

AUTOMATED TRANSIT RIDERSHIP DATA COLLECTION
Pilot Test and User's Guide

By
Kirk E. Barnes
Assistant Research Scientist
and
Thomas Urbanik II
Research Engineer

Technical Report Number 1087-2
Study Number 2-11-87-1087

Sponsored by
Texas State Department of Highways and Public Transportation

In cooperation with
Urban Mass Transportation Administration
U.S. Department of Transportation

Texas Transportation Institute
The Texas A&M University System
College Station, Texas 77843-3135

September 1989

The preparation of this study was financed in part through a grant from the Urban Mass Transportation Administration, United States Department of Transportation under the Urban Mass Transportation Act of 1964, as amended.

METRIC (SI*) CONVERSION FACTORS

APPROXIMATE CONVERSIONS TO SI UNITS

Symbol	When You Know	Multiply By	To Find	Symbol
LENGTH				
in	inches	2.54	millimetres	mm
ft	feet	0.3048	metres	m
yd	yards	0.914	metres	m
mi	miles	1.61	kilometres	km

AREA				
in ²	square inches	645.2	millimetres squared	mm ²
ft ²	square feet	0.0929	metres squared	m ²
yd ²	square yards	0.836	metres squared	m ²
mi ²	square miles	2.59	kilometres squared	km ²
ac	acres	0.395	hectares	ha

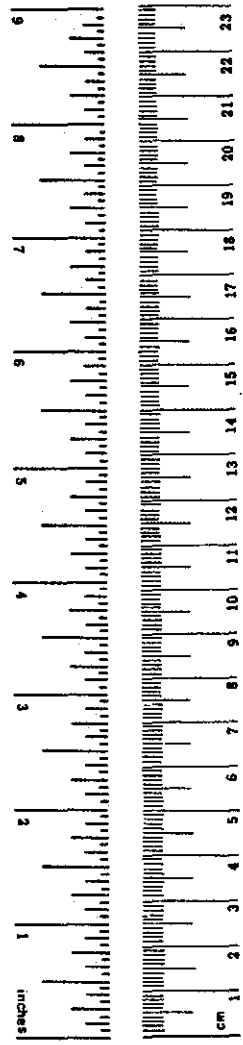
MASS (weight)				
oz	ounces	28.35	grams	g
lb	pounds	0.454	kilograms	kg
T	short tons (2000 lb)	0.907	megagrams	Mg

VOLUME				
fl oz	fluid ounces	29.57	millilitres	mL
gal	gallons	3.785	litres	L
ft ³	cubic feet	0.0328	metres cubed	m ³
yd ³	cubic yards	0.0765	metres cubed	m ³

NOTE: Volumes greater than 1000 L shall be shown in m³.

TEMPERATURE (exact)				
°F	Fahrenheit temperature	5/9 (after subtracting 32)	Celsius temperature	°C

* SI is the symbol for the International System of Measurements



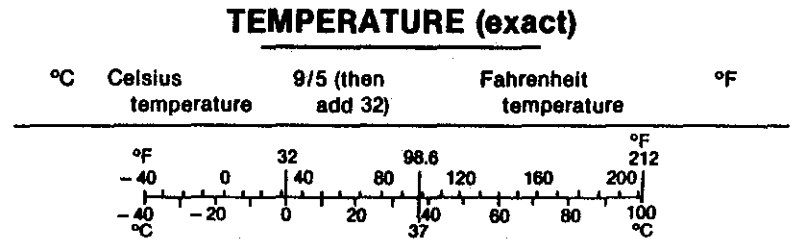
APPROXIMATE CONVERSIONS TO SI UNITS

Symbol	When You Know	Multiply By	To Find	Symbol
LENGTH				
mm	millimetres	0.039	inches	in
m	metres	3.28	feet	ft
m	metres	1.09	yards	yd
km	kilometres	0.621	miles	mi

AREA				
mm ²	millimetres squared	0.0016	square inches	in ²
m ²	metres squared	10.764	square feet	ft ²
km ²	kilometres squared	0.39	square miles	mi ²
ha	hectares (10 000 m ²)	2.53	acres	ac

MASS (weight)				
g	grams	0.0353	ounces	oz
kg	kilograms	2.205	pounds	lb
Mg	megagrams (1 000 kg)	1.103	short tons	T

VOLUME				
mL	millilitres	0.034	fluid ounces	fl oz
L	litres	0.264	gallons	gal
m ³	metres cubed	35.315	cubic feet	ft ³
m ³	metres cubed	1.308	cubic yards	yd ³



These factors conform to the requirement of FHWA Order 5190.1A.

