2175081,2175097,16
"AV753",2175103,2175130,27
                .
                .
                .
```

until 1000 scans were recorded.

The first line recorded the number of entries in the database. The second line was the number of random reads from the database and the rest of the data file lines were similar. These lines contained the bar code retrieved, the start time in hundredths of seconds, the end time in hundredths of seconds, and the differences of the two times.

These times were obtained by using the DOS interrupt 2C which returns the hours, minutes, seconds, and hundredths of seconds since midnight. The hundredth of seconds were not true hundredths since the system clock was only updated 18.2 times a second. This gave a true resolution of only 0.05495 seconds per clock tick.

Even though the numbers generated by the system clock were not true to the hundredths position, the calculation of the average retrieval time was used. Any final numbers derived form the data were truncated to the tenths position before being used in any decisions about the system.

After the load.c ran and 150,000 entries were entered into the database, the data files created by loadcode.c were run through another program, which calculated the maximum, minimum, and average retrieval times for each of the 1000 entries. Also the standard deviation and variance for each of the 1000 entries were calculated. The flow chart and source code for this program, avgstd.c, can be found in Appendix V.

There was a point, about 2 times per data file, where the ending time was less than the starting time when the data generated by loadcode.c was inspected. When the subtraction was made to determine the time difference, a very large number was returned because of the unassigned values. It was decided that this problem occurred at a point when the clock value rolled over, so these large numbers were not included in the calculations, Avgstd.c was written to watch for these numbers and to perform any operations without them.

The timing tests were run on a 20 Mhz 386 based system with a 180 Megabyte ESDI hard disk, 4 Megabytes of memory and a VGA color video. The system was run using MSDOS 4.01 so the hard disk could be set up as one logical hard disk and the database files could grow as needed.

The first set of timing runs were made with no disk cache and with Btrieve working in the normal mode. This mode did some error checking during the read and write operations. The graph in Figure 5 shows the average times and maximum times for the database as it grows. As can be seen from this graph, the maximum times were above five seconds and the size of the data base reached 140,000 entries, with the average time remaining below one second. This was determined to be too slow and faster access times were needed.

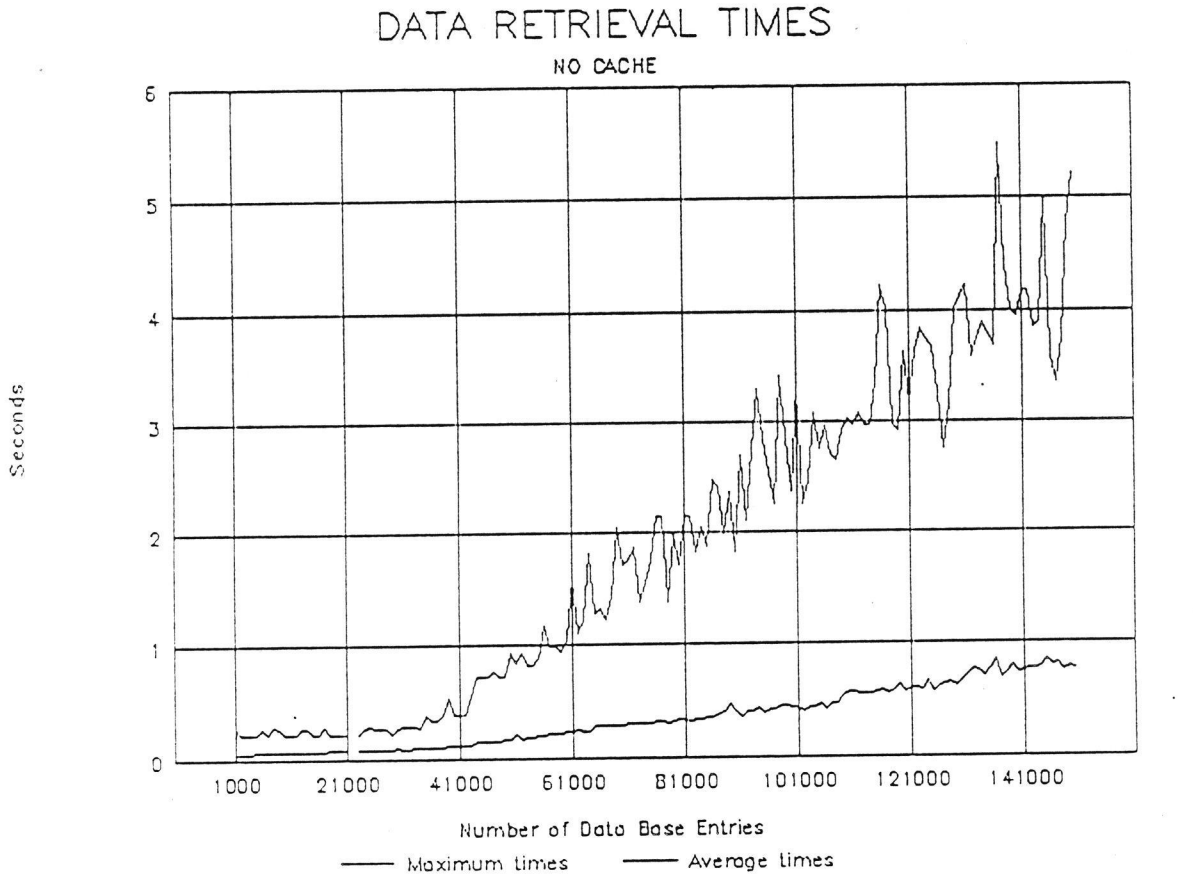

DATA RETRIEVAL TIMES
NO CACHE
Seconds
0
1
2
3
4
5
6
1000
21000
41000
61000
81000
101000
121000
141000
Number of Data Base Entries
Maximum times
Average times

Figure 5

To improve performance to an acceptable level, two things were done; the data base files were opened in the accelerated mode and a software hard disk cache was used. When the data base files were opened in an accelerated mode, no error checking took place during the read and write operations. The buffer was not written to disk until a record not in the buffer was requested by the program. This factor had little effect on read operations, but increased the time between disk write operations. The software cache, PC_CACHE included with PC-Tools Deluxe was used and set to read 4 sectors ahead.

The graph in Figure 6 shows the average and maximum times for this set of timing tests. The maximum access times were reduced to 1.9 seconds for the maximum number of records with the average access time being less than 0.8 seconds. To determine if the times would improve with a larger cache, a 10 sector look ahead cache was used. The graph in Figure 7 shows the results. There were no significant improvements from a 4 sector to a 10 sector look ahead cache.

The system delivered for Phase II contained the above data base design with a 4 sector look ahead cache. Further studies were conducted to see if even faster retrieval times were possible.

The approach taken this time was to create a key only file that consisted of only legal bar codes. The same timing tests were run on this file and the results showed an improvement over the system delivered. With no cache, the maximum retrieval time was around three seconds while the average was around .8 seconds. With a 4 sector cache, the maximum times were down around .3 seconds with an average around .1 second.

Database Integrity

Database integrity and recovery are major problems concerning database users. If a system crashed or program failure were to occur while the program was updating the database, the database would be in an inconsistent state. Invalid access paths could possibly be introduced which would make data retrieval impossible. In the case of multiple file databases, an update to the database acting on the more than one file, could leave one file updated while the remainder was not.

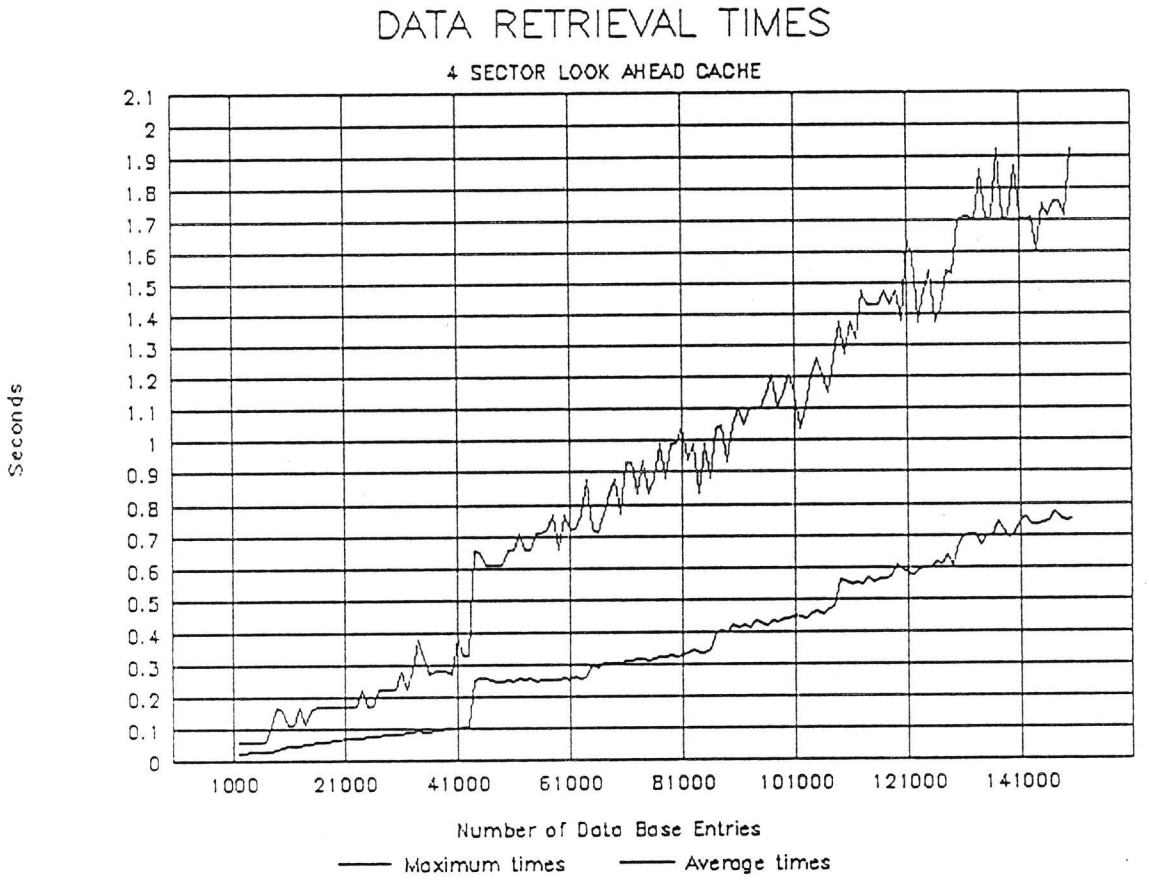

DATA RETRIEVAL TIMES
4 SECTOR LOOK AHEAD CACHE
Seconds
2.1
2
1.9
1.8
1.7
1.6
1.5
1.4
1.3
1.2
1.1
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
1000
21000
41000
61000
81000
101000
121000
141000
Number of Data Base Entries
Maximum times
Average times

Figure 6

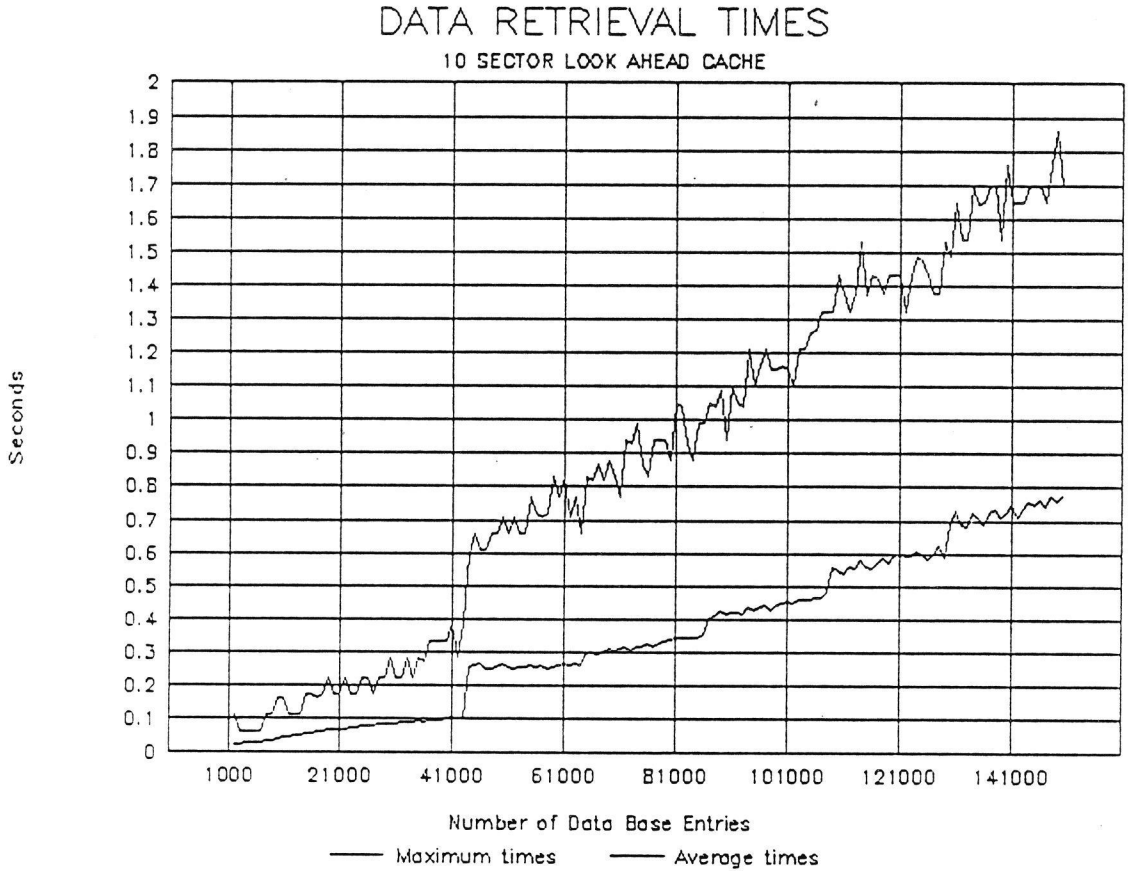

DATA RETRIEVAL TIMES
10 SECTOR LOOK AHEAD CACHE
Seconds
2
1.9
1.8
1.7
1.6
1.5
1.4
1.3
1.2
1.1
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
1000
21000
41000
61000
81000
101000
121000
141000
Number of Data Base Entries
Maximum times
Average times

Figure 7

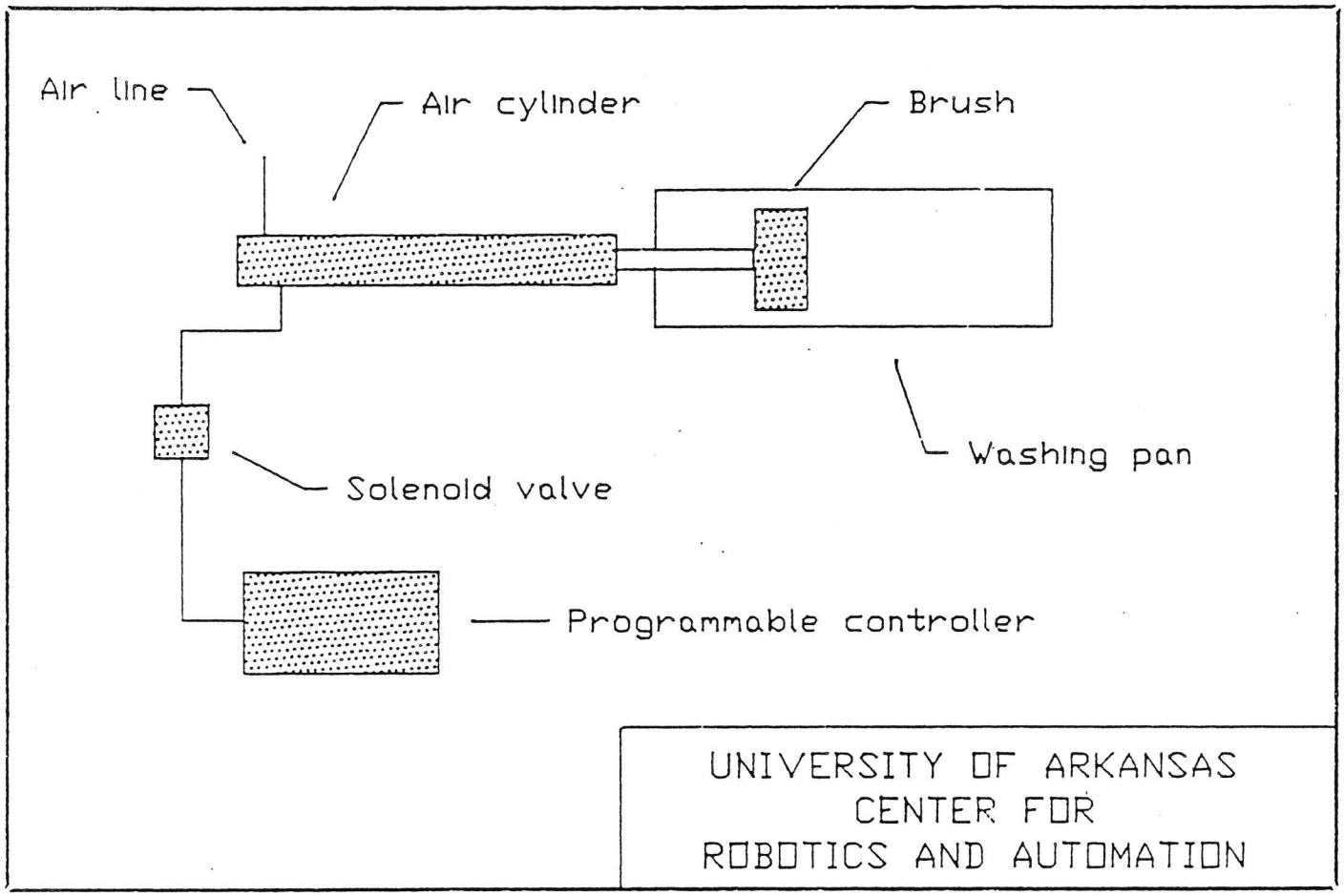


Diagram of Wash-Test Set-Up
Figure 9

After the system or program is restated, some attempt must be made to return the database to a consistent state. The usual method is in remove any incomplete updates to the database that were in progress when the systems or program failed. Then what to remove from the database is determined using a log file which is usually kept describing the action about to take place. This file is written to before any update is made to the data base. This form of database recovery is called transaction processing.

Transaction processing uses a logical unit of work called a transaction consisting of a number of database updates. All of the updates need to be executed for the database to be in a consistent state. Transaction processing guarantees that if a transaction is interrupted by a failure, all of the updates in the transaction would be undone. Btrieve uses two levels of transaction processing to keep the database in a consistent state.

Btrieve creates a pre-imaging file for each file in the database to protect individual Insert, Update, and Delete operations. Before Btrieve writes a change to a file page on the disk, a copy of the file page is written to a pre-image buffer. When the buffers become full or the operation is ended, Btrieve first writes the pre-image buffers to the disk and then the data base buffers.

If a system or program failure were to occur during the write operation to the pre-imaging file, the database file would not be corrupted. If a failure occurred during the write operation the database file, the file page would be copied from the pre-imaging file when the file was first opened after the failure.

The pre-imaging of operations only works on individual files. To obtain a higher level of data integrity, transactions may be used. Btrieve supplies two commands, Begin Transaction and End Transaction to bracket operations into a transaction. When a transaction is specified by these commands, all operations in the transaction must be completed or Btrieve will automatically remove all operations performed during the transaction. Even though transactions processing will protect the database from system or program failures, it will not protect the database from a media failure. Making backups on a regular basis is still the only method that will protect the database from a media failure.

TEST PROCEDURES

During previous studies numerous factors were found to influence the read rate of the bar code system. These variables can be grouped into one of four categories:

1. Factors effecting the laser scanner, i.e. power, scan rate, housing unit for the scanner, etc.
2. Factors effecting the bar code, i.e. size of the printed bar code (min. bar width), width of code, type of backing material, type of lamina, etc.
3. Factors concerning the environment, i.e. the intensity of sunlight present, angle of sunlight, rain, snow, fog, etc.
4. Factors affecting the truck with the applied bar code, i.e. the speed of the truck, distance from the scanner, amount of dirt on the code, etc.

In order to quantitatively evaluate the multiple factors affecting the system read rate, a scientifically designed experiment was conducted. Factors influencing the read rate were evaluated and priortized. The following variables were identified and tested: **Minimal Bar Width, Type of Lamina, Size of Code, Skew Placement of Emblem, Speed of "Truck", and Distance Between Scanner and Code.**

A rotating fixture was designed, built, and used to simulate various truck speeds. Figure 1 shows the configuration of the rotating device and its interface to the test computer.

A computer program was developed to record the speed of the rotating arm and accept inputs from the bar code reader. A photodetector sensor attached to the test device sent a pulse to the computer upon each successive revolution of the arm. Additionally, the sensor sent a scanner enable pulse to the bar code reader (signaling the scanner to begin reading). The interface of a AMD9513 counter/timer board with the personnel computer allowed the calculation of the speed of the rotating arm. Once the bar code reader received the enable pulse it signaled the test computer of a good read or a bad read. In this fashion various independent variables were tested as to their effect upon the dependant variable - percent good reads.

Readability Tests

Figure 8 graphically shows the effects of the variables which most effected the system, e.g. distance between scanner and code, speed of code, and minimum bar width.

The plots are based upon 62,000 test runs:

a) 15500 samples of lamina 1 - 60 mil codes
b) 15500 samples of lamina 2 - 50 mil codes
c) 15500 samples of lamina 1 - 60 mil codes
d) 15500 samples of lamina 2 - 50 mil codes

It should be noted that the original design of experiment called for the testing of three types of lamina and three sizes of bar codes e.g. 40, 50, and 60 mil (minimum bar width). Preliminary testing determined the read rate of the 40 mil code was below 70% at a distance of seven feet. Therefore the 40 mil code was rejected. Similarly, one of the three lamina materials chosen for the study did not pass the 1000 hour advanced weatherization test and was eliminated from further consideration.

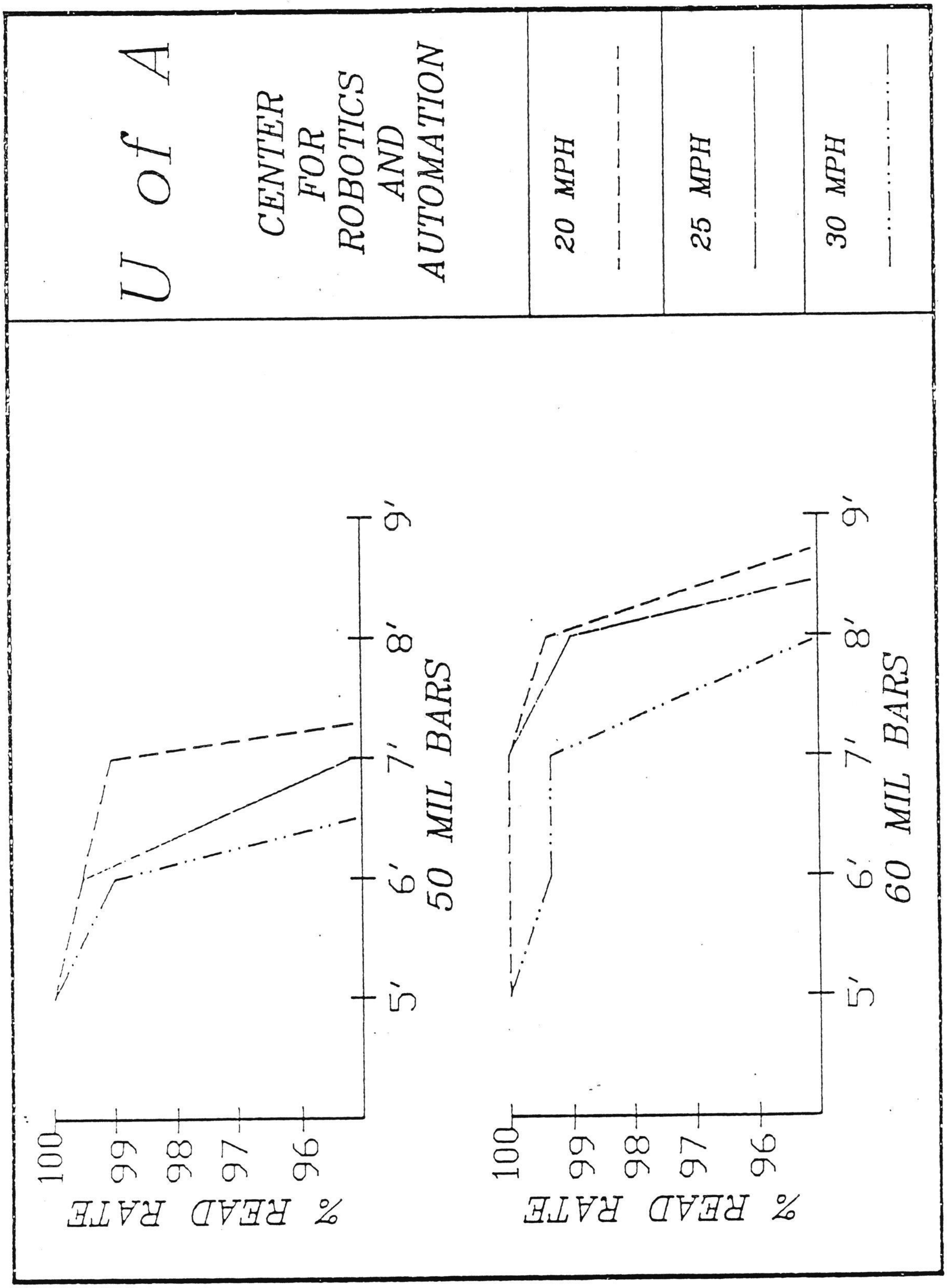
Special bar codes with no laminating material were printed and tested. The results of the test provided relevant data concerning the effects of laminating material vs. read rate.

Environmental Testing of Codes

The bar code emblems produced for this project were tested in an environmental testing chamber to determine the long term effects of weathering upon the various laminating materials.

Mr. George Green section head of the Chemistry Laboratory, Material and Research Division of Arkansas State Highway Department was responsible for the environmental testing. The machine in which the codes were tested is a QUV Accelerated Weathering Tester, which is made by Q-Panel Co. The testing method is the one recommended in ASTM G53. The test involves a continuous cycle of lights on or off and condensation on or off:

1. Four (4) hours of UV Lights ON, Condensation OFF, unit temperature at 60 deg. C.
2. Four (4) hours of Condensation ON, UV Lights OFF, unit temperature at 50 deg. C.
3. Repeat the cycle, continuously.



Effects of the Variables Affecting the System
Figure 8

Test codes were exposed at intervals of 250, 500, 750, and 1000 hours each. Upon removal from the QUV tester the codes were inspected for signs of material deterioration and tested for readability. As mentioned previously, one of the three test lamina did not successfully pass the 1000 test.

Wash-Testing of Codes

Figure 9 is a block diagram of the wash-test set-up designed and built to simulate the repeated washings commercial vehicles are subjected to. The wash test was comprised of placing the test bar code in a solution of commercial truck wash detergent and repeatedly activating an air cylinder with a brush attached to its end. All three test codes successfully passed 10,000 repetitions of this test with no visible damage.

TEST RESULTS

Effect of Minimum Bar Width vs. Readability

1) The read rate of the 50 mil bar codes was greater than 95% at distances less than seven (7) feet and speeds of up to twenty five miles per hour.

2) The read rate of the 60 mil bar codes was greater than 98% at distances less than eight feet and speeds up to twenty five miles per hour.

Effect of Speed vs. Readability (tested at 7 feet)

1) The 60 mil bars were readable at speeds up to 45 MPH (read rate above 97%)

2) The 50 mil bars were readable at speeds up to 20 MPH, however at higher speeds the read rate dropped rapidly.

Effect of Laminating Material on Readability

Tests on bar codes with different lamina indicated that the read rate is independent of the lamina type studied.