IMPLEMENTATION STATEMENT

The study findings indicate that an automated data collection system is an effective means of collection, checking and summarizing bus ridership data. The data can be compiled and reported quickly so that analysis and operational decision making can be more timely. The concept of an equipment "pool" so that small transit agencies would have access to an automated data collection system is not feasible, data must be collected too frequently for such a program. The existing documentation of the automated system's software, was inadequate for implementation, so a supplemental user's guide was developed to aid in the setup and operation of the system. Field testing and then implementation of the automated data collection system into an actual transit agency demonstrated that such a system could be used as an effective means of reducing data collection costs, improving employee efficiency and aid in making scheduling changes.

DISCLAIMER

The contents of the report reflect the views of the authors who are responsible for the opinions, findings and conclusions presented herein. The contents do not necessarily reflect the official views of the Texas State Department of Highways and Public Transportation or the Urban Mass Transportation Administration. This report does not constitute a standard specification or regulation.

THE HISTORY OF THE UNITED STATES

The history of the United States is a complex and multifaceted story that spans centuries. It begins with the early Native American civilizations, such as the Mayans, Aztecs, and Incas, who built sophisticated societies in the Americas. The arrival of European explorers in the late 15th and early 16th centuries marked the beginning of a new era. The Spanish, French, and British established colonies and fought wars for control of the continent. The American Revolution (1775-1783) was a pivotal moment, leading to the birth of the United States as an independent nation. The Constitution was drafted in 1787, and the country grew through westward expansion, the Civil War (1861-1865), and the Reconstruction era. The 20th century saw the rise of the United States as a global superpower, with significant events like World War I, the Great Depression, and the Cold War. The country has continued to evolve, facing challenges like the Vietnam War, the Civil Rights Movement, and the 9/11 attacks. Today, the United States remains a major world power, with a rich and diverse cultural heritage.

The history of the United States is a complex and multifaceted story that spans centuries. It begins with the early Native American civilizations, such as the Mayans, Aztecs, and Incas, who built sophisticated societies in the Americas. The arrival of European explorers in the late 15th and early 16th centuries marked the beginning of a new era. The Spanish, French, and British established colonies and fought wars for control of the continent. The American Revolution (1775-1783) was a pivotal moment, leading to the birth of the United States as an independent nation. The Constitution was drafted in 1787, and the country grew through westward expansion, the Civil War (1861-1865), and the Reconstruction era. The 20th century saw the rise of the United States as a global superpower, with significant events like World War I, the Great Depression, and the Cold War. The country has continued to evolve, facing challenges like the Vietnam War, the Civil Rights Movement, and the 9/11 attacks. Today, the United States remains a major world power, with a rich and diverse cultural heritage.

ACKNOWLEDGMENTS

The Texas Transportation Institute would like to acknowledge the assistance and cooperation provided by the following individuals during this study. The study was done under the direction of Ed Collins of D-11, Texas State Department of Highways and Public Transportation, who was instrumental in identifying the opportunity for small transit systems to improve their data collection and analysis procedures. Mr. Bob Menhart of Multisystems Inc. provided the technical assistance needed during the setup of the system. Mr. Dusty Peters of Citibus in Lubbock, Texas provided his time and patience during the pilot testing of the data collection system.

SUMMARY

Transit systems need to collect bus ridership data to evaluate operational efficiency and to comply with the reporting requirements for federal government subsidies. Manual data collection, coding and summarization is a tedious, inefficient and expensive process. An automated data collection system such as the one produced by Multisystems was found to be the most cost effective and versatile system for Texas small transit agencies.

The automated system consists of two software packages. The Check*mate software is used on a portable computer to collect ridership data and the TIM software is used on a PC to correct, store and summarize the data.

The supplied documentation for the system was inadequate for implementation and so a supplemental user's guide was developed. The automated system (including the new user's guide) was furnished to Citibus, the local transit agency in Lubbock. The implementation and their use of the automated system was then documented. Several comments and recommendations were generated for the automated system. Overall, the pilot test demonstrated that the automated system could be used effectively by transit systems to improve efficiency.

The system does require considerable time to set up, and was one of the major concerns resulting from the pilot testing. State assistance in training and set-up may be one possible alternative for small transit agencies. TTI and Citibus personnel agree that the concept of an equipment "pool" for transit agencies to borrow the hardware necessary for an automated data collection system is not feasible due to the frequency at which data must be collected.

TABLE OF CONTENTS

	Page
Implementation Statement	iii
Disclaimer	iii
Acknowledgements	v
Summary	vii
Introduction	1
Objectives	1
Description	2
Field Test	5
Documentation	5
Implementation	5
Observations	7
Developments	9
Demonstration	10

TABLE OF CONTENTS (CONT.)

Conclusions and Recommendations	10
Appendix A	11

INTRODUCTION

In order to improve operating efficiency, transit administrators must make intelligent and informed decisions. Such decisions require extensive and accurate ridership data. Transit agencies also need ridership data to complete the numerous reports the federal government requires for operating subsidies. Depending on the transit agency's size, ridership data collection may need to be collected on several routes every day, thus resulting in a substantial size database. Establishing a manual data collection system that requires continually coding the ridership data into a database is labor intensive and very expensive. For these reasons, transit agencies are beginning to implement automated data collection techniques. Possible benefits of an automated data collection system over a manual system include:

- Fewer chances of errors and less tedious (from coding and transferring)
- Data can be compiled and organized quicker
- Analysis of the data can be made more quickly, resulting in more timely service alteration decisions
- Large quantities of ridership data can be stored or retrieved instantly
- Less expensive in the long term

OBJECTIVES

In an effort to aid small transit agencies, the Texas State Department of Highways and Public Transportation (SDHPT) contracted the Texas Transportation Institute (TTI) to identify those transit agencies interested in an automated data collection system, test and recommend a system and then evaluate its effectiveness (Phase 1). In that study (1) an automated data collection system was identified that was the most versatile and cost effective for small transit agencies. The system, produced by Multisystems Inc. was field tested by TTI personnel on two of the bus routes of Citibus in Lubbock, Texas. The conclusions of phase 1 were that the automated system appeared to be a reasonable expenditure for small transit agencies when integrated into an ongoing performance

monitoring system. This study is a continuation of the previous study and has the following objectives:

- Develop a more detailed users manual for the automated system
(Supplemental User's Guide)
- Procure an IBM compatible PC and dot matrix printer
- Supply and actual transit agency (Citibus) with the PC, printer, Epson portable computer, automated data collection software and the more detailed users manual
- Train the transit personnel in the use of the automated system
- Evaluate the transit agencies use of the automated system
- Demonstrate the automated data collection system to small transit agencies

DESCRIPTION

The automated data collection system that was chosen to be evaluated in this study is one developed by Multisystems Inc. The system is actually a semi-automated system, in that it requires a checker to press certain keys on a portable computer as passengers board and alight at each bus stop. The system is comprised of two integrated software packages, Check*mate is the on-board data collection software, and Transit Information Manager (TIM) is the database, error checking and report generation software.

Check*mate is the name of the software package that is used for the actual recording of the passenger boardings and alightings or bus arrivals at each stop. The software is written in BASIC and is intended to run an Epson HX-40 portable computer (Figure 1). The Check*mate system consists of two program modules, LOADCHEC and RIDECHec.

LOADCHEC is used for collecting bus arrival times and passenger loads at selected stops. A checker loads a "buslist" into the Epson portable computer that contains the scheduled arrival times, route number and bus number for a certain stop. The checker then visits the stop and when buses arrive, he aligns the cursor on the screen with the route and

bus number corresponding to the arriving bus and presses the key marked "arrive" and then codes in the passenger load. The internal clock of the computer is used to record the arrival time.

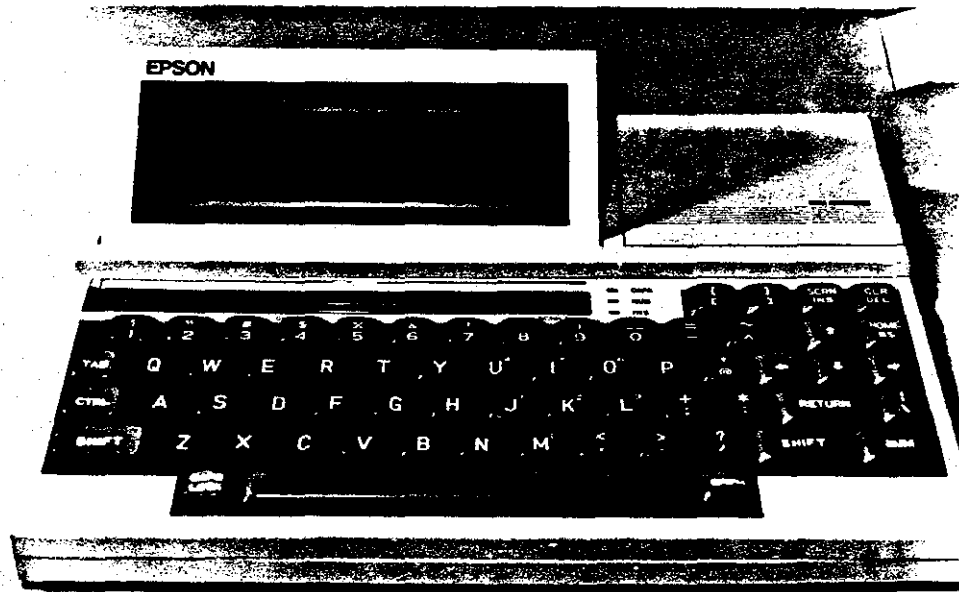


Figure 1. Epson HX-40 Portable Computer

RIDECHEC is used for collecting passenger boardings and alightings by stop and times at selected stops. The checker first creates a "stoplist" using a word processing program and then transfers it to the Epson portable computer. The stoplist contains the information that is necessary for the RIDECHEC program to be executed. Each individual stop name, in the sequence which they occur and the stops where time is to be recorded are all included in the stoplist. The checker boards the bus with the Epson portable computer that contains the RIDECHEC software and the stoplist for the route on which he is about to collect data. RIDECHEC will prompt the checker with each stop name and allow him to code in the number of passengers boarding and alighting as each stop is

encountered. The time is recorded for certain stops called "key" stops using the internal clock of the computer. The route data is stored in the internal RAM memory of the Epson until it can be transferred to a PC. Transferral of the stoplist to the Epson portable computer and ridership data transferral to a PC require two BASIC separate communications programs that are supplied with the system. The transfer process also requires a commercially available communications program such as PROCOMM to run on the PC.