```
                        /* OWFT */
position=50
length=1
duplicates=y
modifiable=y
type=string
alternate=n
null=y
value=20
segment=n

                        /* HMPT */
position=51
length=1
duplicates=y
modifiable=y
type=string
alternate=n
null=y
value=20
segment=n
```

As can be seen from the data description file, Btrieve only cares about the keys themselves. Any other data stored in the record is saved in the data file on a byte by byte basis and it is up to the programmer to us the correct data structure to retrieve and save the data.

The Btrieve Record Manager is loaded into memory as a Terminate and Stay Resident (TSR) program and is called via a interrupt call using the calling sequence provided by Btrieve. The following is a sample call with a description of the parameters:

```
                        /* Make */
position=33
length=2
duplicates=y
modifiable=y
type=binary
alternate=n
null=y
value=0
segment=n

                        /* Color 1 */
position=35
length=2
duplicates=y
modifiable=y
type=binary
alternate=n
null=y
value=0
segment=n

                        /* Color 2 */
position=37
length=2
duplicates=y
modifiable=y
type=binary
alternate=n
null=y
value=0
segment=n

                        /* Ap_Lic */
position=39
length=11
duplicates=y
modifiable=y
type=string
alternate=n
null=y
value=20
segment=n
```

```
                        /* VIDN */
position=21
length=2
duplicates=y
modifiable=y
type=binary
alternate=n
null=y
value=0
segment=n

                        /* M_YEAR */
position=23
length=2
duplicates=y
modifiable=y
type=binary
alternate=n
null=y
value=0
segment=n

                        /* Own_Num */
position=25
length=2
duplicates=y
modifiable=y
type=binary
alternate=n
null=y
value=0
segment=n

                        /* Bar_Code */
position=27
length=6
duplicates=n
modifiable=n
type=string
alternate=n
null=n
segment=n
```

```
                        /* AHUTED */
position=5
length=4
duplicates=y
modifiable=y
type=binary
alternate=n
null=y
value=0
segment=n

                        /* AFTED */
position=9
length=4
duplicates=y
modifiable=y
type=binary
alternate=n
null=y
value=0
segment=n

                        /* OWFED */
position=13
length=4
duplicates=y
modifiable=y
type=binary
alternate=n
null=y
value=0
segment=n


                        /* HMPED
position=17
length=4
duplicates=y
modifiable=y
type=binary
alternate=n
null=y
value=0
segment=n
```

DATA STRUCTURE

```
struct truck_struct {
                    struct mth_yr_struct FHUTED;  /* Fed Highway Use Tax Expiration Date. */
                    struct mth_yr_struct AHUTED;  /* Arkansas Highway Use Tax Exp. Date. */
                    struct mth_yr_struct AFTED;   /* Arkansas Fuel Tax Exp. Date. */
                    struct mth_yr_struct OWFED;   /* Overweight Exp Date. */
                    struct mth_yr_struct HMPED;   /* Hazardous Material Date. */
                    int VIDN;          /* Owner Vehicle Id Number. */
                    int M_YEAR;        /* Model Year */
                    int Own_Num;       /* Owner Number, Pointer to Owner info */
                    char Bar_Code[6];  /* Bar code char. */
                    int Make;          /* Model number, Pointer to Make. */
                    int Color1;        /* Color1 number, Pointer to color info */
                    int Color2;        /* Color2 number, Pointer to color info */
                    char Ap_Lic[11];   /* Apportioned License */
                    char OWFT; /* Overweight Type */
                    char HMPT; /* Hazardous Mat. Type */
                    char state[3];     /* State were Registered. */
                    char App;          /* Is licensed Apportioned */
                  } Truck_Data, *t_ptr;
```

DESCRIPTION FILE

```
record=55          Record Length in bytes.
variable=n         No variable length records.
key=15             Number of keys.
page=4096          buffer page size in bytes.

                   /* FHUTED */
position=1         Position in data structure.
length=4           Length in bytes.
duplicates=y       Allows duplicate keys.
modifiable=y       Keys are modifiable after entered.
type=binary        Binary type data.
alternate=n        No alternate collating sequence.
null=y             There is a null character.
value=0            Null character value.
segment=n          Not a segmented key.
```

RESEARCH WORK COMPLETED

This project was conducted in accordance with the Procedural Manual For Transportation Research Projects and under the terms of the Basic Agreement. The proposed plan of work was to proceed in four different areas:

A. Weigh Station Modifications
B. Bar Code Label Studies
C. Scanner Research
D. System Review and Analysis

Following is a detailed description of the work completed with comments on the timing, problems encountered, and suggestions for additional study.

A. Weigh Station Modifications

1) Define system hardware requirements - Task 100% complete, performed as planned.

2) Write specifications for microcomputer and bid computer - Task 100% complete performed as planned (see Appendix VI).

3) Write software to support system integration between electronic weigh-in-motion scales, bar code scanner, and microcomputer data base - Task 100 % complete (see Appendix VII).

4) Installed microcomputer and trained weigh station personnel in its use - Task 100% complete.

B. Bar Code Label Studies

1) Design experiment to determine the significance of the differing variables associated with the reading of codes - Task 100% complete. As previously noted the significant variables identified were: minimum bar width, speed of code, and distance between code and scanner. Appendix VIII contains a copy of this experiment.

2) Find new lamina - Task 100% complete, but two week delay was experienced in the completion of this task due to difficulty locating a supplier of the lamina material.

3) Test lamina in AHTD UV tester - Task 100% complete.

4) Test lamina for resistance to multiple washings - Task 100 % complete. Test codes were subjected to 10,000 washings without physical deterioration.

5) Determine most effective bar code symbology for reading outdoors - Task 100% complete. The difficulties encountered around the complexity associated with developing a device which would safely generate speeds up to 35 MPH in a laboratory. There were also associated problems of "capturing" and analyzing this data in a timely manner. A suitable device was designed and built.

6) Determine material specifications for production of bar codes - Task 100% complete.

7) Write specifications for bar codes - Task 100% complete, specifications concerning the size, type of printing, and the symbology of codes are complete as shown in Figure 10.

C. Scanner Research

1) Develop a laser for system operation in direct sunlight - work completed as planned.

2) Write functional laser and bid specifications - Task 100 % complete, as planned (see Appendix IX).

3) Analyze readability of test codes to determine system readability - Task 100% complete. The first set of test bar code emblems were rejected due to poor quality. correcting the quality problem and reprinting the codes delayed this task by six weeks.

D. System Review and Analysis

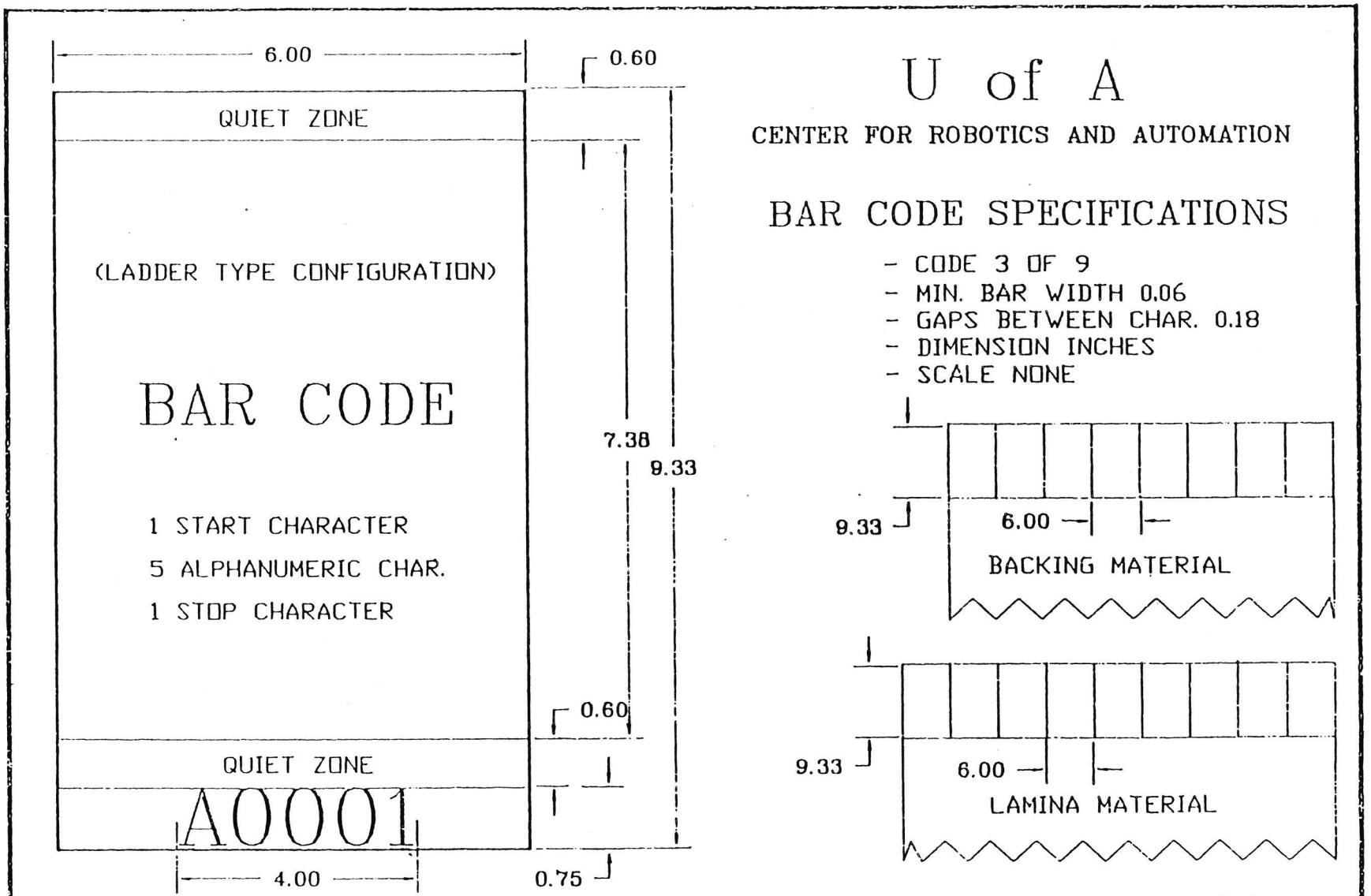
1) Comprehensive review and analysis of data collected - Task 100% complete, as planned.

2) Conduct a survey to determine the response of trucking companies to the concept of bar code vehicles identification - Task 30% complete, letters were formulated, and trucking firms identified. Due to the initiation of the controversial Weigh-Distance Tax during the time this task was to be completed, truck companies were not contacted to avoid having this system associated with the "weigh-distance" taxing.

3) Determine which variables most affect the operation of the system - Task 100% complete, extensive indoor testing was completed. Operating under the premise that a code maximized for readability indoors (under simulated outdoor conditions) would read correspondingly well outdoor.

4) Developed communications interface for scanner data transfer - Task 100% complete. A RS-422 interface was integrated to the scanner and computer for data transfer and scanner control.

5) Adoption of communication software package to allow control of the computer at the weigh station by the computer located at the Center.



Bar Code Specifications
Figure 10

CONCLUSION

In conclusion, the investigation of this research project in providing quantitative data concerning the readability of successful implemention of a bar code reading system of trucks at weigh stations was achieved by the following objectives:

1) Maximization of system hardware to achieve above a 95% first-pass read rate.

2) Research to provide a bar code emblem which will last the anticipated life of the truck.

3) A communication system for linking a remote site and main computer data base was adapted from commercially available software.

4) Participating trucks were bar coded and the corresponding codes were incorporated into a data base.

5) An interactive data base software package was developed for accessing the truck data base in an efficient manner.

The first objective was reached by making two modifications to the laser scanner: a) improving the optics of the scanner which processes the signal reflected back from the bar codes, and b) installing a "snout" onto the housing in which the laser was contained. These modifications had the effect of decreasing the signal to noise ratio encountered by the scanner in sunlight. The percent of good reads for the system (in bright sunlight) was increased from 75% to over 99%.

The second objective was reached by conducting research on new materials for the bar code emblems. A new base material was identified and tested. This retro-reflective material read well and displayed adequate durability characteristics. New laminating materials were identified and tested. Two of the laminating materials passed comprehensive advanced weatherization and wash tests. Tedlar was the final lamina selected. Testing showed the lamina did not effect the readability of the codes.

To meet objective three we chose Carbon Copy an "off the shelf" software. A special program was written to customize the software to allow remote communications with the bar code scanner. Not only did this method allow us to retrieve and analyze the data from the trucks passing through the Lehi weigh station in a timely manner, but also allowed us to download updated software as needed and to trouble shoot scanner problems and reprogram whenever necessary.

Bar codes were installed on trucks from Silica Transport, Inc., ABF, Pam Trucking, Carco Trucking, Soloman Brothers and U.P.S. to meet objective four. The corresponding information from the codes was incorporated into the data base. Examples of the type of data collected are included in Appendix X.

BTRIEVE, a low cost file indexing system which allows multiple files per database and multiple keys per file was used to meet objective five. The software package has database programming capabilities which allowed a very user friendly program to be written in order to simplify the training necessary for the weigh station personnel.

Appendix I

Btrieve Calling Code

```
/*                                                                      */
/*           MicroSoft C Version 3.0 interface to the                   */
/*              Btrieve Record Manager, version 4                       */
/*                                                                      */
/*      Note: if compiling for the "D" or "L" memory model, remove      */
/*            the comments surrounding the following definition.  */
/*                                                                      */
/*                                  LMODEL means 32-bit pointers in use
#define LMODEL 1
*/

/*         make sure this constant is not commented if
                compiling for MS windows
#define WINDOWS 1
*/

#define BTR_ERR      20                    /* record manager not started */
#define BTR_INT      0x7B               /* Btrieve interrupt vector */
#define BTR2_INT     0x2F            /* multi-user interrupt vector */
#define BTR_VECTOR   BTR_INT * 4           /* offset for interrupt */
#define BTR_OFFSET   0x33         /* Btrieve offset within segment */
#define VARIABLE_ID 0x6176      /* id for variable length records - 'va' */
#define _2FCODE      0xAB00         /* function code for int 2F to btrieve */

/* ProcId is used for communicating with the Multi Tasking Version of  */
/* Btrieve. It contains the process id returned from BMulti and should */
/* not be changed once it has been set.                                */
/*                                                                      */

static unsigned ProcId = 0;              /* initialize to no process id */
static char MULTI = 0;                  /* flag set to true if MultiUser */
static char VSet = 0;             /* flag set to true if checked version */

BTRV (OP, POS_BLK, DATA_BUF, DATA_LEN, KEY_BUF, KEY_NUM)
  int  OP;
  char POS_BLK[];
  char DATA_BUF[];
  int  *DATA_LEN;
  char KEY_BUF[];
  int  KEY_NUM;

{
struct REGVAL { int AX, BX, CX, DX, SI, DI, CY; } REGS;

struct SEGREG { short ES, CS, SS, DS; } SREGS;

struct BTRIEVE_PARMS       /* structure passed to Btrieve Record Manager */
 {
   char *BUF_OFFSET;                       /* callers data buffer offset */
#ifndef LMODEL
   int   BUF_SEG;                         /* callers data buffer segment */
#endif
```

```
  REGS.AX = 0x3500 + BTR_INT;
#ifdef WINDOWS
  int86 (0x21, &REGS, &REGS);          /* check for btrieve being loaded */
#else
  int86x (0x21, &REGS, &REGS, &SREGS);
#endif
  if (REGS.BX != BTR_OFFSET)
     return (BTR_ERR);
 )

/*  Read segment registers and initialize segment part of addresses to     */
/*  user's data segment.                                                  */
/*                                                                        */

segread (&SREGS);
#ifndef LMODEL
XDATA.BUF_SEG = XDATA.CUR_SEG = XDATA.FCB_SEG =
  XDATA.KEY_SEG = XDATA.STAT_SEG = SREGS.SS;
#endif

/*                                                                        */
/*  Move user parameters to XDATA, the block where Btrieve expects them.*/
/*                                                                        */

XDATA.FUNCTION     = OP;
XDATA.STAT_OFFSET = &STAT;
XDATA.FCB_OFFSET   = POS_BLK;
XDATA.CUR_OFFSET   = POS_BLK + 38;
XDATA.BUF_OFFSET   = DATA_BUF;
XDATA.BUF_LEN      = *DATA_LEN;
XDATA.KEY_OFFSET   = KEY_BUF;
XDATA.KEY_LENGTH   = 255;          /* use max since we don't know */
XDATA.KEY_NUMBER   = KEY_NUM;
XDATA.XFACE_ID     = VARIABLE_ID;

/*                                                                        */
/*  Make call to the Btrieve Record Manager.                              */
/*                                                                        */