Transit Information Manager (TIM) is the software package for processing the bus ridership data. Processing in this case, means to check, correct, store and summarize the data that has been collected. TIM was written for an IBM-PC using MS-DOS operating system. Minimum hardware requirements for the TIM software include: 192K RAM memory, monochrome or color 80 column display monitor, 5 mb of hard disk memory and an 80 column character printer. The main menu of TIM consists of four major subsystems: data management, data entry, analysis functions and report generation.

The data management subsystem allows the user to move verified data from "on-line" (data that is currently being used) to inactive storage and retrieve and verified data from inactive storage. It is also used to define the network (bus stops) and scheduling information for the agency. This section is the most laborious part of the entire automated system, however, once it is completed only minor updating is required due to service changes.

The data entry subsystem is used to input unverified data (data that has not been checked for errors), verify the data, modify the data if needed and load the verified data into the master data base for summarization and reporting.

The analysis functions provide a statistical means of processing the data. Presently included in this subsystem is the difference of means and sample size calculation.

The report generation subsystem is used to generate and print up to 17 predefined reports. These reports include basic, summary, profile, trend, and comparison reports.

FIELD TEST

Phase 1 of this study included the purchase of the Multisystems automated data collection system and then evaluation of the system by TTI personnel. The automated system consisted of an Epson portable computer, Check*mate software, TIM software and the accompanying literature. Citibus, a small transit system in Lubbock, Texas was contacted and allowed TTI personnel to test the automated system using their facilities. Route and scheduling information were obtained and a stoplist for RIDECHEC was prepared for two routes prior to data collection in Lubbock. The checker boarded the bus and collected ridership data for four trips for each route. The checker returned to the office, set up the schedule and network in TIM, and loaded the RIDECHEC data into TIM. Some errors in the data were detected (checker errors only), so they corrected and the data was verified. The "verified" data was then loaded into the master database and all of the applicable reports were then produced.

DOCUMENTATION

Field testing of the automated system proved to be difficult with the minimal amount of documentation that was provided with the system. One of the recommendations of Phase 1 of this study was to develop a more complete set of instructions, to guide the user (in a step by step fashion) through the set-up and execution of the automated system. A supplemental user's guide for the automated bus data collection system is provided in Appendix A.

IMPLEMENTATION

Observation and evaluation of the actual set-up and use of the automated data collection system would provide some insight into the usefulness and feasibility of such a

system. Citibus of Lubbock was again selected as the candidate for implementation (pilot test site). The purpose of the pilot testing was to evaluate the automated system in actual use by a transit agency.

An IBM compatible PC and dot matrix printer were purchased. The TIM software was loaded onto the PC and all of the automated systems functions were tested. The Supplemental Users Guide, the PC, the printer, the Epson Portable Computer, the RIDECHec software and the TIM software and documentation were then supplied to Citibus personnel. Instructions were given as to the set-up and execution of the various hardware and software components. One of Citibus' thirteen routes was selected and used as a step by step demonstration on how to create a stoplist for RIDECHec data collection, transfer it to the Epson, actually collect the data, transfer the data to the PC, define the network in TIM, load and verify the data, and then produce standardized reports. Creation of the remaining twelve stoplists, adding the appropriate route and schedule information to the TIM data base and then collecting and summarizing the data were left up to the Citibus personnel.

As Citibus continued to set up the remaining twelve routes into the network portion of TIM, some flaws in the TIM software were revealed. TIM contains a self diagnostic test sequence and when an entry is invalid or incorrect a "fatal error" code is reported. However, only UMTA has a listing of the nature of each error code. When a fatal error occurs, it is up to the individual user to identify the problem himself. Two fatal errors that occurred for Citibus were: (1) That a stoplist (RIDECHec) spelling of a bus stop did not match the spelling in TIM. TIM checks the order of the stops and the spelling of the stops for the various screens contained in TIM, but it does not cross check the spelling of stops from RIDECHec to TIM. (2) That the data base was full after only a few days worth of data. The version of TIM that TTI had purchased was not capable of expanding the data base to the size necessary for actual use. Both of these errors required assistance from Multisystems, but were corrected.

OBSERVATIONS

Citibus personnel have used the automated data collection system for several months. The following are observations that they had concerning the system:

RIDECHEC

It seems repetitive that the user must first create a stoplist using a word processing program and then type in a list of bus stops for the TIM network definition. A more integrated system could create a stoplist that could be loaded into the Epson portable computer from the network portion of TIM.

As data is being collected using the RIDECHEC module, when "depart" is pressed a display of the next bus stop would be extremely helpful. This would allow the checker to identify and anticipate the next stop, without having to press the "next" and "previous" keys.

These are the only aspects of the data collection portion of the system that the Citibus users felt needed improving. The Epson portable computer and RIDECHEC software has performed without incident since its delivery to Citibus.

TIM

Several observations, comments and recommendations were made concerning the TIM software package.

The user of the automated system at Citibus felt that this system really required (as a minimum) an employee to use the system daily in order to become familiar and efficient in its use. Data collection of UMTA's Section 15 reports require routes to be sampled every other day and the days when data is not collected could be used for data verification and reporting. He also felt that the Lubbock bus system (13 routes and approximately 25

peak hour buses) would require about a month (full time) to set up the automated data collection system and become operational.

Set-up of the TIM system (network and service definition) is the most time consuming portion of the automated system. The nine screens required to define the network and service in TIM could be reduced, since several of the screens only add one input beyond the previously entered screen.

The "Data Archival/Deletion" option in TIM does not archive the data (create a file) it simply deletes it.

When the user uses the "Network Archival" option, and then tries to retrieve that network, the period data is lost. The user must re-enter the periods, run time classes, service and trips, otherwise a fatal error will occur when entering RIDECHEC data.

A listing and definition of the fatal errors that occur in TIM would be very beneficial to the user in checking and debugging the network.

Overall the TIM software seems overly complex, and is not as user friendly as it could be. However, the system is effective and can be very beneficial especially for collecting ridership data for UMTA reports. The concept of an equipment pool to provide automated data capabilities to small transit agencies also did not seem feasible to the Citibus personnel, due mainly to the frequency that data must be collected for UMTA reports. However, they were interested in the concept of a state supported training program in the use and/or set-up of such an automated data collection system. Citibus is currently using the automated system as a special project in an attempt to modify the scheduled routes around the Texas Tech University area. Peak passenger loads in this area are dependent on the class schedules. Ridership data will help Citibus develop operational strategies.

DEVELOPMENTS

Included in the hardware procurement for Phase 2 of this study, an additional Epson portable computer was to be obtained. The additional Epson was needed to aid in the answering of questions and set-up procedures for Citibus from the TTI home location. The Epson HX-40 that was previously purchased had been discontinued. The CP/M operating system that was used on the HX-40 had become obsolete. As a replacement, a Toshiba T1000 was purchased that used a more common MS-DOS operating system. The Toshiba also had other advantages over the Epson:

- The internal ram of the Toshiba was 640K, capable of running most commercially available software
- An additional 640K of internal memory was configured as a battery backed hard disk
- The 3.25 inch flexible disk drive could be used to transfer the stoplist to the portable computer and transfer the RIDECHec data to the PC, eliminating the need for the BASIC communications programs
- A word processing program such as Wordperfect could be used on the portable to create the stoplist
- The Toshiba is only slightly larger than the Epson and battery life was nearly identical on the two machines.

The only drawback that the Toshiba had was that Check*mate software was written for a CP/M machine and required rewriting of the software for the new MS-DOS machine. At this time the RIDECHec software has been rewritten and compiled and is operational on the Toshiba portable computer. Further research and development on the MS-DOS version of RIDECHec is needed to include the additional enhancements identified in this study. A more simplified version of TIM has also been planned and is under development by TTI personnel. Further research is also needed to complete this task.

DEMONSTRATION

The capabilities and an overview of the implementation of the automated data collection system were demonstrated at the 17th Annual Texas Public Transportation Conference. A slide show was presented that described all of the inputs and outputs of both the Check*mate and TIM software packages and included the step by step procedures from stoplist preparation to report generation.

CONCLUSIONS AND RECOMMENDATIONS

The conclusion in phase 1 of this study was that a state supported equipment pool to provide automated data collection capabilities for small transit agencies did not appear feasible. The observations of and recommendations from Citibus personnel concur with this finding. Data must be collected too often for a system to continually "check out" equipment. One possibility is for the SDHPT to supply small transit agencies with the training and/or set-up of the network and require the agencies to purchase the hardware themselves. The implementation or pilot test, did indicate that small transit agencies would find such an automated data collection system beneficial, it also indicated the intensive effort needed for initial start-up. The reports produced by TIM can be used directly in UMTA reporting. Citibus has also shown that the automated system can be used to aid in scheduling. Citibus personnel have indicated that they will continue to use the automated system as long as possible.