REGS.DX = (int) &XDATA;      /* parameter block is expected to be in DX */
SREGS.DS = SREGS.SS;
if (!MULTI)
#ifdef WINDOWS                       /* if bmulti not loaded, call Btrieve */
  int86 (BTR_INT, &REGS, &REGS);
#else
  int86x (BTR_INT, &REGS, &REGS, &SREGS);
#endif
else
 (                                                        /* call bmulti */
  while (1)
   (
    REGS.AX = 1;             /*  assume no proc id obtained yet */
```

```
   int    BUF_LEN;                          /* length of data buffer */
   char *CUR_OFFSET;                       /* user position block offset */
#ifndef LMODEL
   int    CUR_SEG;                         /* user position block segment */
#endif
   char *FCB_OFFSET;                          /* offset of disk FCB */
#ifndef LMODEL
   int    FCB_SEG;                         /* segment of disk FCB */
#endif
   int    FUNCTION;                          /* requested function */
   char *KEY_OFFSET;                       /* offset of user's key buffer */
#ifndef LMODEL
   int    KEY_SEG;                         /* segment of user's key buffer */
#endif
   char  KEY_LENGTH;                       /* length of user's key buffer */
   char  KEY_NUMBER;                       /* key of reference for request */
   int   *STAT_OFFSET;                        /* offset of status word */
#ifndef LMODEL
   int    STAT_SEG;                        /* segment of status word */
#endif
   int    XFACE_ID;                          /* language identifier */
 ) XDATA;

int STAT = 0;                              /* status of Btrieve call */

/*                                                                      */
/*  Check to see that the Btrieve Record Manager has been started. */
/*                                                                      */

if (!VSet)                      /* if we don't know version of Btrieve yet */
 {
  VSet = 1;
  REGS.AX = 0x3000;                                 /* check dos version */
#ifdef WINDOWS
  int86 (0x21, &REGS, &REGS);
#else
  int86x (0x21, &REGS, &REGS, &SREGS);
#endif
  if ((REGS.AX & 0x00FF) >= 3)              /* if DOS version 3 or later */
   {
    REGS.AX = _2FCODE;
#ifdef WINDOWS
    int86 (BTR2_INT, &REGS, &REGS);              /* is bmulti loaded? */
#else
    int86x (BTR2_INT, &REGS, &REGS, &SREGS);
#endif
    MULTI = ((REGS.AX & 0xFF) == 'M');  /* if al is M, bmulti is loaded */
   }
 }

if (!MULTI)
 {                                          /* if bmulti not loaded */
```

```
    if ((REGS.BX = ProcId) != 0)          /* if we have a proc id */
      REGS.AX = 2;                                  /* tell bmulti that */
    REGS.AX += _2FCODE;
#ifdef WINDOWS
    int86 (BTR2_INT, &REGS, &REGS);                          /* call bmulti */
#else
    int86x (BTR2_INT, &REGS, &REGS, &SREGS);
#endif
    if ((REGS.AX & 0x00FF) == 0) break;         /* if call was processed */
#ifndef WINDOWS                              /* by bmulti, leave loop */
    REGS.AX = 0x0200;         /* if multilink advanced is loaded, it will */
    int86x (0x7F, &REGS, &REGS, &SREGS);     /* it will switch processes */
#endif
   )
  if (ProcId == 0) ProcId = REGS.BX;
 )

*DATA_LEN = XDATA.BUF_LEN;
return (STAT);                                          /* return status */
)
```

Appendix II

Data Structure

```
BT_STATUS = BTRV(GET_NEXT,POS_BLOCKS[COLOR],c_ptr,sizeof(Color_Data),&KB_ptr->Key_int,0);
```

BT_STATUS:	Return status code from BTRV call. A non-zero return value means an error occurred during processing of the BTRV call.
GET_NEXT:	A sample operation. A list of all operations can be found in btrieve.h.
POS_BLOCKS[]:	Used by Btrieve to contain positioning pointers for each file.
COLOR:	Defined in btrieve.h as 5.
c_ptr:	Pointer to data structure to store color information.
sizeof():	Size of data structure were color data is stored.
&KB_ptr->Key_int:	Pointer to area holding key value.
0:	Key number for operation.

Appendix III

BTRIEVE.H Source Code

```
/*************************************************************************/
/* Filename:     btrieve.h                Programmer: Gary Woodbridge    */
/* Description:  This file is to be used with AHTD software for reading and */
/*               scanning bar codes.  This file provides global variables  */
/*              and data structures used in retreiving data from database  */
/*              Requires BTRIEVE driver.                                   */
/*************************************************************************/

#include "mscxbtrv.c"           /* BTRIEVE code for Microsoft C. */
#include <string.h>
#include <stdlib.h>
#include <dos.h>
#include <ctype.h>

              /* Function Prototypes. */
int FILE_OPEN(); /* Opens all database files. */
int FILE_CLOSE();       /* Closes all database files. */

char *Error_Code();     /* Returns Error messege base on current status. */

  /* The following are BTRIEVE operation codes */

#define OPEN            0       /* Open file */
#define CLOSE           1       /* Close file */
#define INSERT          2       /* Insert new record into file */
#define UPDATE          3       /* Change current record in file */
#define DELETE_DB       4       /* Delete record in file */
#define GET_EQ          5       /* Get record equal to key */
#define GET_NEXT        6       /* Get next record from file */
#define GET_PREV        7       /* Get previous record from file */
#define GET_GRET        8       /* Get record greater than key */
#define GET_GE          9       /* Get record greater than or equal to key */
#define GET_LT          10      /* Get record less than key */
#define GET_LE          11      /* Get record less than or equal to key */
#define GET_FIRST       12      /* Get first record in file */
#define GET_LAST 13     /* Get last record in file */
#define CREATE          14      /* Create file */
#define STAT            15      /* Retrieves the information about file */
#define EXTEND          16      /* Devides file between two logical drives */
#define SET_DIR         17      /* Change current directory */
#define GET_DIR         18      /* Get current directory */
#define BEGIN_TRAN      19      /* Begin transaction section of code */
#define END_TRAN        20      /* End transaction section of code */
#define ABORT_TRAN      21      /* Abort transaction section of code */
#define GET_POS         22      /* Get current file position */
#define GET_DIRECT      23      /* Get record at current file position */
#define STEP_DIRECT     24      /* Get next record at file position */
#define STOP            25      /* Remove Record Manager from memory */
#define VERSION         26      /* Return Version number */
#define UNLOCK          27      /* Unlock record previously locked */
#define RESET           28      /* Release all resources held by station */
#define SET_OWN         29      /* Set owner of file */
```

```
#define CLEAR_OWN       30      /* Clear owner of file */

                /* The following defines set filenames to buffer arrays */
#define TRUCK           0       /* Truck.db */
#define FLAG            1       /* Flag.db */
#define STATE           2       /* State.db */
#define DAILY           3       /* Daily.db */
#define MANUFACT        4       /* Manufact.db */
#define COLOR           5       /* Color.db */
#define OWNER           6       /* Owner.db */
#define MONTH           7       /* Month.db */
#define BAD_READ        8       /* Bad_read.db */
#define COMMENT         9       /* Comment.db */

        /* The following are error strings for btrieve based on return status */
char ERROR_CODES[99][50] = {" 1: Invalid Operation",
                            " 2: I/O Error",
                            " 3: File Not Open",
                            " 4: Key Not Found",
                            " 5: Duplicate Key Value",
                            " 6: Invalid Key Number",
                            " 7: Different Key Number",
                            " 8: Invalid Positioning",
                            " 9: End of File",
                            "10: Modifiable Key Value Error",
                            "11: Invalid File Name",
                            "12: File not Found",
                            "13: Extended File Error",
                            "14: Pre-Image Open Error",
                            "15: Pre-Image I/O Error",
                            "16: Expansion Error",
                            "17: Close Error",
                            "18: Disk Full",
                            "19: Unrecoverable Error",
                            "20: Record Manager Inactive",
                            "21: Key Buffer Too Short",
                            "22: Data Buffer Length Error",
                            "23: Position Block Length",
                            "24: Page Size",
                            "25: Create I/O Error",
                            "26: Number of Keys",
                            "27: Invalid Key Position",
                            "28: Invalid Record Length",
                            "29: Invalid Key Length",
                            "30: Not a Btrieve File Name",
                            "31: File Already Extended",
                            "32: Extend I/O Error",
                            "33: ",
                            "34: Invalid Extension Name",
                            "35: Directory Error",
                            "36: Transaction Error",
                            "37: Transaction is Active",
```

```
"38: Transaction Control File I/O Error",
"39: End/Abort Transaction Error",
"40: Transaction Max Files",
"41: Operation Not Allowed",
"42: Incomplete Accelerated Access",
"43: Invalid Data Record Address",
"44: Null Key Path",
"45: Inconsistent Key Flags",
"46: Access Denied",
"47: Maximum Open Files",
"48: Invalid Alternate Sequence Definition",
"49: Key Type Error",
"50: Owner Already Set",
"51: Invalid Owner",
"52: Error Writing Cache",
"53: Invalid Interface",
"54: Variable Page Error",
"55: Autoincrement Error",
"56: Incomplete Index",
"57: Expanded Memory Error",
"58: Compression Buffer Too Short",
"59: File Already Exists",
"60: ",
"61: ",
"62: ",
"63: ",
"64: ",
"65: ",
"66: ",
"67: ",
"68: ",
"69: ",
"70: ",
"71: ",
"72: ",
"73: ",
"74: ",
"75: ",
"76: ",
"77: ",
"78: ",
"79: ",
"80: Conflict",
"81: Lock Error",
"82: Lost Position",
"83: Read Outside Transaction",
"84: Record in Use",
"85: File in Use",
"86: File Table Full",
"87: Handle Table Full",
"88: Incompatible Open Mode Error",
"89: Name Error",
```

```
                        "90: Device Full",
                        "91: Server Error",
                        "92: Transaction Full",
                        "93: Incompatible Lock Type",
                        "94: Permission Error",
                        "95: ",
                        "96: ",
                        "97: ",
                        "98: ",
                        "99: Demo Error");

char POS_BLOCKS[10][128],  /* Position blocks used by Btrieve, 1 per file. */
     B_CODE[7],            /* Global Bar Code variable. */
     color1[21], color2[21],  /* Global strings for colors. */
     datestring[9], monthstring[4], yearstring[6], /* strings used to hold date, abreavated months and
years. */
     *slash_ptr;  /* Global char pointer. */

int BUF_LENGTH[10],  /* Buffer length for each file. */
    BT_STATUS,       /* Status returned by Btrieve call. */
    correct=0,          /* used for correct response to questions. */
    quit=0;          /* Used to exit loops. */

struct mth_yr_struct {
                        int Month;       /* Month 1 - 12 */
                        int Year; /* Year 1980 - */
                      };


                      /* Data Structure for truck.db file. */

struct truck_struct {
                       struct mth_yr_struct FHUTED;  /* Fed Highway Use Tax Experation Date. */
                       struct mth_yr_struct AHUTED;  /* Arkansas Highway Use Tax Exp. Date. */
                       struct mth_yr_struct AFTED;   /* Arkansas Fuel Tax Exp. Date. */
                       struct mth_yr_struct OWFED;   /* Overweight Exp Date. */
                       struct mth_yr_struct HMPED;   /* Hazardous Material Date. */
                       int VIDN;          /* Owner Vehicle Id Number. */
                       int M_YEAR;        /* Model Year */
                       int Own_Num;       /* Owner Number, Pointer to Owner info */
                       char Bar_Code[6];  /* Bar code char. */
                       int Make;          /* Model number, Pointer to Make. */
                       int Color1;        /* Color1 number, Pointer to color info */
                       int Color2;        /* Color2 number, Pointer to color info */
                       char Ap_Lic[11];   /* Apportioned License */
                       char OWFT; /* Overweight Type */
                       char HMPT; /* Hazordus Mat. Type */
                       char state[3];     /* State were Registered. */
                       char App;          /* Is licensed Apportioned */
                     } Truck_Data, *t_ptr;

                     /* Data Structure for Daily.db */
```

```
struct daily_struct {
                      char Bar_Code[6];          /* Bar Code */
                      unsigned char hour;        /* Time the vehicle went past reader. */
                      unsigned char minute;
                      unsigned char second;
                      unsigned char day; /* Date the vehicle went past reader. */
                      unsigned char month;
                      unsigned int year;
                      int location;              /* Location of scanner. */
                    } Daily_Data, *d_ptr, Bad_Read_Data, *br_ptr;


                    /* Data Structure for Flag.db */

struct flag_struct {
                      char SDate[9];      /* Start Date */
                      char EDate[9];      /* End Date */
                      char Bar_Code[6];   /* Bar Code */
                      char Reason[51];    /* Description of flag */
                    } Flag_Data, *f_ptr;

                    /* Data Structure for Manufact.db */

struct manufact_struct {
                         int man_code;           /* Code for Manufacture */
                         char Name[31];          /* Make name. */
                       } Man_Data, *m_ptr;


                       /* Data Structure for Color.db */

struct color_struct {
                      int C_Code;        /* Color Code */
                      char Color[21];    /* String for color */
                    } Color_Data, *c_ptr;


                    /* Data Structure for Owner.db */

struct owner_struct {
                      int Own_Num;       /* Owner Number */
                      char Name[21];     /* Name of Owner */

                                /* Provide for 2 address. */
                      char Addr[2][31];  /* Street address */
                      char City[2][21];
                      char State[2][3];
                      char Zip[2][11];
                      char Phone[2][15];
                      char Ext[2][5];
                    } Owner_Data, *o_ptr;
```

```
                        /* Data Structure for Comment.db */
struct comment_struct {
                        char Bar_Code[6];       /* Bar Code */
                        char Comment1[81];      /* Two Lines of comments */
                        char Comment2[81];
                      } Comment_Data, *com_ptr;

                      /* Key holder for database calls */
union key_buf_union {
                      char Key_Char[11];
                      int Key_int;
                    } KEY_BUF, *KB_ptr;


struct dostime_t M_time;          /* Structure used for getting system time. */


                      /* Structure used to save time. */
struct savetime {
                  unsigned char hour;
                  unsigned char minute;
                  unsigned char second;
                } s_time;

struct dosdate_t date, file_date; /* Used to get system date. */

                      /* Structure used to save date. */
struct savedate {
                  unsigned char day;
                  unsigned char month;
                  unsigned int year;
                } s_date, f_date;

                /* This function opens all database files. */
FILE_OPEN()
{

        /* Open truck.db */
  BT_STATUS = BTRV(OPEN,POS_BLOCKS[TRUCK],t_ptr,&BUF_LENGTH[TRUCK],"truck.db",-1);
  if(BT_STATUS != 0) fprintf(stderr,"Error opening file truck. Status = %s.\n",Error_Code());

        /* Open flag.db */
  BT_STATUS = BTRV(OPEN,POS_BLOCKS[FLAG],f_ptr,&BUF_LENGTH[FLAG],"flag.db",-1);
  if(BT_STATUS != 0) fprintf(stderr,"Error opening file flag. Status = %s.\n",Error_Code());

        /* Open daily.db */
  BT_STATUS = BTRV(OPEN,POS_BLOCKS[DAILY],d_ptr,&BUF_LENGTH[DAILY],"daily.db",-1);
  if(BT_STATUS != 0) fprintf(stderr,"Error opening file daily. Status = %s.\n",Error_Code());

        /* Open manufact.db */
  BT_STATUS = BTRV(OPEN,POS_BLOCKS[MANUFACT],m_ptr,&BUF_LENGTH[MANUFACT],"manufact.db",-1);
```

```
  if(BT_STATUS != 0) fprintf(stderr,"Error opening file manufact. Status = %s.\n",Error_Code());

        /* Open color.db */
  BT_STATUS = BTRV(OPEN,POS_BLOCKS[COLOR],c_ptr,&BUF_LENGTH[COLOR],"color.db",-1);
  if(BT_STATUS != 0) fprintf(stderr,"Error opening file color. Status = %s.\n",Error_Code());

        /* Open owner.db */
  BT_STATUS = BTRV(OPEN,POS_BLOCKS[OWNER],o_ptr,&BUF_LENGTH[OWNER],"owner.db",-1);
  if(BT_STATUS != 0) fprintf(stderr,"Error opening file owner. Status = %s.\n",Error_Code());

        /* Open bad_read.db */
  BT_STATUS = BTRV(OPEN,POS_BLOCKS[BAD_READ],br_ptr,&BUF_LENGTH[BAD_READ],"bad_read.db",-1);
  if(BT_STATUS != 0) fprintf(stderr,"Error opening file bad_read. Status = %s.\n",Error_Code());

        /* Open comment.db */
  BT_STATUS = BTRV(OPEN,POS_BLOCKS[COMMENT],com_ptr,&BUF_LENGTH[COMMENT],"comment.db",-1);
  if(BT_STATUS != 0) fprintf(stderr,"Error opening file comment. Status = %s.\n",Error_Code());

}


                /* This function closes all database files. */
FILE_CLOSE()
{
        /* Close truck.db */
  BT_STATUS = BTRV(CLOSE,POS_BLOCKS[TRUCK],t_ptr,&BUF_LENGTH[TRUCK],"truck.db",0);
  if(BT_STATUS != 0) fprintf(stderr,"Error closing file truck. Status = %s.\n",Error_Code());

        /* Close flag.db */
  BT_STATUS = BTRV(CLOSE,POS_BLOCKS[FLAG],f_ptr,&BUF_LENGTH[FLAG],"flag.db",0);
  if(BT_STATUS != 0) fprintf(stderr,"Error closing file flag. Status = %s.\n",Error_Code());

        /* Close daily.db */
  BT_STATUS = BTRV(CLOSE,POS_BLOCKS[DAILY],d_ptr,&BUF_LENGTH[DAILY],"daily.db",0);
  if(BT_STATUS != 0) fprintf(stderr,"Error closing file daily. Status = %s.\n",Error_Code());

        /* Close manufact.db */
  BT_STATUS = BTRV(CLOSE,POS_BLOCKS[MANUFACT],m_ptr,&BUF_LENGTH[MANUFACT],"manufact.db",0);
  if(BT_STATUS != 0) fprintf(stderr,"Error closing file manufact. Status = %s.\n",Error_Code());

        /* Close color.db */
  BT_STATUS = BTRV(CLOSE,POS_BLOCKS[COLOR],c_ptr,&BUF_LENGTH[COLOR],"color.db",0);
  if(BT_STATUS != 0) fprintf(stderr,"Error closing file color. Status = %s.\n",Error_Code());

        /* Close owner.db */
  BT_STATUS = BTRV(CLOSE,POS_BLOCKS[OWNER],o_ptr,&BUF_LENGTH[OWNER],"owner.db",0);
  if(BT_STATUS != 0) fprintf(stderr,"Error closing file owner. Status = %s.\n",Error_Code());

        /* Close bad_read.db */
  BT_STATUS = BTRV(CLOSE,POS_BLOCKS[BAD_READ],br_ptr,&BUF_LENGTH[BAD_READ],"bad_read.db",0);
  if(BT_STATUS != 0) fprintf(stderr,"Error closing file bad_read. Status = %s.\n",Error_Code());
```

```
        /* Close comment.db */
  BT_STATUS = BTRV(CLOSE,POS_BLOCKS[COMMENT],com_ptr,&BUF_LENGTH[COMMENT],"comment.db",0);
  if(BT_STATUS != 0) fprintf(stderr,"Error closing file comment. Status = %s.\n",Error_Code());
}

                /* This Function returns a pointer to a string containing the */
                /* Error Message for the status code returned from Btrieve call. */
char *Error_Code()
{
  char *error_ptr;
  error_ptr = ERROR_CODES[BT_STATUS-1];  /* Uses global BT_STATUS determine address. */
  return(error_ptr);
}
```

Appendix IV

LOADCODE.C Source and Flow Chart

```
/*************************************************************************/
/* Program name: Loadcode.c              Programer: Gary Woodbridge      */
/* Purpose: This program loads dummy information into a database and then */
/*          times the retrevial process.  1000 records are added at a time*/
/*          and then 1000 random reads are made from the file.           */
/*          timeing information is stored in a data file specified by the */
/*          user.                                                        */
/*************************************************************************/
#include<stdio.h>
#include "btrieve.h"     /* Used to access data base pointers. */

                /* FUNCTION PROTOTYPES */
void LoadCode(void);    /* Loads dummy information into data structures. */
int SaveCode(void);      /* Saves data structures to data base files. */

                /* This function generates a Random offest into a data file. */
unsigned long int GenROffset(unsigned long int, unsigned long int);

                /* This function counts and saves bar codes in ASCII file. */
unsigned long int GetCountSaveBCODES(FILE *);

                /* This function is used to get bar code from ASCII file. */
void GetBCodeKey(unsigned long int, FILE *);


void GetBCodeInfo();     /* Retreives bar code info from data base. */


main()
{
  char firsttwolets[3],  /* Used when generating dummy bar codes.  Starting */
                          /* bar code letters. */

       lasttwolets[3];    /* Ending bar code letters. */

  int x, /* Used as general counter. */

      rv,        /* Used as return value from functions. */

      rcsk;      /* Used when creating data file names.  Number of K */
                 /* records in data base. */

  unsigned long int range=0,     /* Number of records in data base. */

                    rmulti,      /* Random Multiplier.  Used to generate */
                                 /* random number offset. */

                    searchs=1000,        /* Number of random searchs to  */
                                         /* do on data base.             */

                    x1,          /* General counter. */
```

```
                    hsec_stime,  /* Starting time in 100 th's of seconds. */
                    hsec_etime, /* Ending time in 100 th's of seconds. */
                    hsec_dtime, /* Difference time in 100 th's of seconds. */
                    offset;      /* Offset into ASCII file. */
FILE *codesfp, /* Used to store barcodes in ASCII format. */
     *ASCfp;           /* Used to store timeing information. */
struct dostime_t stime, /* Start time data structure. */
                 etime; /* End time data structure. */
char filename[13];     /* String used to hold file name. */
      /*****************************************************************/
      /* The following pointers are used by Btrieve to load data into */
      /* data structures for each file.  Defined in btrieve.h.        */
      /*****************************************************************/
t_ptr = &Truck_Data;   /* Pointer to truck data structure. */
d_ptr = &Daily_Data;   /* Pointer to daily data structure. */
f_ptr = &Flag_Data;    /* Pointer to flag data structure. */
m_ptr = &Man_Data;     /* Pointer to make data structure. */
c_ptr = &Color_Data;   /* Pointer to color data structure. */
o_ptr = &Owner_Data;   /* Pointer to owner data structure. */
br_ptr = &Bad_Read_Data;       /* Pointer to bad read data structure. */
com_ptr = &Comment_Data;       /* Pointer to comment data structure. */
KB_ptr = &KEY_BUF;             /* Pointer to key data structure. */
              /*********************************************************/
              /* The following statements specify the length of buffer */
              /* for Btrieve for each data structure.                  */
              /*********************************************************/
BUF_LENGTH[0] = sizeof(Truck_Data);
BUF_LENGTH[1] = sizeof(Flag_Data);
BUF_LENGTH[3] = sizeof(Daily_Data);
BUF_LENGTH[4] = sizeof(Man_Data);
BUF_LENGTH[5] = sizeof(Color_Data);
BUF_LENGTH[6] = sizeof(Owner_Data);
BUF_LENGTH[8] = sizeof(Bad_Read_Data);
BUF_LENGTH[9] = sizeof(Comment_Data);
```

```
system("startup");      /* Make system call to load BTRIEVE. */

FILE_OPEN();            /* Open database files.  Code in btrieve.h */

      /* This next section of code gets the first two letters of bar code sequence. */
      /* Bar codes used in this project have a 5 character format of */
      /* 2 characters and 3 digits.  */

fprintf(stdout,"Enter beginning first two letters of Bar Codes: ");
fgets(firsttwolets,3,stdin);
fflush(stdin);

      /* Want to be sure characters entered are a-Z */
while(!((isalpha(firsttwolets[0])) && (isalpha(firsttwolets[1]))))
{
      /* This loop will continue until valid codes are entered. */

  fprintf(stdout,"Enter beginning first two letters of Bar Codes: ");
  fgets(firsttwolets,3,stdin);
}

firsttwolets[0] = toupper(firsttwolets[0]);     /* Convert characters to */
firsttwolets[1] = toupper(firsttwolets[1]);     /* upper case.           */

      /* This section of code does the same thing but gets the last  */
      /* two letters of bar code run.                                */
fprintf(stdout,"Enter last first two letters of Bar Codes: ");
fgets(lasttwolets,3,stdin);
fflush(stdin);

              /* Want to be sure characters entered are a-Z */
while(!((isalpha(lasttwolets[0])) && (isalpha(lasttwolets[1]))))
{
      /* This loop will continue until valid codes are entered. */
  fprintf(stdout,"Enter last first two letters of Bar Codes: ");
  fgets(lasttwolets,3,stdin);
}

lasttwolets[0] = toupper(lasttwolets[0]);   /* Convert char. to upper case. */
lasttwolets[1] = toupper(lasttwolets[1]);

              /* This places in the file name how many 1000 entrys there */
              /* are in the data file.                                   */
fprintf(stdout,"Enter beginning K number: ");
fscanf(stdin,"%d%*c",&rcsk);
fflush(stdin);

srand(inp(64));         /* Seed random number generator with real time clock tick. */

      /* This loop is executed until firsttwolets is greater than lasttwolets. */
```

```
while(strcmp(firsttwolets,lasttwolets) <= 0)
{
  for(x=0;x<1000;x++)  /* This loop generates 1000 barcodes and stores them in database. */
  {
    sprintf(t_ptr->Bar_Code,"%s%03d\0",firsttwolets,x);  /* Place bar code in */
                                                         /* truck data structure. */

    sprintf(com_ptr->Bar_Code,"%s%03d\0",firsttwolets,x);  /* Place bar code in */
                                                           /* comment data structue. */

    fprintf(stdout,"%s\n",t_ptr->Bar_Code);      /* Display barcode to screen. */

    LoadCode();         /* Place more dummy info in data structures. */

    rv = SaveCode();    /* Save dummy info to database. */

    if(rv == 18) x = 1000;  /* if diskfull, exit loop. */
  }

  fclose(codesfp);      /* Make sure file pointer is free.. */

  codesfp = fopen("bcodes.dat","w");    /* Open file to store list of barcodes. */

  if(codesfp == NULL)  /* Display error message if unable to open file. */
  {
    fprintf(stderr,"Error opening file bcodes.dat.\n");
  }

  else
  {
    sprintf(filename,"t%dk.prn",rcsk++); /* Create filename. */

    fclose(ASCfp);  /* Make sure file pointer is free. */

    ASCfp = fopen(filename,"w"); /* Open file for write. */

    if(ASCfp == NULL)  /* Display error message if unable to open file. */
    {
      fprintf(stderr,"Error opening file %s.\n",filename);
    }

    else
    {
      searchs = 1000;  /* Set number of random searches. */

      range = GetCountSaveBCODES(codesfp);  /* Count and save bar codes in ascii file. */

      codesfp = fopen("bcodes.dat","r");  /* Open file to read bar codes. */

      if(codesfp == NULL)  /* Display error message if unable to open file. */
      {
        fprintf(stderr,"Error opening file bcodes.dat.\n");
```

```
}

else
{
  fprintf(ASCfp,"%cNumber of Entries%c,%lu\n",34,34,range);  /* Write info to data file. */
  fprintf(ASCfp,"%cNumber of Reads%c,%lu\n",34,34,searchs);

  rmulti = (range / 32767) + 1;  /* Figure Random Multiplier */
                /* The random number generator generates a number */
                /* between 0 and 32767.  Because the number of bar */
                /* codes will be greater than 32767, a multiplier must */
                /* used to retreive a bar code from the entire range of */
                /* bar codes.  */

  srand(inp(64));        /* Seed random number generator from time clock. */

        /* This loop will perform 1000 random retrevials on data base */
        /* and time them.      */
  for(x1=0;x1<searchs;x1++)
  {
        /* Want to be sure random offset is within file. */
        /* 6 is length of 5 character bar code plus NULL char. */
    while((offset = GenROffset(rmulti,range)) >= range*6) {}

    GetBCodeKey(offset,codesfp); /* Get key value from ASCII file. */

    _dos_gettime(&stime);        /* Get start time. */

    GetBCodeInfo();      /* Get Information from data base. */

    _dos_gettime(&etime);        /* Get ending time. */

        /* Convert start time to hundreds of seconds. */
    hsec_stime = ((unsigned long int)(stime.hour * 360000))+
                 ((unsigned long int)(stime.minute * 6000))+
                 ((unsigned long int)(stime.second * 100))+
                 ((unsigned long int)(stime.hsecond));

        /* Convert end time to hundreds of seconds. */
    hsec_etime = ((unsigned long int)(etime.hour * 360000))+
                 ((unsigned long int)(etime.minute * 6000))+
                 ((unsigned long int)(etime.second * 100))+
                 ((unsigned long int)(etime.hsecond));

                 /* Figure difference. */
    hsec_dtime = hsec_etime - hsec_stime;

    /* Print info to data file. */
    fprintf(ASCfp,"%c%s%c,%lu,%lu,%lu\n",34,t_ptr->Bar_Code,34,hsec_stime,hsec_etime,hsec_dtime);

    /* Print search number and barcode to screen. */
    fprintf(stderr,"%lu   %s\n",x1,t_ptr->Bar_Code);
```

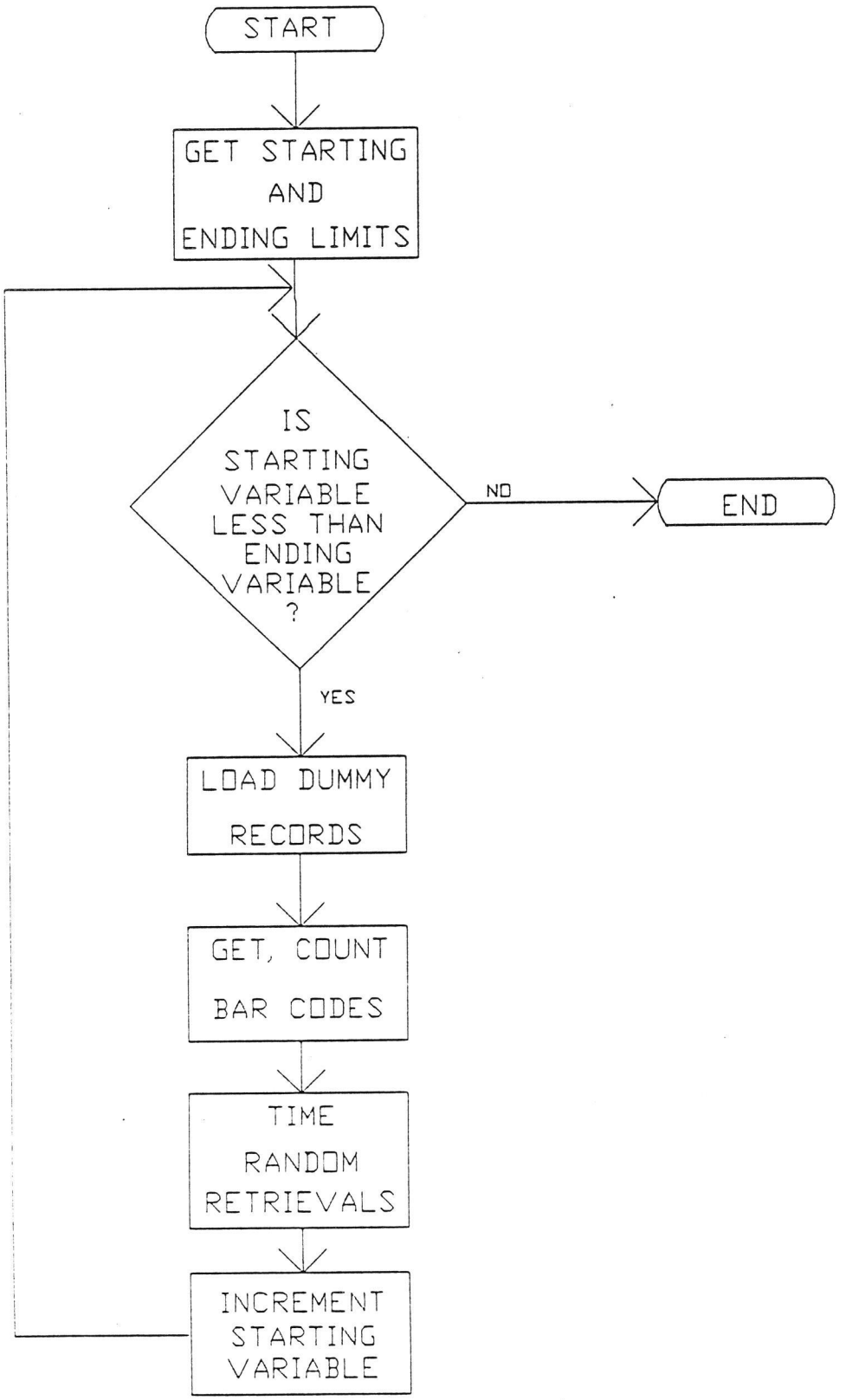
```
        }
      }
    }
  }

  firsttwolets[1]++;   /* Increment letter sequence. */

  if(firsttwolets[1] == 91)  /* If last of 2 letters is greater than Z */
  {
    firsttwolets[0]++; /* increment first letter and set last to A. */
    firsttwolets[1] = 65;
  }
 }

 FILE_CLOSE(); /* Close all database files. */

 fclose(codesfp);       /* Close ascii files. */
 fclose(ASCfp);

 system("stop");        /* Unload Btrieve from memory. */

 exit(0);       /* Clean exit. */
}

void LoadCode()          /* This function loads data structures with dummy info. */
{
 t_ptr->FHUTED.Month = 0;
 t_ptr->FHUTED.Year = 0;
 t_ptr->AHUTED.Month = 0;
 t_ptr->AHUTED.Year = 0;
 t_ptr->AFTED.Month = 0;
 t_ptr->AFTED.Year = 0;
 t_ptr->OWFED.Month = 0;
 t_ptr->OWFED.Year = 0;
 t_ptr->HMPED.Month = 0;
 t_ptr->HMPED.Year = 0;
 t_ptr->VIDN = 1234;
 t_ptr->M_YEAR = 1990;
 t_ptr->Own_Num = (rand()/6553) + 1;     /* Generate random owner number. */
 t_ptr->Make = rand()/1310;      /* Generate random make number. */
 t_ptr->Color1 = rand()/2047;    /* Generate random color numbers. */
 t_ptr->Color2 = rand()/2047;
 sprintf(t_ptr->Ap_Lic,"P1111-1\0");
 t_ptr->OWFT = 'Y';
 t_ptr->HMPT = 'Y';
 sprintf(t_ptr->state,"AR\0");
 t_ptr->App = 'Y';
 com_ptr->Comment1[0] = '\0';
 com_ptr->Comment2[0] = '\0';
}

int SaveCode()  /* This function saves the dummy data to data base. */
```

```
{
        /* Save comment info to comment.db. */
  BT_STATUS = BTRV(INSERT,POS_BLOCKS[COMMENT],com_ptr,sizeof(Comment_Data),com_ptr->Bar_Code,0);

  if(BT_STATUS) /* If there is an error. */
  {
    fprintf(stderr,"Error in saving comment field.\n");

    Error_Code();         /* Display error based on BT_STATUS. */
  }
  else
  {
        /* Save truck info to truck.db. */
    BT_STATUS = BTRV(INSERT,POS_BLOCKS[TRUCK],t_ptr,sizeof(Truck_Data),t_ptr->Bar_Code,8);

    if(BT_STATUS)  /* If there is an error. */
    {
      fprintf(stderr,"Error in saving truck field.\n");

      Error_Code();       /* Display error based on BT_STATUS. */
    }
  }
  return(BT_STATUS);
}

        /* This function generates a file that holds all the bar codes in the data base. */
        /* It also counts those bar codes.  The file is used to generate random access to*/
        /* the data base based on bar codes. */
unsigned long int GetCountSaveBCODES(FILE *fp /* file pointer to data file.*/)
{
  unsigned long int count=0;      /* Used to count bar codes. */

        /* Get the first bar code in data base. */
  BT_STATUS = BTRV(GET_FIRST,POS_BLOCKS[TRUCK],t_ptr,sizeof(Truck_Data),t_ptr->Bar_Code,8);
  while(!BT_STATUS)       /* Loop until end of file. */
  {
    count++;        /* Count bar code. */

    fwrite(t_ptr->Bar_Code,sizeof(t_ptr->Bar_Code),1,fp); /* Write bar code to file. */

        /* Get next bar code in database. */
    BT_STATUS = BTRV(GET_NEXT,POS_BLOCKS[TRUCK],t_ptr,sizeof(Truck_Data),t_ptr->Bar_Code,8);
  }

  fclose(fp);    /* Close data file. */

  fprintf(stderr,"Finished key count.\n");        /* Display info message to screen. */

  return(count); /* Return number of bar codes in data base. */
}
```

```
unsigned long int GenROffset(unsigned long int rmulti, /* Random multiplier. */
                             unsigned long int range /* Number of bar codes in data base. */)
{
  unsigned long int randnum;      /* random number. */

  randnum = range * 6;    /* Set to maximum number. */

  while(randnum >= (range * 6)) /* Generate random number in range of file. */
  {
    randnum = (unsigned long)((long)rand());      /* Generate random number. */

    randnum = randnum * rmulti * 6;       /* derive offset into file. */
  }
  return(randnum);        /* Return random offset into file. */
}

        /*  This function retreives the 6 char. bar code at offset in ASCII file. */
void GetBCodeKey(unsigned long int offset, /* Random offset. */
                 FILE *fp /* Data file pointer. */)
{
  int rv;        /* Return value of functions. */

  rv = fseek(fp,offset,SEEK_SET); /* Seek to random location. */

  if(!rv)   /* If no error in seek. */
  {
     /* Read bar code into truck data structure. */
    fread(t_ptr->Bar_Code,sizeof(t_ptr->Bar_Code),1,fp);
  }
  else t_ptr->Bar_Code[0] = NULL;   /* If an error. */
}
```

```
        /* This function retreives the data from the data base. */
void GetBCodeInfo()
{
        /* Retreive from truck.db based on Bar Code. */
  BT_STATUS = BTRV(GET_EQ,POS_BLOCKS[TRUCK],t_ptr,&BUF_LENGTH[TRUCK],t_ptr->Bar_Code,8);

        /* Retreive from owner.db based on owner number in truck data structure. */
  BT_STATUS = BTRV(GET_EQ,POS_BLOCKS[OWNER],o_ptr,&BUF_LENGTH[OWNER],&t_ptr->Own_Num,0);

        /* Retreive from color.db based on color number in truck data structure. */
  BT_STATUS = BTRV(GET_EQ,POS_BLOCKS[COLOR],c_ptr,&BUF_LENGTH[COLOR],&t_ptr->Color1,0);

        /* Retreive from color.db based on color number in truck data structure. */
  BT_STATUS = BTRV(GET_EQ,POS_BLOCKS[COLOR],c_ptr,&BUF_LENGTH[COLOR],&t_ptr->Color2,0);
```

```
        /* Retreive from manufact.db based on make number in truck data structure. */
    BT_STATUS = BTRV(GET_EQ,POS_BLOCKS[MANUFACT],m_ptr,&BUF_LENGTH[MANUFACT],&t_ptr->Make,0);

        /* Retreive from flag.db based on bar code. */
    BT_STATUS = BTRV(GET_EQ,POS_BLOCKS[FLAG],f_ptr,&BUF_LENGTH[FLAG],t_ptr->Bar_Code,0);

}
```


START
GET STARTING AND ENDING LIMITS
IS STARTING VARIABLE LESS THAN ENDING VARIABLE ?
NO
END
YES
LOAD DUMMY RECORDS
GET, COUNT BAR CODES
TIME RANDOM RETRIEVALS
INCREMENT STARTING VARIABLE

Appendix V

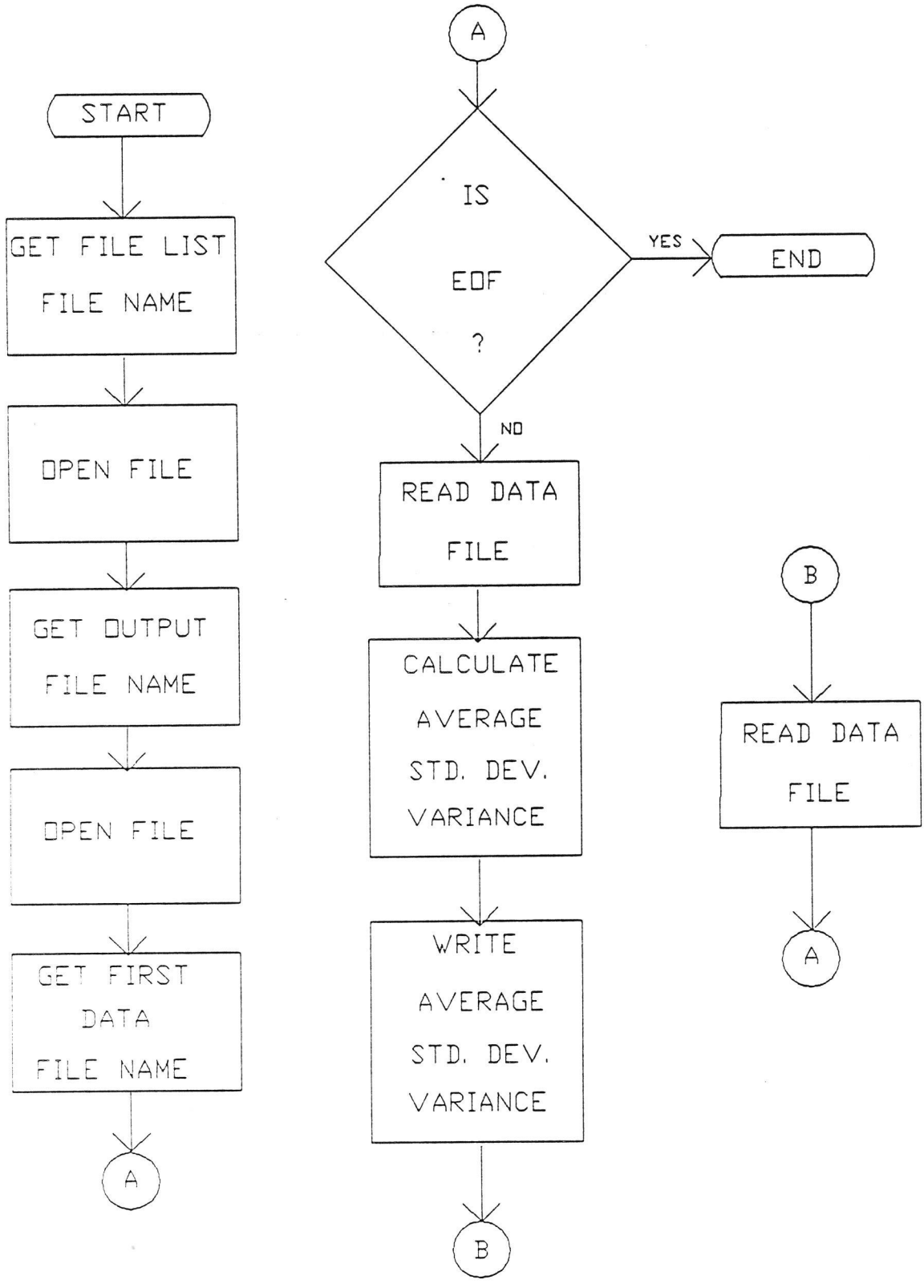
AVGSTD.C Source Code and Flow Chart

```
/*************************************************************************/
/* Program Name: avgstd.c                Programmer: Gary Woodbridge     */
/* Description:  This program takes the timeing data created by loadcode.c */
/*               and finds the maximum, minimum, average, standard       */
/*               deviation, and variance for each data file created.     */
/*               These values are then stored in another data file.      */
/*************************************************************************/
#include<stdio.h>
#include<math.h>
main()
{
  char filename[13], /* Used to enter filenames. */
       trash[80];    /* String used to collect character data in file. */

  double times[1000],   /* Array to hold time values. */
         avg,           /* Average of times. */
         stdev,         /* Standerd deviation. */
         cv,            /* Variance. */
         scans,         /* Number of scans. Used to figure avg. */
         entries,       /* Number of entries in data base. */
         max[2],min[2]; /* Max. and Min. figures. */