The addition of more current hardware (Toshiba portable computer) and corresponding software development indicate that this automated data collection system can be improved. Additional development is needed on the RIDECHEC software to make it more efficient, as well as the development of a less complex, more user friendly data base (TIM type) package. Fewer input screens could be used in TIM and the development of a RIDECHEC stoplist taken directly from the network portion of TIM would produce a more integrated system.

APPENDIX A

AUTOMATED BUS DATA COLLECTION SYSTEM



A Supplemental User's Guide

September 1989

**Texas Transportation Institute
The Texas A&M University System
College Station, Texas 77843-3135**

INTRODUCTION

There are two software packages that make up the Multisystems automated bus data collection system. The Check*mate software is for "automating the collection of passenger boarding and alighting data by stop (RIDECHEC program) and bus passenger load and time data at selected points (LOADCHEC program)." The software was written to be run on the Epson HX-20 or HX-40 computer. The portable computer is used to run the Check*mate programs and collect the bus transit data while in the field. The Epson that was used was a portable laptop computer with a rechargeable battery and removable memory cartridges (HX-40) to store the collected data. Once the data has been collected, the checker returns to the office and the data is transferred to a PC. The Transit Information Manager (TIM) software is then used for checking, processing and reporting bus transit data. The (TIM) software runs on an IBM compatible personal computer (PC). Together the two software packages provide transit managers with a tool to make timely decisions based on operational data and also produces statistics which are required of them for governmental funding.

This manual is a supplement to the users' manuals provided by Multisystems Inc. for the use with its Check*mate and Transit Information Manager software packages and includes the author's experience with these packages. This manual was not intended to replace the manufacturer's documentation, only to clarify it and provide a better explanation of several key operations which can be confusing. The flow chart in Figure 1 illustrates the steps involved in the collection and reporting of bus transit data using the Multisystems Inc. automated data collection system. The four basic steps associated with the collection of transit data are discussed first with a high degree of detail. The last sections of this manual include the six basic steps of data management (Transit Information Manager). These sections are covered in Multisystems "Transit Information Manager Functional Specifications & Users Guide," so this manual only contains information that was omitted and instructions for specific applications. Sample inputs and outputs for the RIDECHEC and TIM systems are contained in Appendix A. The samples are listed in their proper order of execution.