         /* The program uses 3 types of files. 1. File holding a list of  */
         /* the data file names.  2. The data files themselves.  3.  The  */
         /* data output file.                                             */

  FILE *fnfp, /* File with list of data file names. */
       *dffp, /* Data files pointer. */
       *avgfp;  /* Data output file. */

  int x,/* General counter. */
      max_loc,   /* Location in array of max value. */
      min_loc;  /* Loacation in array of min value. */

  unsigned long int hold,        /* Used to read in data. */
                    u_scans;    /* Unsigned scans. */

        /* This next section of code prompts the user to enter filename */
        /* of file holding list of data file names. */
  fprintf(stdout,"Enter filename of file holding filenames: ");

  fflush(stdin); /* Clear input buffer. */

  fgets(filename,13,stdin);

               /* Remove trailing carriage return. */
  if(filename[strlen(filename) - 1] == '\n') filename[strlen(filename) - 1] = '\0';


  fnfp = fopen(filename,"r");     /* Attempt to open file for read. */

  if(fnfp == NULL)       /* If there is an problem opening file. */
```

```
{
              /* Display error message and exit with value of 1. */
  fprintf(stderr,"Error opening file %s.\n",filename);
  exit(1);
}

      /* This section of code prompts the user for data output file. */
fprintf(stdout,"Enter filename of output file: ");

fflush(stdin); /* Clear input buffer. */

fgets(filename,13,stdin);

              /* Remove trailing carriage return. */
if(filename[strlen(filename) - 1] == '\n') filename[strlen(filename) - 1] = '\0';


avgfp = fopen(filename,"w");    /* Attempt to open file. */

if(avgfp == NULL)       /* If error, */
{
              /* Display error message and exit with a 1. */
  fprintf(stderr,"Error opening file %s.\n",filename);
  exit(1);
}

fprintf(stdout,"Working");      /* Display message to screen. */

fgets(filename,13,fnfp);        /* Get first data filename. */

while(!feof(fnfp))
{
  fprintf(stdout," .");         /* Display . to screen. */

              /* Remove trailing carriage return. */
  if(filename[strlen(filename) - 1] == '\n') filename[strlen(filename) - 1] = '\0';

  dffp = fopen(filename,"r");   /* Attempt to open data file. */

  if(dffp == NULL)      /* If error, display message. */
  {
    fprintf(stderr,"Error opening file %s\n",filename);
  }

  else
  {
    avg = stdev = cv = 0;       /* Set variables to 0. */

    max[0] = max[1] = 0;        /* Set max values low. */

    min[0] = min[1] = 1000;     /* Set min values high. */
```

```
fgets(trash,21,dffp);       /* Get character data. */

fscanf(dffp,"%ld%*c",&hold);        /* Get number of entries in database. */

entries = (double)((long)hold);     /* Convert to double pression. */

fgets(trash,19,dffp);       /* Get character data. */

fscanf(dffp,"%ld%*c",&hold);        /* Get number of scans. */

scans = (double)((long)hold);       /* Convert to double. */

u_scans = hold;     /* Also save scans as unsigned to use in loop. */

          /* This loop reads the file and computes avg. and std. dev. */
for(x=0;x<u_scans;x++)
{
  fgets(trash,9,dffp);      /* Get bar code. */

  fscanf(dffp,"%*U,%*U,%U%*c",&hold);       /* Get time difference info. */

  /* After looking at the data, it was found that the DOS interrupt 2C */
  /* that was used to determine the times sometimes returned ending times */
  /* less than the starting times.  When the subtraction was performed to */
  /* determine the difference time, a large number was produced due to the*/
  /* fact that unsigned values were used.  To elimnate these times, times */
  /* above 320000 were not used in the calculations.  These were at the   */
  /* most 3 out of 1000 reads.                                            */

  if(hold < 320000)
  {
    times[x] = (double)((long)hold);        /* Store time in array. */

    times[x] = times[x]/100.0;      /* Convert to seconds. */

    avg += times[x];                /* add to avg variable. */

    if(times[x] >= max[0]) /* Determine if greater than max. */
    {
      max[0] = times[x];            /* if it is, save it and location. */
      max_loc = x;
    }
    else if(times[x] > max[1]) max[1] = times[x]; /* Save second highest max also. */

    else if(times[x] <= min[0])     /* Check against min. */
    {
      min[0] = times[x];    /* Save min and location. */
      min_loc = x;
    }

    else if(times[x] < min[1]) min[1] = times[x];   /* Save second lowest. */
  }
```

```
        else  /* if invalid number. */
        {
          x--;  /* Back up x */

          u_scans--; /* decrement number of valid scans. */
        }
      }
      scans = ((double)((long)u_scans));  /* Convert to double. */

      avg /= scans;       /* Figure average. */

      stdev = 0.0;        /* Clear stdev variable. */

        /* This loop is used to sum the powers of the differences between the */
        /* average and each value to determine the standard dev. */
      for(x=0;x<u_scans;x++)
      {
        stdev = stdev + pow((times[x] - avg),2.0);
      }
      stdev = sqrt(stdev/(scans-1.0));    /* Figure Standard Dev. */

      cv = (stdev/avg)*100.0;     /* Figure variance. */

      /* Store number on entries, max, min average, std. dev. and var. in data file. */
      fprintf(avgfp,"%f,%f,%f,%f,%f,%f\n",entries,max[0],min[0],avg,stdev,cv);

      fclose(dffp);       /* close current data file. */


    }

    fgets(filename,13,fnfp);      /* Get next file name */
  }
  fclose(avgfp); /* Close output data file. */
}
```


START
GET FILE LIST
FILE NAME
OPEN FILE
GET OUTPUT
FILE NAME
OPEN FILE
GET FIRST
DATA
FILE NAME
A
A
IS
EOF
?
YES
END
NO
READ DATA
FILE
CALCULATE
AVERAGE
STD. DEV.
VARIANCE
WRITE
AVERAGE
STD. DEV.
VARIANCE
B
B
READ DATA
FILE
A

Appendix VI

Phase II Computing Equipment Specifications

Phase II Computing Equipment Specifications

PC/AC clone (Zenith ZW-284-84)

* 80286 CPU
* EGA card
* 40 meg. hard disk
* 1 parallel port
* 1 serial port

EGA monitor (Zenith ZVM-1380)

Appendix VII

Computer Interface Program

```
' ---------------

DEF FNUnsign (Intgr%)
    IF Intgr% < 0 THEN
        FNUnsign = Intgr% + 65536
    ELSE
        FNUnsign = Intgr%
    END IF
END DEF

DEF FNSign% (Real)
    IF Real < 32768 THEN
        FNSign% = Real
    ELSE
        FNSign% = Real - 65536
    END IF
END DEF

' ---------------
' Initialization routine

BaseAddr% = &H300
IntLvl% = 3
BarCode$ = "A5904"
NoRead$ = "XXXXX"

OPEN "com1:600,e,,,rs,ds0" FOR RANDOM AS 1

ON COM(1) GOSUB CommTrap
ON KEY(1) GOSUB RequestTrap
ON KEY(2) GOSUB ExitTrap

KEY 1, "NewRun"
KEY 2, "Exit"

CLS
PRINT "           Rim speed:"
PRINT
PRINT "    Scans requested:"
PRINT "    Scans completed:"
PRINT
PRINT "   Current bar code: "; BarCode$
PRINT
PRINT "          Good reads:"
PRINT "           Bad reads:"
PRINT "         Other reads:"
PRINT "       Missing reads:"

COM(1) ON

CALL Install(BaseAddr%, IntLvl%)

KEY ON
KEY(1) ON
KEY(2) ON

GoodReads = 0
BadReads = 0
OtherReads = 0
```

```
' ----------------------
' main loop

DO
    DO
        CALL Status(Flag%, Period%, ScansLeft%)
    LOOP UNTIL Flag%

    Period = FNUnsign(Period%)

    COM(1) STOP
    LOCATE 1, 21
    IF Period > 1 THEN
        PRINT USING "###.## mph"; 64260 / Period
    ELSE
        PRINT " <0.98 mph"
    END IF

    ScansDone = Scans - FNUnsign(ScansLeft%)
    MissingReads = ScansDone - (GoodReads + BadReads + OtherReads)

    LOCATE 4, 22: PRINT USING "#####"; ScansDone
    LOCATE 11, 22: PRINT USING "#####"; MissingReads
    COM(1) ON

LOOP
' ---------------
RequestTrap:

LOCATE 13, 20
INPUT "Number of scans requested"; Scans
Scans% = FNSign%(Scans)

LOCATE 3, 22: PRINT USING "#####"; Scans

GoodReads = 0
BadReads = 0
OtherReads = 0

LOCATE 8, 22: PRINT USING "#####"; GoodReads
LOCATE 9, 22: PRINT USING "#####"; BadReads
LOCATE 10, 22: PRINT USING "#####"; OtherReads
LOCATE 13, 1: PRINT TAB(70);

CALL Request(Scans%)

RETURN
' ----------------
CommTrap:

n = LOC(1)
IF n < 5 THEN RETURN
Scan$ = INPUT$(5, 1)

IF Scan$ = BarCode$ THEN
    GoodReads = GoodReads + 1
    LOCATE 8, 22: PRINT USING "#####"; GoodReads
ELSEIF Scan$ = NoRead$ THEN
    BadReads = BadReads + 1
    LOCATE 9, 22: PRINT USING "#####"; BadReads
```

```
ELSE
    OtherReads = OtherReads + 1
    LOCATE 10, 22: PRINT USING "#####"; OtherReads
END IF

MissingReads = ScansDone - (GoodReads + BadReads + OtherReads)
LOCATE 11, 22: PRINT USING "#####"; MissingReads

RETURN
' ----------------
ExitTrap:

CALL Remove
CLS
END
```

A. H. T. D. BAR CODE SCANNING PROJECT

EXPERIMENT I -- DATA SHEET

No.	TRIAL NO.	SPEED BEFORE/AFTER	GOOD READS	BAD READS	MIS-READS	TIME & DATE	TESTER
1	7	14.93 / 14.95	799	201	—	7:30PM 2-2-1980	m.H
2	10	15.05 / 15.03	831	169	—	7.45	〃
3	14	15.46 / 14.96	1000	—	—	8.05	〃
4	8	15.12 / 15.07	1000	—	—	8.16	〃
5	14	15.25 / 15.07	1000	—	—	8.36	〃
6	3	15.14 / 15.01	729	271	—	8.47	〃
7	2	15.08 / 15.01	215	785	—	9.09	〃
8	13	15.41 / 15.06	1000	—	—	9.22	〃
9	10	15.16 / 14.90	936	64	—	11.45PM 2-3-1980	〃
10	4	15.01 / 14.88	1000	—	—	11.57	〃
11	6	14.83 / 14.78	1000	—	—	12.02	〃
12	9	15.40 / 15.09	1000	—	—	12.15	〃
13	11	15.35 / 15.20	887	113	—	12.28	〃
14	5	15.39 / 15.20	1000	—	—	12.40	〃
15	5	15.20 / 15.14	1000	—	—	1.05	〃

* Experiment I: Testing 3 emblem variables to determine the emblem specifications which will give the best results.

A. H. T. D. BAR CODE SCANNING PROJECT

EXPERIMENT I - DATA SHEET

NO.	TRIAL NO.	SPEED BEFORE/AFTER	GOOD READS	BAD READS	MIS-READS	TIME & DATE	TESTER
16	6	15.20 / 15.04	1000	—	—	1.15	
17	9	15.68 / 15.33	1000	—	—	1.30	"
18	10	15.46 / 15.17	856	144	—	1.43	
19	1	15.39 / 15.21	675	325	—	2.00	"
20	3	15.27 / 15.12	764	236	—	2.15	"
21	7	15.04 / 15.12	781	219	—	2.30	"
22	4	15.a / 15.21	1000	—	—	2.45	"
23	6	14.99 / 15.15	1000	—	—	3.03	
24	12	14.98 / 14.86	1000	—	—	3.15	"
25	6	15.05 / 15.20	1000	—	—	3.28	
26	9	15.0 / 15.15	1000	—	—	3.40	"
27	3	14.91 / 14.90	755	245	—	3.55	"
28	4	14.95 / 15.20	1000	—	—	4.05	
29	14	15.0 / 14.88	1000	—	—	4.18	"
30	7	14.94 / 14.78	763	237	—	4.30	

* Experiment I: Testing 3 emblem variables to determine the emblem specifications which will give the best results.

A. H. T. D. BAR CODE SCANNING PROJECT

EXPERIMENT I - DATA SHEET

NO.	TRIAL NO.	SPEED BEFORE/AFTER	GOOD READS	BAD READS	MIS-READS	TIME & DATE	TESTER
31	9	14.93 / 15.0	1000	—	—	4.42	[illegible] H
32	3	14.92 / 14.58	678	322	—	4.54	
33	1	15.25 / 15.13	724	276	—		
34	11	15.55 / 15.32	850	150	—		
35	13	15.47 / 15.40	1000	—	—		
36	12	15.25 / 15.14	1000	—	—		
37	2	15.37 / 15.24	212	788	—		
38	2	15.20 / 15.14	244	756			
39	1	15.02 / 14.85	645	355	—	5:52 2/3/88	[illegible]
40	6	14.95 / 15.03	1000	—		6:05	[illegible]
41	13	15.10 / 15.00	1000	—		6:15	[illegible]
42	5	14.96 / 14.92	1000	—		6:28	[illegible]
43	8	15.09 / 15.10	1000	—		6:37	[illegible]
44	1	15.02 / 14.89	646	354	—	6:49	[illegible]
45	11	15.04 / 14.80	847	153		7:00	[illegible]

* Experiment I: Testing 3 emblem variables to determine the emblem specifications which will give the best results.

Appendix IX

Phase II Laser Scanning Equipment Specifications

Specifications

Phase II Laser Scanning Equipment

Moving-beam Laser Scanner (Insta Read 610)

* Class II HeNe Laser
* Fixed-mount moving-beam scanner (400 scans/sec.)
* Max. scanning range 6-10 ft. (code & optics dependent)
* Reads Code 39
* Operates in external enable mode
* RS-232 Serial output

Appendix X

Samples of Vehicle Data

Carco Trucking Co.

FIRST ADDRESS

[illegible]: Smith , AR

SECOND ADDRESS

,

AR002 Colors: White ,
Year: 1987 Make: International
Id Number: 2022 License: P17-264
Permits: Overweight: N Hazardus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR003 Colors: White , Tan
Year: 1984 Make: International
Id Number: 1900 License: P17-154
Permits: Overweight: N Hazardus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

[illegible]04 Colors: White , Tan
Year: 1985 Make: International
Id Number: 1938 License: P17-184
Permits: Overweight: N Hazardus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR005 Colors: White ,
Year: 1986 Make: International
Id Number: 1981 License: P17-224
Permits: Overweight: N Hazardus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR007
Colors: White
Year: 1987 Make: International
Id Number: 3819 License: P27-588
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR008
Colors: White
Year: 1985 Make: International
Id Number: 1961 License: P17-207
Permits: Overweight: N Hazordus: N
Fees: Highway Use: Y Fuel Tax: Y
Comments:
Flag:

AR009
Colors: White
Year: 1984 Make: International
Id Number: 3568 License: P17-334
Permits: Overweight: N Hazordus: N
Fees: Highway Use: Y Fuel Tax: Y
Comments:
Flag:

AR010
Colors: White
Year: 1987 Make: International
Id Number: 2000 License: P17-242
Permits: Overweight: N Hazordus: N
Fees: Highway Use: Y Fuel Tax: Y
Comments:
Flag:

AR012
Colors: White
Year: 1984 Make: Freightliner
Id Number: 3401 License: P17-322
Permits: Overweight: N Hazordus: N
Fees: Highway Use: Y Fuel Tax: Y
Comments:
Flag:

AR013
Colors: White
Year: 1987 Make: International
Id Number: 3818 License: P17-557
Permits: Overweight: N Hazordus: N
Fees: Highway Use: Y Fuel Tax: Y
Comments:
Flag:

AR014
Colors: White
Year: 1987 Make: International
Id Number: 3820 License: P27-559
Permits: Overweight: N Hazordus: N
Fees: Highway Use: Y Fuel Tax: Y
Comments:
Flag:

AR015
Colors: White
Year: 1987 Make: International
Id Number: 3827 License: P27-562
Permits: Overweight: N Hazordus: N
Fees: Highway Use: Y Fuel Tax: Y
Comments:
Flag:

AR016
Colors: White
Year: 1984 Make: International
Id Number: 1919 License: P17-167
Permits: Overweight: N Hazordus: N
Fees: Highway Use: Y Fuel Tax: Y
Comments:
Flag:

AR017
Colors: White
Year: 1987 Make: International
Id Number: 2021 License: P17-263
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR018
Colors: White
Year: 1985 Make: International
Id Number: 1252 License: K716131
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR019
Colors: White
Year: 1985 Make: International
Id Number: 1255 License: K716139
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR020
Colors: White
Year: 1984 Make: International
Id Number: 1917 License: P17-165
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR021
Colors: White
Year: 1986 Make: International
Id Number: 1984 License: P17-227
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR022
Colors: White
Year: 1986 Make: International
Id Number: 1270 License: K716138
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR023
Colors: White
Year: 1987 Make: International
Id Number: 3816 License: P27-555
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR025
Colors: White
Year: 1987 Make: International
Id Number: 2054 License: P27-296
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR026
Colors: White
Year: 1985 Make: International
Id Number: 1251 License: K716130
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR027
Colors: White
Year: 1986 Make: International
Id Number: 1993 License: P17-236
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR028
Colors: White
Year: 1987 Make: International
Id Number: 3821 License: P27-560
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR029
Colors: White
Year: 1986 Make: International
Id Number: 1974 License: P17-217
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR030
Colors: White
Year: 1986 Make: International
Id Number: 1408 License: K19226
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR031
Colors: White
Year: 1986 Make: International
Id Number: 1402 License: K501776
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR032
Colors: White
Year: 1986 Make: International
Id Number: 1403 License: K501777
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR033
Colors: White ,
Year: 1986 Make: International
Id Number: 1401 License: K501775
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR034
Colors: White ,
Year: 1986 Make: International
Id Number: 1404 License: K501778
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR035
Colors: White ,
Year: 1986 Make: International
Id Number: 1405 License: K501779
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR036
Colors: White ,
Year: 1986 Make: International
Id Number: 3802 License: P26-963
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR037
Colors: White ,
Year: 1983 Make: International
Id Number: 1802 License: P17-394
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR039
Colors: White ,
Year: 1985 Make: International
Id Number: 1939 License: P17-185
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR040
Colors: White
Year: 1987 Make: International
Id Number: 2063 License: P17-305
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR041
Colors: White
Year: 1986 Make: International
Id Number: 1978 License: P17-221
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR042
Colors: White
Year: 1983 Make: International
Id Number: 1806 License: K716369
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR043
Colors: White
Year: 1987 Make: International
Id Number: 3828 License: K19619
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR044
Colors: White
Year: 1987 Make: International
Id Number: 2065 License: P17-307
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR045
Colors: White
Year: 1987 Make: International
Id Number: 2031 License: P17-273
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR046
Colors: White
Year: 1985 Make: International
Id Number: 1253 License: K716132
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR047
Colors: White
Year: 1985 Make: International
Id Number: 1258 License: K716136
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR048
Colors: White
Year: 1985 Make: International
Id Number: 1256 License: K716134
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR050
Colors: White
Year: 1985 Make: International
Id Number: 1951 License: P17-197
Permits: Overweight: N Hazordus: N
Fees: Highway Use: Y Fuel Tax: Y
Comments:
Flag:

AR051
Colors: White
Year: 1985 Make: International
Id Number: 1250 License: K716129
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR052
Colors: White
Year: 1984 Make: International
Id Number: 1908 License: P17-157
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR053 Colors: White
Year: 1985 Make: International
Id Number: 1948 License: P17-194
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR054 Colors: White
Year: 1987 Make: International
Id Number: 2076 License: P17-318
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

AR055 Colors: White
Year: 1987 Make: International
Id Number: 2018 License: P17-260
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR057 Colors: White
Year: 1984 Make: International
Id Number: 1906 License: P17-155
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR058 Colors: White
Year: 1984 Make: International
Id Number: 1925 License: P17-171
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR059 Colors: White
Year: 1986 Make: International
Id Number: 1406 License: K716937
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR060 Colors: White ,
Year: 1983 Make: International
Id Number: 7505 License: P17-353
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR061 Colors: White ,
Year: 1987 Make: International
Id Number: 2044 License: P17-286
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR062 Colors: White ,
Year: 1986 Make: International
Id Number: 1995 License: P17-238
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR064 Colors: White ,
Year: 1987 Make: International
Id Number: 2012 License: P17-254
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR065 Colors: White ,
Year: 1984 Make: International
Id Number: 306 License: P17-149
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR066 Colors: White ,
Year: 1987 Make: International
Id Number: 2040 License: P17-282
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR067 Colors: White ,
Year: 1985 Make: International
Id Number: 1944 License: P17-190
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR069 Colors: White ,
Year: 1985 Make: International
Id Number: 1351 License: K716366
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR070 Colors: White ,
Year: 1987 Make: International
Id Number: 2048 License: P17-290
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR071 Colors: White ,
Year: 1987 Make: International
Id Number: 2041 License: P17-283
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR072 Colors: White ,
Year: 1986 Make: International
Id Number: 1990 License: P17-233
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR073 Colors: White ,
Year: 1985 Make: International
Id Number: 1955 License: P17-201
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR074 Colors: White ,
Year: 1985 Make: International
Id Number: 1259 License: K716137
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR075 Colors: White ,
Year: 1985 Make: International
Id Number: 1254 License: K716133
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR076 Colors: White ,
Year: 1984 Make: International
Id Number: 1930 License: P17-176
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR077 Colors: White ,
Year: 1987 Make: International
Id Number: 1409 License: K716940
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR078 Colors: White ,
Year: 1985 Make: International
Id Number: 1257 License: K716135
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR079 Colors: White ,
Year: 1986 Make: International
Id Number: 1972 License: P17-215
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR080 Colors: White
Year: 1986 Make: International
Id Number: 1407 License: K716938
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR081 Colors: White
Year: 1987 Make: International
Id Number: 2004 License: P17-246
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR082 Colors: White
Year: 1985 Make: International
Id Number: 1953 License: P17-199
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR083 Colors: White
Year: 1984 Make: International
Id Number: 1929 License: P17-175
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR084 Colors: White
Year: 1985 Make: International
Id Number: 1941 License: P17-187
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR085 Colors: White
Year: 1985 Make: International
Id Number: 1947 License: P17-193
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR086 Colors: White ,
Year: 1984 Make: International
Id Number: 315 License: P17-150
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR087 Colors: White ,
Year: 1985 Make: International
Id Number: 1352 License: K716367
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR088 Colors: White ,
Year: 1986 Make: International
Id Number: 3805 License: P26-965
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR089 Colors: White ,
Year: 1984 Make: International
Id Number: 1932 License: P17-178
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR090 Colors: White ,
Year: 1987 Make: International
Id Number: 2007 License: P17-249
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR091 Colors: White ,
Year: 1987 Make: International
Id Number: 2045 License: P17-287
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR092
Colors: White ,
Year: 1986 Make: International
Id Number: 1996 License: P17-239
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR093
Colors: White ,
Year: 1984 Make: International
Id Number: 1911 License: P17-159
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR094
Colors: White ,
Year: 1984 Make: International
Id Number: 1958 License: P17-204
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR095
Colors: White ,
Year: 1984 Make: International
Id Number: 1918 License: P17-166
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: EDGE SEALANT USED.
Flag:

AR096
Colors: White ,
Year: 1984 Make: International
Id Number: 7160 License: P17-352
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: EDGE SEALANT USED.
Flag:

AR097
Colors: White ,
Year: 1984 Make: International
Id Number: 3575 License: K716372
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: EDGE SEALANT USED.
Flag:

AR098 Colors: White ,
Year: 1987 Make: International
Id Number: 2042 License: P17-284
Permits: Overweight: N Hazardus: N
Fees: Highway Use: N Fuel Tax: N
Comments: EDGE SEALANT USED.
Flag:

AR099 Colors: White ,
Year: 1986 Make: International
Id Number: 1980 License: P17-223
Permits: Overweight: N Hazardus: N
Fees: Highway Use: N Fuel Tax: N
Comments: EDGE SEALANT USED.
Flag:

AR113 Colors: White ,
Year: 1985 Make: International
Id Number: 1937 License: P17-183
Permits: Overweight: N Hazardus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR114 Colors: White ,
Year: 1985 Make: International
Id Number: 1933 License: P17-179
Permits: Overweight: N Hazardus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR115 Colors: White ,
Year: 1985 Make: International
Id Number: 1957 License: P17-203
Permits: Overweight: N Hazardus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR11.. Colors: White ,
Year: 1984 Make: International
Id Number: 1902 License: J606065
Permits: Overweight: N Hazardus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR117 Colors: White ,
Year: 1985 Make: International
Id Number: 1952 License: P17-198
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR118 Colors: White ,
Year: 1985 Make: International
Id Number: 1956 License: P17-202
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR119 Colors: White ,
Year: 1985 Make: International
Id Number: 1942 License: P17-188
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: EDGE SEALANT USED.
Flag:

AR120 Colors: White ,
Year: 1984 Make: International
Id Number: 1927 License: P17-173
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: EDGE SEALANT USED.
Flag:

AR121 Colors: White ,
Year: 1986 Make: International
Id Number: 1976 License: P17-219
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: EDGE SEALANT USED.
Flag:

AR122 Colors: White ,
Year: 1986 Make: International
Id Number: 3826 License: P17-561
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: EDGE SEALANT USED.
Flag:

AR123
Colors: White
Year: 1986 Make: International
Id Number: 1424 License: K716368
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: EDGE SEALANT USED.
Flag:

AR124
Colors: White
Year: 1987 Make: International
Id Number: 2052 License: P17-294
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: EDGE SEALANT USED.
Flag:

AR125
Colors: White
Year: 1987 Make: International
Id Number: 2019 License: P17-261
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: EDGE SEALANT USED.
Flag:

AR126
Colors: White
Year: 1986 Make: International
Id Number: 1971 License: P17-214
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: EDGE SEALANT USED.
Flag:

AR127
Colors: White
Year: 1987 Make: International
Id Number: 3817 License: P27-556
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: EDGE SEALANT USED.
Flag:

AR128
Colors: White
Year: 1985 Make: International
Id Number: 1934 License: P17-180
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: EDGE SEALANT USED.
Flag:

AR129
Colors: White
Year: 1987 Make: International
Id Number: 2034 License: P17-276
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: EDGE SEALANT USED.
Flag:

AR130
Colors: White
Year: 1984 Make: International
Id Number: 6581 License: P17-345
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: EDGE SEALANT USED.
Flag:

AR131
Colors: White
Year: 1987 Make: International
Id Number: 2049 License: P17-291
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: EDGE SEALANT USED.
Flag:

AR132
Colors: White
Year: 1986 Make: International
Id Number: 1991 License: P17-234
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: EDGE SEALANT USED.
Flag:

AR133
Colors: White
Year: 1987 Make: International
Id Number: 2028 License: P17-270
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: EDGE SEALANT USED.
Flag:

AR134
Colors: White
Year: 1987 Make: International
Id Number: 2079 License: P17-321
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: EDGE SEALANT USED.
Flag:

AR133 Colors: White , White
Year: 0 Make: International
Id Number: 3577 License: P17336
Permits: Overweight: N Hazordus: N
Fees: Highway Use: Y Fuel Tax: N
Comments: Cloudy.
Flag:

AR195 Colors: White , White
Year: 0 Make: International
Id Number: 1910 License: P17158
Permits: Overweight: N Hazordus: N
Fees: Highway Use: Y Fuel Tax: N
Comments:
Flag:

AR196 Colors: White , White
Year: 0 Make: International
Id Number: 12036 License: P17278
Permits: Overweight: N Hazordus: N
Fees: Highway Use: Y Fuel Tax: N
Comments:
Flag:

AR281 Colors: White , White
Year: 0 Make: International
Id Number: 1975 License: P17-218
Permits: Overweight: N Hazordus: N
Fees: Highway Use: Y Fuel Tax: N
Comments: Cloudy.
Flag:

AR483 Colors: Gray ,
Year: 1985 Make: International
Id Number: 221 License: P18084
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR483 Colors: Gray ,
Year: 1985 Make: International
Id Number: 22916 License: P18461
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR484
Colors: Gray ,
Year: 1985 Make: International
Id Number: 22995 License: P18534
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR485
Colors: Gray ,
Year: 1985 Make: White
Id Number: 373 License: P18190
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR486
Colors: Gray ,
Year: 1985 Make: International
Id Number: 22962 License: P18501
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR487
Colors: Gray ,
Year: 1985 Make: International
Id Number: 350 License: P18167
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR726
Colors: White ,
Year: 0 Make: International
Id Number: 2108 License: P30342
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR727
Colors: White ,
Year: 0 Make: International
Id Number: 3086 License: P29614
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR728
Colors: White
Year: 0 Make: International
Id Number: 2094 License: P30328
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR729
Colors: White
Year: 1988 Make: International
Id Number: 2103 License: P30337
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR730
Colors: White
Year: 1984 Make: International
Id Number: 0 License:
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR731
Colors: White
Year: 0 Make: International
Id Number: 3804 License: P26964
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

R732
Colors: White
Year: 0 Make: International
Id Number: 2104 License: P130338
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR733
Colors: White
Year: 1987 Make: International
Id Number: 12023 License: P17265
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR734 Colors: White
Year: 1987 Make: International
Id Number: 12053 License: P17295
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR735 Colors: White
Year: 1988 Make: International
Id Number: 2093 License: P30327
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR736 Colors: White
Year: 1989 Make: International
Id Number: 2013 License: P17255
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR737 Colors: White
Year: 1987 Make: International
Id Number: 12069 License: PP17311
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR738 Colors: White
Year: 1986 Make: International
Id Number: 1926 License: P17172
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR739 Colors: White
Year: 1986 Make: International
Id Number: 6583 License: P17346
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR740
Colors: White ,
Year: 1985 Make: International
Id Number: 3576 License: P17335
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR741
Colors: White ,
Year: 0 Make: International
Id Number: 12070 License: P17312
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR745
Colors: White ,
Year: 1987 Make: International
Id Number: 6538 License: P17320
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR999
Colors: White ,
Year: 1987 Make: International
Id Number: 6538 License: p17320
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

TX031
Colors: Gray ,
Year: 1985 Make: White
Id Number: 386 License: P18203
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

TX032
Colors: Gray ,
Year: 1985 Make: International
Id Number: 347 License: P18164
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

ABF

FIRST ADDRESS

Fort Smith , AR

SECOND ADDRESS

,

AR101 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18289 License: P28641
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR102 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18288 License: P28682
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

03 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18208 License: P28574
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR104 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18106 License: P28543
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR105 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18290 License: P28708
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR106 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18284 License: P28734
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR107 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18016 License: P29674
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR108 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18149 License: P28675
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR109 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18212 License: P58581
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR110 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18116 License: P28507
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR112 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18181 License: P28546
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR136 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18163 License: P28494
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR138 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18032 License: P29673
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR139 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18142 License: P28500
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR140 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18224 License: P28571
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR141 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18261 License: P28703
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR142 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18360 License: P28624
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR143 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18285 License: P28715
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR144 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18344 License: P28637
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR145 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18049 License: P29660
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR146 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18328 License: P28636
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR147 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18030 License: P29646
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR148 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18001 License: P28466
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR149 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18221 License: P28576
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR150 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18352 License: P28588
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR197 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18053 License: P28636
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR198 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18165 License: P28518
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR199 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18260 License: P28483
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR262 Colors: Green , White
Year: 1988 Make: Mack
Id Number: 18035 License: P29683
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR263 Colors: Green , White
Year: 1988 Make: Mack
Id Number: 18274 License: P28710
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR264 Colors: Green , White
Year: 1988 Make: Mack
Id Number: 18283 License: P28674
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR265 Colors: Green , White
Year: 1988 Make: Mack
Id Number: 18251 License: P28704
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR266 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18026 License: P29649
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR267 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18010 License: P29681
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR268 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18245 License: P28714
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR269 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18306 License: P28621
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR270 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18219 License: P28600
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR742 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18201 License: P28566
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR743 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18341 License: P28709
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR744 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18250 License: P28665
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR746 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18034 License: P28634
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR747 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18262 License: P28678
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR748 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18353 License: P28672
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR749 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18359 License: P28693
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR750 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18318 License: P28644
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR751 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18333 License: P28587
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR752 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18182 License: P28557
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR768 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18092 License: P29659
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR769 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18297 License: P28657
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR770 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18229 License: P28671
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR771 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18327 License: P28670
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR772 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18372 License: P28727
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR773 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18270 License: P28553
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR778 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18139 License: P28541
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR779 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18345 License: P28719
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

TX002 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18076 License: P29640
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

TX007 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18295 License: P28725
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

TX008 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18366 License: P28691
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

TX009 Colors: Green , Yellow
Year: 1988 Make: Mack
Id Number: 18276 License: P28740
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

Silica Transport Inc

FIRST ADDRESS
P.O.Box 9 W Market St.
Sulon , AR 72540

SECOND ADDRESS

,

AR366 Colors: , White
Year: 0 Make: International
Id Number: 896 License: P18914
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR368 Colors: , White
Year: 0 Make: International
Id Number: 897 License: P18915
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: CLOUDY.
Flag:

AR369 Colors: , White
Year: 0 Make: International
Id Number: 892 License: P18947
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

AR372 Colors: , White
Year: 0 Make: International
Id Number: 884 License: P18944
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

AR373 Colors: , White
Year: 0 Make: International
Id Number: 883 License: P18943
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

AR374 Colors: , White
Year: 0 Make: International
Id Number: 882 License: P18942
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

AR375 Colors: , White
Year: 0 Make: International
Id Number: 881 License: P18941
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

AR376 Colors: , White
Year: 88 Make: International
Id Number: 880 License: P106497
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

AR377 Colors: , White
Year: 0 Make: International
Id Number: 879 License: P18939
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny
Flag:

AR378 Colors: , White
Year: 0 Make: International
Id Number: 877 License: P18937
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: CLOUDY, CALM, RAINY, 40-50 DEGREES
Flag:

AR379 Colors: , White
Year: 0 Make: International
Id Number: 876 License: P18936
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Cloudy.
Flag:

AR381 Colors: , White
Year: 0 Make: International
Id Number: 873 License: P18933
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: CLOUDY.
Flag:

AR382 Colors: , White
Year: 0 Make: International
Id Number: 872 License: P18932
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

AR383 Colors: , White
Year: 0 Make: International
Id Number: 871 License: P18931
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

AR384 Colors: , White
Year: 0 Make: International
Id Number: 870 License: P18930
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny
Flag:

AR385 Colors: White ,
Year: 86 Make: International
Id Number: 869 License: P52032
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: CLOUDY, RAINY 40-50 DEGREES
Flag:

AR386 Colors: White ,
Year: 86 Make: International
Id Number: 869 License: P52032
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: CLOUDY, RAINY, 40-50 DEGREES
Flag:

AR387 Colors: , White
Year: 0 Make: International
Id Number: 867 License: P52030
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR389 Colors: , White
Year: 0 Make: International
Id Number: 865 License: P18925
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny.
Flag:

AR390 Colors: , White
Year: 0 Make: International
Id Number: 864 License: P18924
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny
Flag:

AR391 Colors: , White
Year: 0 Make: International
Id Number: 863 License: P18923
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

AR392 Colors: , White
Year: 0 Make: International
Id Number: 862 License: P18922
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

AR393 Colors: , White
Year: 0 Make: International
Id Number: 861 License: P18921
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

AR394 Colors: , White
Year: 0 Make: International
Id Number: 860 License: P18920
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Cloudy
Flag:

AR395 Colors: , White
Year: 0 Make: International
Id Number: 859 License: P18919
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR396 Colors: , White
Year: 0 Make: International
Id Number: 855 License: P18917
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Cloudy.
Flag:

AR397 Colors: , White
Year: 0 Make: International
Id Number: 854 License: P18916
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR398 Colors: , White
Year: 0 Make: International
Id Number: 899 License: P18913
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny
Flag:

AR400 Colors: , White
Year: 0 Make: International
Id Number: 898 License: P18918
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny
Flag:

AR753 Colors: White ,
Year: 86 Make: International
Id Number: 868 License: P53031
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: CLOUDY, RAINY, 40-50 DEGREES
Flag:

AR754 Colors: White ,
Year: 0 Make: International
Id Number: 875 License: P52038
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: CLOUDY, RAINY, 40-50 DEGREES
Flag:

AR755 Colors: , White
Year: 88 Make: International
Id Number: 880 License: P106497
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

AR756 Colors: White ,
Year: 86 Make: International
Id Number: 869 License: P52032
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: CLOUDY, RAINY 40-50 DEGREES
Flag:

AR757 Colors: , White
Year: 0 Make: International
Id Number: 859 License: P18919
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR758 Colors: White ,
Year: 86 Make: International
Id Number: 853 License: P52016
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

AR759 Colors: White ,
Year: 86 Make: International
Id Number: 852 License: P52015
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR760 Colors: , White
Year: 0 Make: International
Id Number: 854 License: P18916
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR761 Colors: , White
Year: 0 Make: International
Id Number: 877 License: P18937
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: CLOUDY, CALM, RAINY, 40-50 DEGREES
Flag:

AR762 Colors: , White
Year: 0 Make: Freightliner
Id Number: 874 License: P18934
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Cloudy, Calm, Rainy, 40-50 degrees
Flag:

AR763 Colors: White ,
Year: 0 Make: Freightliner
Id Number: 834 License: K718946
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Cloudy, Calm, Rainy, 40-50 degrees
Flag:

AR764 Colors: , White
Year: 0 Make: International
Id Number: 860 License: P52023
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR765 Colors: , White
Year: 0 Make: International
Id Number: 867 License: P52030
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

TX040 Colors: , White