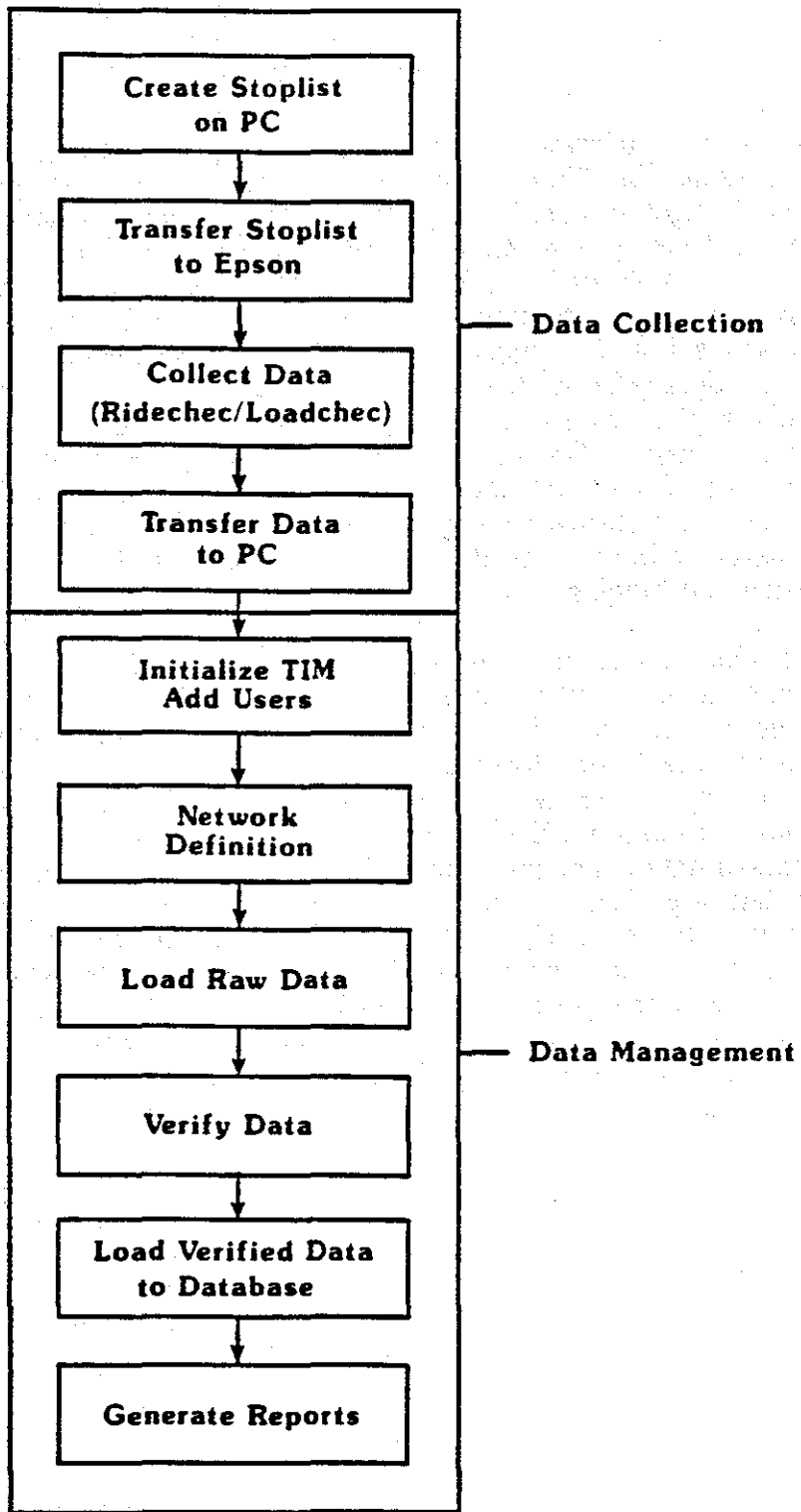


Figure 1. Steps Involved in the Multisystems Automated Data Collection System

CREATING A STOPLIST

The RIDECHEC program uses a stoplist file (a listing of all stops on a route in both directions) in the order in which they would be encountered as one travelled along the route. The simplest way to create a stoplist is using a word processing program on a PC. Left margin justification should be used and the stoplist should be saved in ASCII format (non-document, no carriage returns or control characters). If the stoplist is saved on the hard-disk of the PC a backup of the file is always recommended on a flexible disk. The stoplist contains three types of records: the route description, pattern description and stop description. The fields in each record must be in a fixed order and must be separated by commas. The following are the formats for each record.

Route Description Records: A,N,N
 N,N,N,N

Record	Field	Type	Description	Comments
1	1	A	route id	
	2	N	# patterns, direction 0	there may be several patterns on each route
	3	N	# patterns, direction 1	
2	1	N	# segments, direction 0	segments are portions of each pattern
	2	N	# segments, direction 1	
	3	N	# stops, direction 0	
	4	N	# stops, direction 1	

A = Alpha-numeric character
N = Numeric character only

The route id is an alpha-numeric field that often corresponds to a certain route number.

Only two directions are permitted. For convenience 0 is often used for outbound and 1 is used for inbound portions of a bus route.

Segments are bounded by key stops, so to define a segment a key stop must first be discussed. Key stops are points on a route where either a service pattern begins or ends, branches meet or split or a recording of time is required (the data from RIDECHEC only contains time recordings for key stops). A segment is defined by each key stop and each group of stops between key stops. For example, if a route has only one service pattern, and five stops and time is to be recorded only at the first and last stops, it would require three segments. The first stop would be segment 1, stops 2, 3 and 4 would be segment 2 and the last stop would be segment 3.

Pattern Description Records: N,N,N,N,N,N,N,N

Record	Field	Type	Description	Comments
1	1	N	pattern segment code 1	the pattern segment code fields are the decimal values of the bit-mapped flags
	2	N	pattern segment code 2	
	3	N	pattern segment code 3	
	4	N	pattern segment code 4	
	5	N	pattern segment code 5	
	6	N	first stop in pattern	
	7	N	last stop in pattern	

A pattern defines a unique set of stops over which a service is operated. This includes any branches or short turns.

The pattern segment codes (PSC) is a two-byte (16-bit) integer, of which the lowest 15 bits are used to indicate whether a segment is on a pattern for each direction. Table 1 shows the value of a PSC for 15 segments. In sample A all 15 segments were included in the pattern, thus the resulting PSC is the summation of the Bit Position values for all 15 segments (PSC = 32767). In the previous example with only three segments and one pattern, the PSC would be the summation of the Bit Positions for segments 1,2 and 3 (PSC = 7). There can be no more than 75 segments defined for each direction and segments that are not used should be coded as 0. Segments are numbered consecutively from 1 to 75 and if a segment (bounded by key stops) is on a pattern (a certain set of stops) then its value is flagged. The values are integers that double with each proceeding segment. The values of the segments that were flagged are summed and provide a numeric code that describes the service.