Year: 0 Make: International
Id Number: 872 License: P18932
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

TX041 Colors: , White
Year: 0 Make: International
Id Number: 862 License: P18922
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

TX042 Colors: , White
Year: 0 Make: International
Id Number: 863 License: P18923
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

TX045 Colors: , White
Year: 0 Make: International
Id Number: 881 License: P18941
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

TX047
Colors:
Year: 0 Make: International , White
Id Number: 861 License: P18921
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

TX048
Colors:
Year: 0 Make: International , White
Id Number: 894 License:
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

TX049
Colors:
Year: 0 Make: International , White
Id Number: 855 License: P18917
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

TX050
Colors:
Year: 0 Make: International , White
Id Number: 883 License: P18943
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

TX051
Colors:
Year: 0 Make: International , White
Id Number: 873 License: P18933
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

X 32
Colors:
Year: 0 Make: International , White
Id Number: 864 License: P18924
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

TX053 Colors: , White
Year: 0 Make: International
Id Number: 884 License: P18944
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

TX054 Colors: , White
Year: 0 Make: International
Id Number: 866 License: P18926
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

TX055 Colors: , White
Year: 0 Make: International
Id Number: 892 License: P18947
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

TX056 Colors: , White
Year: 0 Make: International
Id Number: 890 License: 618945
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

TX057 Colors: , White
Year: 0 Make: International
Id Number: 891 License: P18946
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

TX059 Colors: , White
Year: 0 Make: International
Id Number: 882 License: P18942
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

TX061 Colors: , White
Year: 0 Make: International
Id Number: 856 License: P18918
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

TX063 Colors: , White
Year: 0 Make: International
Id Number: 893 License:
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

TX064 Colors: , White
Year: 0 Make: International
Id Number: 879 License: P18939
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

TX065 Colors: , White
Year: 0 Make: International
Id Number: 878 License: P18938
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

TX067 Colors: , White
Year: 0 Make: International
Id Number: 871 License: P18931
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

TX068 Colors: , White
Year: 0 Make: International
Id Number: 870 License: P18930
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

TX069 Colors: , White
Year: 0 Make: International
Id Number: 865 License: P18925
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

TX388 Colors: , White
Year: 0 Make: International
Id Number: 866 License: P18926
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: Sunny and 55 degrees.
Flag:

Quality Foods

FIRST ADDRESS
Asher Ave.
Little Rock , AR 72203

SECOND ADDRESS

Pam Trucking Co.

FIRST ADDRESS

SECOND ADDRESS

AR301 Colors: Gray
Year: 0 Make: White
Id Number: 393 License: P18210
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR302 Colors: Gold
Year: 0 Make: International
Id Number: 4535 License: P18309
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

303 Colors: Gold
Year: 0 Make: White
Id Number: 394 License: P18211
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR305 Colors: Gray
Year: 0 Make: International
Id Number: 1935 License: P18053
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR306
Colors: Gray
Year: 0 Make: Freightliner
Id Number: 1514 License: P100498
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR307
Colors: Gray
Year: 0 Make: International
Id Number: 173 License: P18037
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR308
Colors: Gray
Year: 0 Make: Freightliner
Id Number: 1530 License: P115051
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR309
Colors: Gold
Year: 0 Make: International
Id Number: 4485 License: P18305
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR310
Colors: Gray
Year: 0 Make: Freightliner
Id Number: 1448 License: P18653
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR311
Colors: Gray
Year: 0 Make: International
Id Number: 269 License: P18132
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR312
Colors: Gold
Year: 0 Make: International
Id Number: 5185 License: P18374
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR313
Colors: Gold
Year: 0 Make: International
Id Number: 4415 License: P18298
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR314
Colors: Gray
Year: 0 Make: Freightliner
Id Number: 1542 License:
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR315
Colors: Gray
Year: 0 Make: White
Id Number: 403 License: Q18220
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR316
Colors: Gray
Year: 0 Make: Freightliner
Id Number: 14495 License: P18654
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR317
Colors: Gray
Year: 0 Make: International
Id Number: 23061 License: P18600
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR408 Colors: Gray ,
Year: 0 Make: International
Id Number: 203 License: P18067
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR409 Colors: Gray ,
Year: 0 Make: International
Id Number: 190 License: P18054
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR410 Colors: Gray ,
Year: 0 Make: International
Id Number: 229 License: P18092
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR411 Colors: Gray ,
Year: 0 Make: Freightliner
Id Number: 1516 License: P100782
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR412 Colors: Gray ,
Year: 0 Make: International
Id Number: 156 License: P18020
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR413 Colors: Gray ,
Year: 0 Make: International
Id Number: 23042 License: P18581
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR414
Colors: Gray
Year: 0 Make: International
Id Number: 172 License: P18036
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR415
Colors: Gray
Year: 0 Make: International
Id Number: 155 License: P18019
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR418
Colors: Gray
Year: 0 Make: Freightliner
Id Number: 1425 License: P18360
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR419
Colors: Gray
Year: 0 Make: International
Id Number: 22930 License: P17936
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR420
Colors: Gray
Year: 0 Make: Freightliner
Id Number: 1443 License: P18648
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR421
Colors: Gray
Year: 0 Make: Freightliner
Id Number: 1478 License: P114714
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR422 Colors: Gray
Year: 0 Make: International
Id Number: 23037 License: P18576
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR423 Colors: Gray
Year: 0 Make: Freightliner
Id Number: 1428 License: P18633
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR424 Colors: Gray
Year: 0 Make: Freightliner
Id Number: 1464 License: P115034
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR425 Colors: Gray
Year: 0 Make: Freightliner
Id Number: 23029 License: P18568
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR426 Colors: Gold
Year: 0 Make: International
Id Number: 4855 License: P18341
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: 80 DEGREES AND SUNNY
Flag:

AR427 Colors: Gray
Year: 0 Make: International
Id Number: 419 License: P18235
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: 80 DEGREES AND SUNNY
Flag:

AR428
Colors: Gray
Year: 0 Make: International
Id Number: 22955 License: P18494
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: 80 DEGREES AND SUNNY
Flag:

AR429
Colors: Gray
Year: 0 Make: International
Id Number: 23047 License: P18586
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: 80 DEGREES AND SUNNY
Flag:

AR430
Colors: Gray
Year: 0 Make: Freightliner
Id Number: 1498 License: P98927
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: 80 DEGREES AND SUNNY
Flag:

AR431
Colors: Gold
Year: 0 Make: International
Id Number: 4065 License: P18264
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: 80 DEGREES AND SUNNY
Flag:

AR432
Colors: Gray
Year: 0 Make: Freightliner
Id Number: 1490 License: P17984
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: 80 DEGREES AND SUNNY
Flag:

AR433
Colors: Gray
Year: 0 Make: International
Id Number: 256 License: P18119
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: 80 DEGREES AND SUNNY
Flag:

AR434 Colors: Gray
Year: 0 Make: International
Id Number: 218 License: P18081
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: 80 DEGREES AND SUNNY
Flag:

AR435 Colors:
Year: 0 Make: International
Id Number: 6585 License: P18424
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR436 Colors:
Year: 0 Make: International
Id Number: 4295 License: P18287
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR438 Colors: Gold
Year: 0 Make: International
Id Number: 5010 License: P18357
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR439 Colors: Gray
Year: 0 Make: International
Id Number: 23020 License: P18559
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR440 Colors: Gray
Year: 0 Make: International
Id Number: 212 License: P18075
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR441
Colors: Gray
Year: 0 Make: Freightliner
Id Number: 1447 License: P18652
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR442
Colors: Gray
Year: 0 Make: International
Id Number: 22958 License: P18497
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR443
Colors:
Year: 0 Make: International
Id Number: 283 License: P18145
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR444
Colors: Gray
Year: 0 Make: Freightliner
Id Number: 1476 License: P17986
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR445
Colors:
Year: 0 Make: International
Id Number: 4545 License: P18310
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR446
Colors: Gray
Year: 0 Make: International
Id Number: 1458 License: P18663
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR448 Colors: Gray ,
Year: 0 Make: International
Id Number: 23019 License: P18558
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR449 Colors: Gray ,
Year: 0 Make: International
Id Number: 351 License: P18168
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR450 Colors: Gray ,
Year: 0 Make: Freightliner
Id Number: 1435 License: PK6640
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR451 Colors: Gray ,
Year: 0 Make: Freightliner
Id Number: 1534 License:
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR452 Colors: Gray ,
Year: 0 Make: Freightliner
Id Number: 1546 License: P115055
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR488 Colors: Gray ,
Year: 0 Make: International
Id Number: 22960 License: P18499
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: 80 DEGREES AND SUNNY
Flag:

AR489 Colors: Gray
Year: 0 Make: International
Id Number: 238 License: P18101
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: 80 DEGREES AND SUNNY
Flag:

AR490 Colors: Gray
Year: 0 Make: International
Id Number: 299 License: P18161
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: 80 DEGREES AND SUNNY
Flag:

AR491 Colors: Gold
Year: 0 Make: International
Id Number: 590 License: P18446
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: 80 DEGREES AND SUNNY
Flag:

AR492 Colors: Gray
Year: 0 Make: White
Id Number: 374 License: P18191
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: 80 DEGREES AND SUNNY
Flag:

AR493 Colors: Gray
Year: 0 Make: International
Id Number: 23034 License: P18573
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: 80 DEGREES AND SUNNY
Flag:

AR494 Colors: Gray
Year: 0 Make: International
Id Number: 23006 License: P18545
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: 80 DEGREES AND SUNNY
Flag:

AR495 Colors: Gold ,
Year: 0 Make: International
Id Number: 478 License: P18334
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: 80 DEGREES AND SUNNY
Flag:

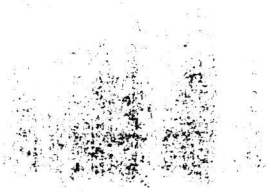
AR496 Colors: Gray ,
Year: 0 Make: International
Id Number: 255 License: P18118
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

AR497 Colors: ,
Year: 0 Make: International
Id Number: 405 License: P18263
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: 80 DEGREES AND SUNNY
Flag:

AR777 Colors: Gold ,
Year: 0 Make: International
Id Number: 4905 License: P18348
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments:
Flag:

TX034 Colors: ,
Year: 0 Make: International
Id Number: 455 License: P18311
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: 80 DEGREES AND SUNNY
Flag:

TX035 Colors: ,
Year: 0 Make: International
Id Number: 414 License: P18272
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: 80 DEGREES AND SUNNY
Flag:



TX036 Colors: Gray ,
Year: 0 Make: International
Id Number: 22975 License: P18514
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: 80 DEGREES AND SUNNY
Flag:

X074 Colors: Gray ,
Year: 0 Make: International
Id Number: 244 License: P18107
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: 80 DEGREES AND SUNNY
Flag:

TX075 Colors: Gold ,
Year: 0 Make: International
Id Number: 509 License: P98284
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: 80 DEGREES AND SUNNY
Flag:

TX077 Colors: Gray ,
Year: 0 Make: Freightliner
Id Number: 1460 License: P18665
Permits: Overweight: N Hazordus: N
Fees: Highway Use: N Fuel Tax: N
Comments: 80 DEGREES AND SUNNY
Flag:

BAR CODE

READING OF

MOVING TRUCKS

PHASE II

TRC-95

Prepared by
University of Arkansas, Fayetteville
College of Engineering
Center for Robotics and Automation

and the
Arkansas State Highway and Transportation Department

in cooperation with
U.S. Department of Transportation
Federal Highway Administration

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
FHWA/AR-89/001		

4. Title and Subtitle	5. Report Date
BAR CODE READING OF MOVING TRUCKS, PHASE II, FINAL REPORT	February 1988
	6. Performing Organization Code
	8. Performing Organization Report No.
7. Author(s)	
Marcus C. Langston	
9. Performing Organization Name and Address	10. Work Unit No. (TRAIS)
College of Engineering Center for Robotics and Automation University of Arkansas Fayetteville, AR 72701	11. Contract or Grant No. TRC-95
	13. Type of Report and Period Covered
12. Sponsoring Agency Name and Address	PHASE II Final Report
Arkansas State Hwy and Transportation Dept. P. O. Box 2261 Little Rock, AR 72203	14. Sponsoring Agency Code TRC-95

15. Supplementary Notes

This study was conducted in cooperation with the Arkansas Highway and Transportation Department and the U.S. Department of Transportation, Federal Highway Administration.

16 Abstract

An eighteen month research project determined that the use of bar codes is a feasible means of automatically identifying commercial vehicles. During initial field testing, this system proved capable of providing a first time read rate of over 75% of the vehicles which passed by the scanner.

Results from Phase I indicated the need for additional research in two areas prior to full scale implementation:

1. Maximization of system hardware to achieve above a 95% first-pass read rate,
2. Research to provide a bar code emblem which will last the anticipated life of the truck.

This report details the results of research in the above two areas, conducted between June 1, 1987 and November 30, 1987.

17. Key Words	18. Distribution Statement
Bar Code Automatic Vehicle Identification (AVI) Laser Reader	

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
UNCLASSIFIED	UNCLASSIFIED	33	

Form DOT F 1700.7 (8-72) Reproduction of completed page authorized

BAR CODE READING OF MOVING TRUCKS, PHASE II,

TRC-95

Prepared by
University of Arkansas, Fayetteville
College of Engineering
Center for Robotics and Automation

and the
Arkansas State Highway and Transportation Department

in cooperation with
U.S. Department of Transportation
Federal Highway Administration

BAR CODE READING OF MOVING TRUCKS

PHASE II

FINAL REPORT

Transportation Research Project No. 95

February, 1988

by

University of Arkansas at Fayetteville
Center for Robotics and Automation

and the

Arkansas State Highway and Transportation Department

In Cooperation

with the

Federal Highway Administration

Marcus C. Langston,
Principal Investigator

GAINS, FINDINGS AND CONCLUSIONS

The primary gains, findings, and conclusions of this study are as follows:

I. Gains and Findings

1. An "off the shelf" laser bar code reader was developed which successfully read 99.63% of the time under simulated weigh station operating conditions.

2. Laminating materials were identified and tested which extended the useful life of standard bar codes to over five years (the anticipated life of a commercial vehicle).

II. Conclusions

1. The researchers responsible for this report have no reservations as to the technological feasibility of automatic vehicle identification (AVI) using bar code technology.

2. Bar coding appears to be a cost effective, practical solution to accomplish automatic vehicle identification at weigh stations. The AVI system developed employs "off the shelf" existing technology. The hardware cost for each AVI station is estimated to be less than $9,000. The unit cost to bar code each vehicle is estimated to be $0.80 (when conducted on a large scale).

TABLE OF CONTENTS

LIST OF FIGURES AND TABLES

INTRODUCTION

An eighteen month research project conducted by the University of Arkansas at Fayetteville and the Arkansas State Highway and Transportation Department determined that the use of bar codes is a feasible means of automatically identifying commercial vehicles. During initial field testing, this system proved capable of providing a first time read rate of over 75% of the vehicles which passed by the scanner.

Results from Phase I indicated the need for additional research in two areas prior to full scale implementation:

1) Maximization of system hardware to achieve above a 95% first-pass read rate,
2) Research to provide a bar code emblem which will last the anticipated life of the truck.

This report details the results of research in the above two areas, conducted between June 1, 1987 and November 30, 1987. The research effort was a joint program conducted by the Arkansas State Highway Department and the Center for Robotics and Automation, University of Arkansas, Fayetteville in cooperation with the Federal Highway Administration.

SUMMARY OF TECHNICAL FINDINGS

Scanner and Emblem Modifications to Enhance Readability

Modifications to the signal processing electronics and the housing unit of the laser bar code reader, coupled with new materials used for the bar code emblem, produced a system which tested to be highly accurate. The testing was conducted in one of the Robotics Center's laboratories which was modified to simulate a truck passing a weigh station. A rotating motion machine was designed and built to move codes past the scanner at speeds of up to 37 MPH (Figure 1). Banks of high intensity lights were installed to simulate sunlight. Reading at a distance of 84 inches (7 ft.) and at a speed of 35 miles per hour, under (simulated) bright sunlight, the system successfully read 99.63% of the time (30 no reads in 8,000 tries).

Extended Bar Code Life

Investigations were conducted to identify plastic materials that could be used as protective sheeting for the bar codes. Three potential candidates were identified and tested. Two of the three materials successfully passed a 1000 hour advanced weatherization test with no apparent physical deterioration.

Preliminary Data Base and Hardware Configuration

A microcomputer system for retrieving and storing information was specified and purchased. The system is capable of storing the information necessary to process 467,000 vehicles, before having to access a larger host data base.

BAR CODE SYSTEM COMPONENT PARTS

This section of the report presents specific technical aspects of the research conducted.

I. Bar Code Scanner - Two modifications to the existing bar code scanner were made during the research effort. Previous research identified a problem with code reading in bright sunlight. The objective of the modifications were to increase the signal which the scanner received from the codes while reducing the masking effects bright sunlight had upon the system.

1) Changes in laser scanner housing design - One modification concerned the "birdhouse" which protected the scanner from the weather. In an effort to reduce the amount of ambient light which reached the scanner a "snout" device was designed and built onto the front of the birdhouse to reduce the field of vision of the scanner. This simple design modification increased the effectiveness of the scanner in bright sunlight. However, the use of the snout poses a problem of having to read the bar codes at a greater distance because the scanner must be setback from the roadway the length of the snout.

2. Electronic Modification - the second modification to the scanner system concerned a change in the circuity of the signal processing module of the scanner. The researchers

found that in switching to the new bar code backing material too much (reflected) light was reaching the photosensor to be properly processed. Modification to the scanner was completed by the manufacturer of the equipment.

II. Bar Code Labels - A redesign of the bar code labels was completed during the research. New backing materials and lamina was tested for readability and durability.

1) Backing Material - New material for construction of the bar codes was identified, purchased and tested. Research identified a retroreflective material commercially available as the most suitable backing material.

2) Lamina Materials - Several new "space age" polymer materials were identified as possible laminating materials. Upon evaluation of what materials were commercially available, three potential products were chosen due to their resistance to ultra violet rays and potential durability in the field. Comprehensive testing was conducted on these three materials. One of the three materials did not pass the advanced weatherization test and was eliminated from further consideration. The section on bar code testing contains additional information of this subject.

TEST PROCEDURES

During previous studies it was noted numerous factors have an influence on the read rate of the bar code system. These variables can be grouped into one of four categories:

1) Factors effecting the laser scanner, i.e. power, scan rate, unit in which the scanner is housed, etc.
2) Factors effecting the bar code, i.e. size of the printed bar code (min. bar width), width of code, type of backing material, type of lamina, etc.
3) Factors concerning the environment, i.e. the intensity of sunlight present, angle of sunlight, rain, snow, fog, etc.
4) Factors effecting the truck on which the bar code is to be applied, i.e. the speed of the truck, distance from the scanner, amount of dirt on the code, etc.

In order to quantitavely evaluate the numerous factors effecting the system read rate, a scientific design of experiment was conducted. The factors effecting the read rate were evaluated and prioritized. The following variables were identified and tested: **Minima Bar Width, Type of Lamina, Size of Code, Skew Placement of Emblem, Speed of "Truck", and Distance Between Scanner and Code.**

A rotary member was designed, built, and used to simulate various truck speeds. Figure 1 shows the configuration of the rotating member and its interface to the test computer.

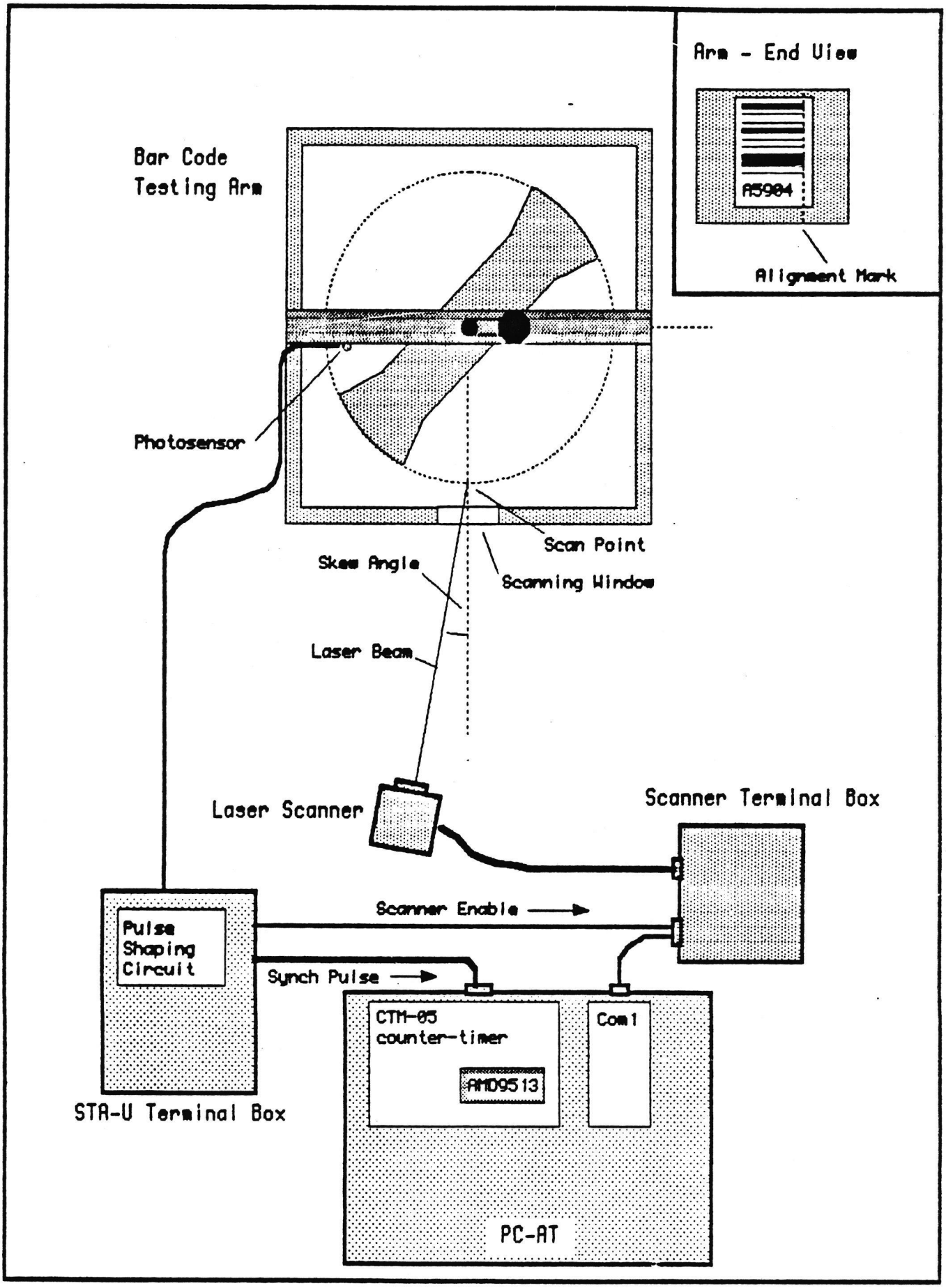


Figure 1 - Testing Equipment

A computer program was developed to record the speed of the rotating arm and accept inputs from the bar code reader. A photodetector sensor attached to the test device sent a pulse to the computer upon each successive revolution of the arm. Additionally the sensor sent a scanner enable pulse to the bar code reader (signaling the scanner to begin reading). The interface of a AMD9513 counter/timer board with the personnel computer allowed the calculation of the speed of the rotating arm. Once the bar code reader received the enable pulse it signaled the test computer of a good read or a bad read. In this fashion various independent variables were tested as to their effect upon the dependant variable - percent good reads.

Table 1 - Bar Code Testing Variables:

INDEPENDENT	DEPENDENT
1. Minimum Bar width (40, 50, 60 mils)	1. Readability of labels (percentage of good reads)
2. Type of Lamina	
3. Size of Code	
4. Skew Placement of Emblem	
5. Distance to Bar Code	
6. Speed of Bar Code	
7. Pitch and Skew of Laser Scanner	

Table 2 - Material Testing Variables:

INDEPENDENT	DEPENDENT
1. Hours in QUV Tester	1. Readability of labels
2. Number of Washings	2. Visual Inspection for Damage

Readability Tests

Figure 3 graphically shows the effects of the variables which most effect the system, e.g. distance between scanner and code, speed of code, and minimum bar width.

The plots are based upon 62,000 test runs:

a) 15500 samples of lamina 1 - 60 mil codes
b) 15500 " " lamina 2 - 50 mil codes
c) 15500 " " lamina 1 - 60 mil codes
d) 15500 " " lamina 2 - 50 mil codes

It should be noted the original design of experiment called for the testing of three types of lamina and three sizes of bar codes e.g. 40, 50, and 60 mil (minimum bar width). During preliminary testing it was determined the read rate of the 40 mil code was below 70% at a distance of seven feet. Therefore the 40 mil code was rejected. Similarly, one of the three lamina materials chosen for the study did not pass the 1000 hour advanced weatherization test and was eliminated from further consideration.

Special bar codes with no laminating material were printed and tested. The results of the test provided reference data concerning the effects of laminating material vs. read rate.

Environmental Testing of Codes

The bar code emblems produced for this project were tested in an environmental testing chamber to determine the long term effects of weathering upon the various laminating materials.

Environmental Testing of Codes Cont.

Mr. George Green section head of the Chemistry Laboratory, Materials and Research Division of Arkansas State Highway Department was responsible for the environmental testing.

The machine in which the codes were tested is a QUV Accelerated Weathering Tester, which is made by the Q-Panel Co. The testing method is the one recommended in ASTM G53. The test involves a continuous cycle of lights on or off and condensation on or off:

1. Four (4) hours of UV Lights ON, Condensation OFF, unit temperature at 60 deg. C.
2. Four (4) hours of Condensation ON, UV Lights OFF, unit temperature at 50 deg. C.
3. Repeat cycle, continuously.

Test codes were exposed at intervals of 250, 500, 750, and 1000 hours each. Upon removal from the QUV tester the codes were inspected for signs of material deterioration and tested for readability. As mentioned previously, one of the three test lamina did not successfully pass the 1000 hour test.

Wash-Testing of Codes

Figure 2 is a block diagram of the wash-test set-up designed and built to simulate the repeated washings commercial vehicles are subjected. The wash test was comprised of emerging the test bar code in a solution of commercial truck wash detergent and repeatedly activating an air cylinder which had a brush attached to its end. All three test codes successfully passed 10,000 repetitions of this test with no visible damage.

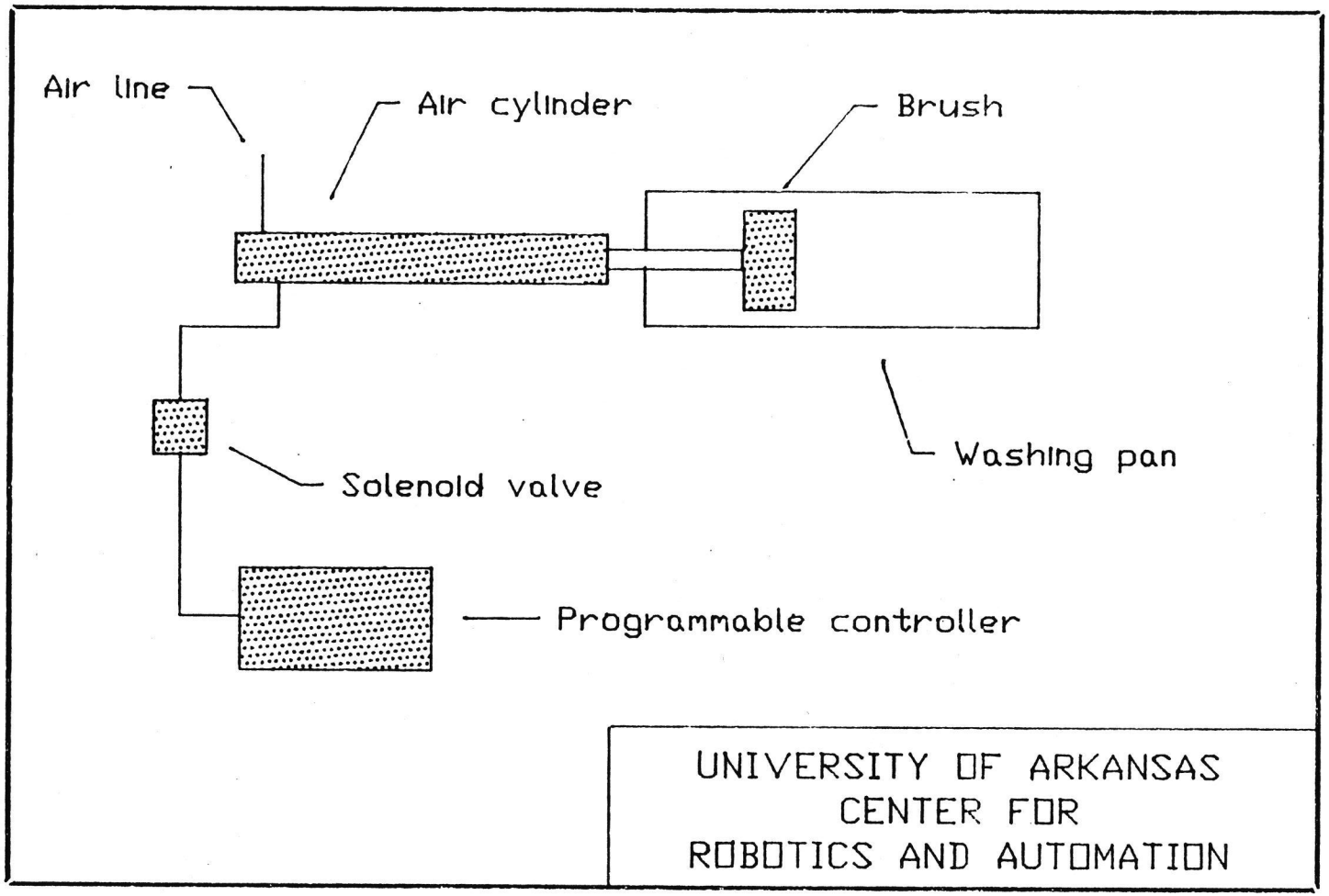


Figure 2: Block Diagram Of Wash-Test Set-Up

TEST RESULTS

A) Effect of Minimum Bar Width vs. Readability

1) The read rate of the 50 mil bar codes is greater than 95% at distances less than seven (7) feet and speeds of up to twenty five miles per hour.
2) The read rate of the 60 mil bar codes is greater than 98% at distances less than eight feet and speeds up to twenty five miles per hour.

B) Effect of Speed vs. Readability (tested at 7 feet)

1) The 60 mil bars were readable at speeds up to 45 MPH (read rate above 97%).
2) The 50 mil bars were readable at speeds up to 20 MPH, however at higher speeds the read rate dropped rapidly.

C) Effect of Laminating Material on Readability

Tests on bar codes with different lamina indicate that the read rate is independent of the lamina type studied.

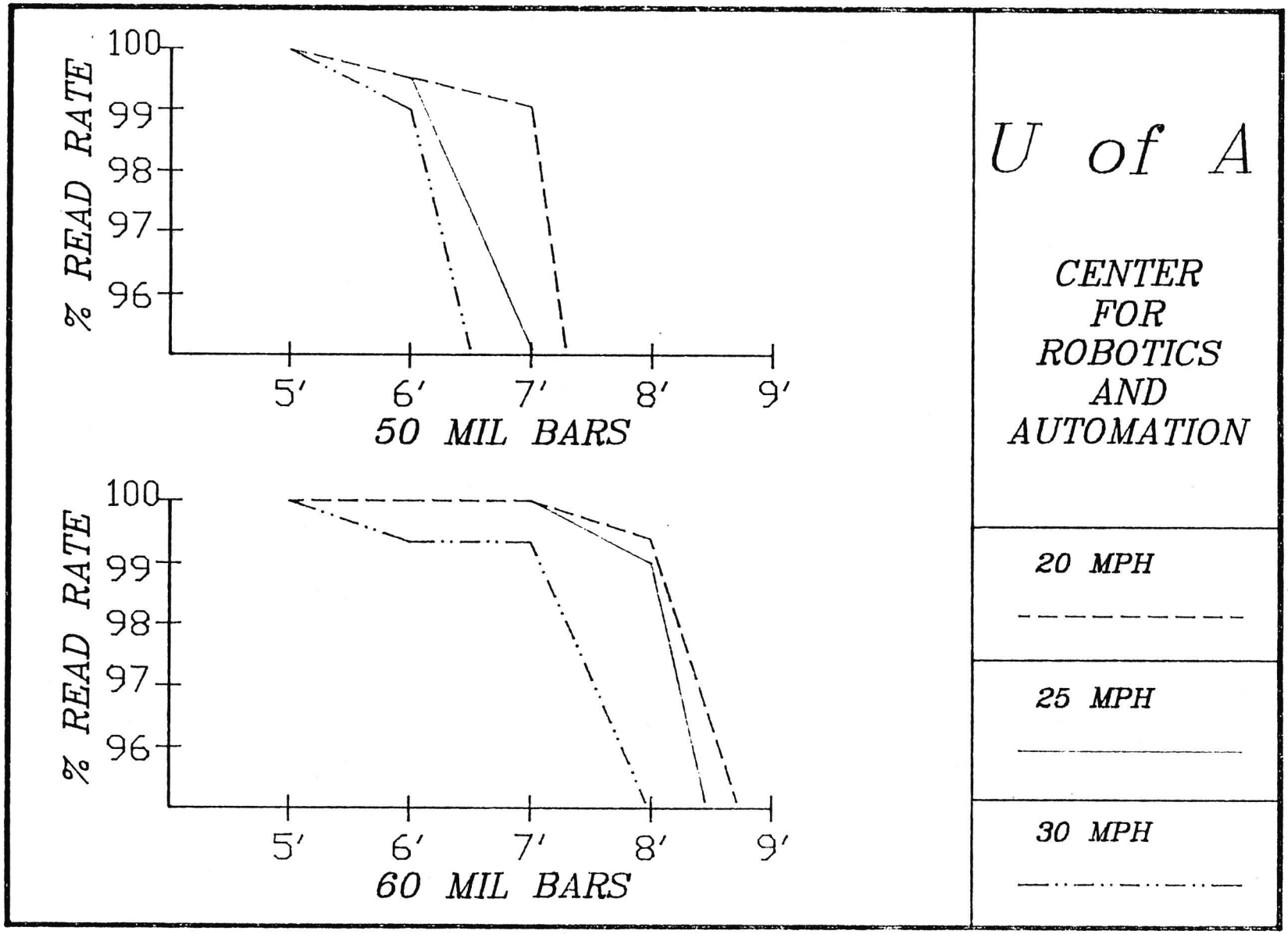


Figure 3 - Effects of Speed & Code Size on Readability

RESEARCH WORK COMPLETED

This project was conducted in accordance with the Procedural Manual for Transportation Research Projects and under the terms of the Basic Agreement. The proposed plan of work was to proceed in four different areas:

A - Weigh Station Modifications

B - Bar Code Label Studies

C - Scanner Research

D - System Review and Analysis.

Following is a detailed description of the work completed with comments on the timing, problems encountered, and suggestions for additional study.

A. Weigh Station Modifications:

1) Define system hardware requirements - Task 100% complete, performed as planned.

2) Write specifications for microcomputer and bid computer - Task 100% complete, performed as planned (see Appendix A).

3) Write software to support system integration between electronic weigh-in-motion scales, bar code scanner, and microcomputer data base - Task 75% complete (see Appendix D) . Items to be completed are: a) the actual interface between the bar code system and the system which controls the weigh in motion scales, completion will require coordination between U of A personnel and engineers responsible for installation of W.I.M. scales; b) definition of exactly what information is to be tracked by system.

4) Install microcomputer - Completed as planned

B. Bar Code Label Studies

1) Design experiment to determine the significance of the various variables associated with the reading of the bar codes: 100% complete, as planned. As previously noted the significant variables identified were: minimum bar width, speed of code, and distance between code and scanner. Appendix C contains a copy of this experiment.
2) Find various new lamina - 100% completed, but a two week delay was experienced in the completion of this task due to difficulty locating a supplier of a new lamina material.
3) Test lamina in AHTD UV tester - planned work 100% completed, however due to the extraordinary long life of these materials; additional testing is suggested.
4) Test lamina for resistance to multiple washings - work 100% completed. Test codes were subjected to 10,000 washings without physical deterioration.
5) Determine most effective bar code symbology for reading outdoors - 100% completed. Difficulties were encountered completing this task. The reason for the difficulties centered around the complexity associated with developing a device which would (safely) generate speeds of up to 35 MPH in a laboratory and the associated problems of "capturing" and analyzing this data in a timely manner. Eventually a suitable device was designed and built.
6) Determine material specifications for production of bar codes - Planned work was completed, but it is suggested additional testing on extreme heat and cold be conducted before final recommendations are made.

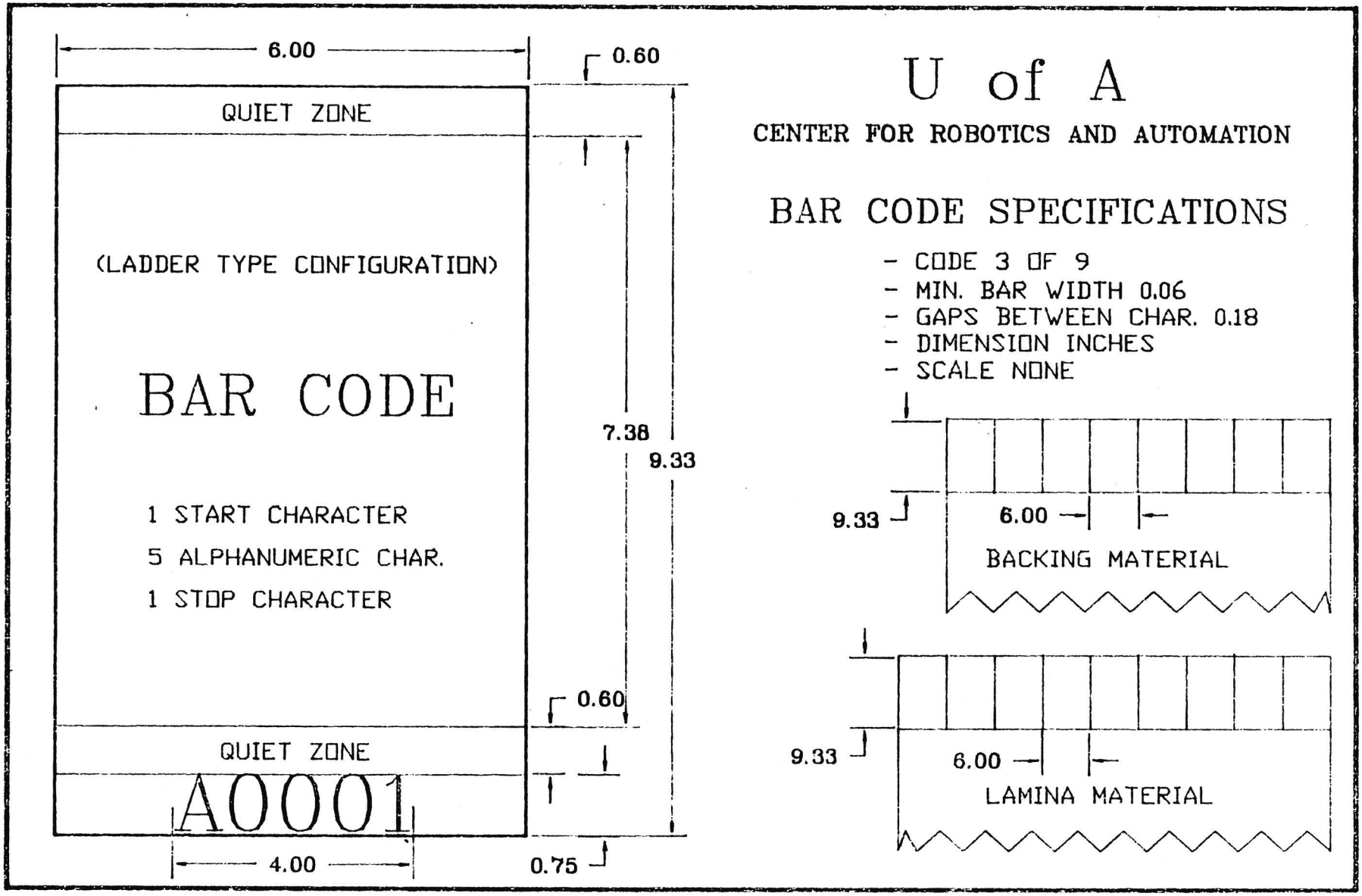


Figure 4 – Bar Code Specifications

7) Write specifications for bar codes - 95% complete, specifications concerning the size, type of printing, and the symbology of codes are complete as shown in Figure 4. Some material specifications will require additional study.

C. Scanner Research

1) Develop a more powerful laser for system operation in direct sunlight - work completed as planned.
2) Write functional specifications for laser and bid specifications - 100% complete, as planned (see Appendix B).
3) Test readability of test codes to determine system reliability - 100% complete. The first set of test bar codes emblems provided by the printers were rejected due to poor quality. Correcting the quality problem and reprinting the codes delayed this task six weeks.

D. System Review and Analysis

1) Comprehensive review of data collected to date - 100% complete, as planned.
2) Analyze data collected - work 100% completed as planned.
3) Conduct survey to determine the response of trucking companies to the concept of bar code vehicle identification - 30% complete, draft letters have been written, and trucking firms identified. Due to the initiation of the controversial Weight-Distance Tax during the time this task was to be completed, truck companies were not contacted to avoid having this system associated with "weight-distance" taxing.

4) Determine which variables most affect the operation of the system - 85% complete, extensive indoor testing has been completed. Operating under the premise that a code which is maximized for readability indoors (under simulated outdoor conditions) will read correspondingly well outside. However, this hypothesis needs to be verified and will require some additional testing outdoors.
5) Prepare preliminary report on system installation, 100% complete.

All of the work scheduled for Phase II was completed. The project was completed $460.00 under budget.

CONCLUSIONS

The objectives of this research project was to provide quantitative data concerning system modification to achieve the following:

1) Maximization of system hardware to achieve above a 95% first-pass read rate,
2) Research to provide a bar code emblem which will last the anticipated life of the truck.

The first objective was reached by making two modifications to the laser scanner: a) improving the optics of the scanner which processes the signal reflected back from the bar codes, and b) installing a "snout" onto the housing in which the laser was contained. These modifications had the effect of decreasing the signal to noise ratio encountered by the scanner in sunlight. The percent of good reads for the system (in bright sunlight) was increased from 75% to over 99%.

The second objective was reached by conducting research on new materials for the bar code emblems. A new base material was identified and tested. This retro-reflective material proved to read well and displayed adequate durability characteristics. New laminating materials were identified and tested. Two of the laminating materials passed comprehensive advanced weatherization and wash tests. Testing showed the lamina did not effect the readability of the codes.

Appendix A

Phase II Computing Equipment Specifications

PC/AC clone (Zenith ZW-284-84)

* 80286 CPU
* EGA card
* 40 meg. hard disk
* 1 parallel port
* 1 serial port

EGA monitor (Zenith ZVM-1380)

Appendix B

Specifications

Phase II Laser Scanning Equipment

Moving-beam Laser Scanner (Insta Read 610)

* Class II HeNe Laser
* Fixed-mount moving-beam scanner (400 scans/sec.)
* Max. scanning range 6-10 ft. (code & optics dependent)
* Reads Code 39
* Operates in external enable mode
* RS-232 Serial output

Appendix C

Design of Experiment

and

Test Data

A. H. T. D. BAR CODE SCANNING PROJECT

EXPERIMENT I - DATA SHEET

No.	TRIAL NO.	SPEED BEFORE/AFTER	GOOD READS	BAD READS	MIS-READS	TIME & DATE	TESTER
1	7	14.98 / 14.95	799	201	—	7:30PM 2-2-1988	m.H
2	10	15.05 / 15.03	831	169	—	7.45	s
3	14	15.46 / 14.96	1000	—	—	8.05	s
4	8	15.12 / 15.07	1000	—	—	8.16	s
5	14	15.25 / 15.07	1000	—	—	8.36	s
6	3	15.14 / 15.01	729	271	—	8.47	s
7	2	15.08 / 15.01	215	785	—	9.09	s
8	13	15.41 / 15.06	1000	—	—	9.22	s
9	10	15.16 / 14.90	936	64	—	11.45PM 2-3-1988	s
10	4	15.01 / 14.88	1000	—	—	11.57	s
11	6	14.83 / 14.78	1000	—	—	12.02	s
12	5	15.40 / 15.09	1000	—	—	12.15	s
13	11	15.35 / 15.20	887	113	—	12.28	s
14	5	15.39 / 15.20	1000	—	—	12.40	s
15	5	15.20 / 15.14	1000	—	—	1.05	s

* Experiment I: Testing 3 emblem variables to determine the emblem specifications which will give the best results.

A. H. T. D. BAR CODE SCANNING PROJECT

EXPERIMENT I - DATA SHEET

NO.	TRIAL NO.	SPEED BEFORE/AFTER	GOOD READS	BAD READS	MIS-READS	TIME & DATE	TESTER
16	6	15.20 / 15.04	1000	—	—	1.15	
17	9	15.68 / 15.33	1000	—	—	1.30	
18	10	15.46 / 15.17	856	144	—	1.43	
19	1	15.39 / 15.21	675	325	—	2.00	
20	3	15.27 / 15.12	764	236	—	2.15	
21	7	15.04 / 15.12	781	219	—	2.30	
22	4	15.a / 15.21	1000	—	—	2.45	
23	6	14.99 / 15.15	1000	—	—	3.03	
24	12	14.98 / 14.86	1000	—	—	3.15	
25	6	15.05 / 15.20	1000	—	—	3.28	
26	9	15.0 / 15.15	1000	—	—	3.40	
27	3	14.91 / 14.90	755	245	—	3.55	
28	4	14.95 / 15.20	1000	—	—	4.05	
29	14	15.0 / 14.88	1000	—	—	4.18	
30	7	14.94 / 14.78	763	237	—	4.30	

* Experiment I: Testing 3 emblem variables to determine the emblem specifications which will give the best results.

A. H. T. D. BAR CODE SCANNING PROJECT

EXPERIMENT I - DATA SHEET

NO.	TRIAL NO.	SPEED BEFORE/AFTER	GOOD READS	BAD READS	MIS-READS	TIME & DATE	TESTER
31	9	14.93 / 15.0	1000	—	—	4.42	MH
32	3	14.92 / 14.58	678	322	—	4.54	
33	1	15.25 / 15.13	724	276	—		
34	11	15.55 / 15.37	850	150	—		
35	13	15.47 / 15.40	1000	—	—		
36	12	15.25 / 15.14	1000	—	—		
37	2	15.37 / 15.24	212	788	—		
38	2	15.20 / 15.14	244	756			
39	1	15.02 / 14.85	645	355	—	5:52 2/3/88	MH
40	6	14.95 / 15.03	1000	—		6:05	MH
41	13	15.10 / 15.00	1000	—		6:15	MH
42	5	14.96 / 14.92	1000	—		6:28	MH
43	8	15.09 / 15.10	1000	—		6:37	MH
44	1	15.02 / 14.89	646	354	—	6:49	MH
45	11	15.04 / 14.80	847	153		7:00	MH

* Experiment I: Testing 3 emblem variables to determine the emblem specifications which will give the best results.

Appendix D

Computer Interface Program

(Written in compiled basic)

```
' ---------------

DEF FNUnsign (Intgr%)
    IF Intgr% < 0 THEN
        FNUnsign = Intgr% + 65536
    ELSE
        FNUnsign = Intgr%
    END IF
END DEF

DEF FNSign% (Real)
    IF Real < 32768 THEN
        FNSign% = Real
    ELSE
        FNSign% = Real - 65536
    END IF
END DEF

' ---------------
' Initialization routine

BaseAddr% = &H300
IntLvl% = 3
BarCode$ = "A5904"
NoRead$ = "XXXXX"

OPEN "com1:600,e,,,rs,ds0" FOR RANDOM AS 1

ON COM(1) GOSUB CommTrap
ON KEY(1) GOSUB RequestTrap
ON KEY(2) GOSUB ExitTrap

KEY 1, "NewRun"
KEY 2, "Exit"

CLS
PRINT "           Rim speed:"
PRINT
PRINT "    Scans requested:"
PRINT "    Scans completed:"
PRINT
PRINT "   Current bar code: "; BarCode$
PRINT
PRINT "         Good reads:"
PRINT "          Bad reads:"
PRINT "        Other reads:"
PRINT "      Missing reads:"

COM(1) ON

CALL Install(BaseAddr%, IntLvl%)

KEY ON
KEY(1) ON
KEY(2) ON

GoodReads = 0
BadReads = 0
OtherReads = 0
```

```
' ----------------------
' main loop

DO
    DO
        CALL Status(Flag%, Period%, ScansLeft%)
    LOOP UNTIL Flag%

    Period = FNUnsign(Period%)

    COM(1) STOP
    LOCATE 1, 21
    IF Period > 1 THEN
        PRINT USING "###.## mph"; 64260 / Period
    ELSE
        PRINT " <0.98 mph"
    END IF

    ScansDone = Scans - FNUnsign(ScansLeft%)
    MissingReads = ScansDone - (GoodReads + BadReads + OtherReads)

    LOCATE 4, 22: PRINT USING "#####"; ScansDone
    LOCATE 11, 22: PRINT USING "#####"; MissingReads
    COM(1) ON

LOOP
' ---------------
RequestTrap:

LOCATE 13, 20
INPUT "Number of scans requested"; Scans
Scans% = FNSign%(Scans)

LOCATE 3, 22: PRINT USING "#####"; Scans

GoodReads = 0
BadReads = 0
OtherReads = 0

LOCATE 8, 22: PRINT USING "#####"; GoodReads
LOCATE 9, 22: PRINT USING "#####"; BadReads
LOCATE 10, 22: PRINT USING "#####"; OtherReads
LOCATE 13, 1: PRINT TAB(70);

CALL Request(Scans%)

RETURN
' ----------------
CommTrap:

n = LOC(1)
IF n < 5 THEN RETURN
Scan$ = INPUT$(5, 1)

IF Scan$ = BarCode$ THEN
    GoodReads = GoodReads + 1
    LOCATE 8, 22: PRINT USING "#####"; GoodReads
ELSEIF Scan$ = NoRead$ THEN
    BadReads = BadReads + 1
    LOCATE 9, 22: PRINT USING "#####"; BadReads
```

```
ELSE
    OtherReads = OtherReads + 1
    LOCATE 10, 22: PRINT USING "#####"; OtherReads
END IF

MissingReads = ScansDone - (GoodReads + BadReads + OtherReads)
LOCATE 11, 22: PRINT USING "#####"; MissingReads

RETURN
' ----------------
ExitTrap:

CALL Remove
CLS
END
```