Segment (Bit Position)	Value of Bit Position	Sample Flag Settings			
1	1	1	1	0	1
2	2	1	1	0	0
3	4	1	1	0	1
4	8	1	1	0	0
5	16	1	1	0	1
6	32	1	1	0	0
7	64	1	1	0	1
8	128	1	1	0	0
9	256	1	0	1	1
10	512	1	0	1	0
11	1024	1	0	1	1
12	2048	1	0	1	0
13	4096	1	0	1	1
14	8192	1	0	1	0
15	16384	1	0	1	1
Resulting PSC's		32767	255	32512	21845

Stop Description Records: N,N,AAAAAAAA

Record	Field	Type	Description	Comments		
				Value	Dir	Key Stop
1	1	N	stop key	0	0	no
				1	0	yes
				2	1	no
				3	1	yes
	2	N	stop segment code			number of segment in which stop is included
	3	A	stop id			up to 8 character stop identifier

After the records are combined, a stoplist will have the following form:

```

A,N,N                    Route Description
N,N,N,N
N,N,N,N,N,N,N        Pattern Description (there may be more than one)
N,N,AAAAAAAA        Stop #1 Description
N,N,AAAAAAAA        Stop #2 Description
.                    .
.                    .
.                    .

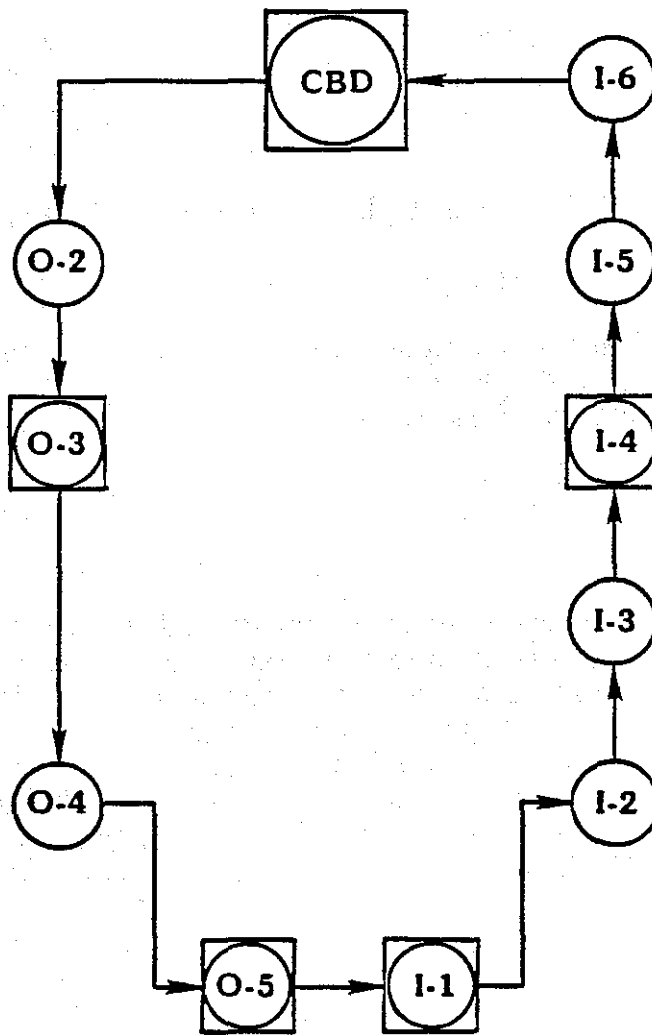
```

There are two ways to construct a bus route depending upon the needs of the provider. The route may have two distinct directions (outbound and inbound), in which case the RIDECHEC program terminates after the outbound direction. The checker must record the number of passengers on the bus when the outbound trip ends so that he may input the number of passengers already on board for the inbound trip. The other alternative is to construct the route to consist of only one direction (all stops are listed sequentially from the beginning to the end with no change in direction). The route with one direction may even be used for routes that start from one point and return to that same point.

Example 1

The following bi-directional bus route (Route #3) contains only one service pattern. The route originates from a stop in the central business district (CBD) and makes 4 additional outbound stops before it begins its inbound run back to the CBD. Of the 5 outbound stops (direction 0), the first, third and fifth are key stops. There are 7 inbound stops (direction 1), the first occurs following the fifth outbound stop and the seventh stop coincides with the route origin in the CBD. The key stops inbound are the first, fourth and seventh stops.

ROUTE 3



□ Denotes Key Stops

The bi-directional stoplist for route #3 would be as follows:

<u>Stoplist</u>	<u>Description</u>
3,1,1	Route #3, 1 pattern in 0 dir, 1 pattern in 1 dir
5,5,5,7	5 segments dir 0, 5 segments dir 1, 5 stops dir 0, 7 stops dir 1
31,0,0,0,0,1,5	PSC dir 0, no PSC for fields 2-5, first stop, last stop
31,0,0,0,0,1,7	PSC dir 1, no PSC for fields 2-5, first stop, last stop
1,1,CBDSTOP	Key stop 0 dir, segment 1, stop id
0,2,STOP2OUT	Non-key stop 0 dir, segment 2, stop id
1,3,STOP3OUT	Key stop 0 dir, segment 3, stop id
0,4,STOP4OUT	Non-key stop 0 dir, segment 4, stop id
1,5,STOP5OUT	Key stop 0 dir, segment 5, stop id
3,1,STOP1IN	Key stop 1 dir, segment 1, stop id
2,2,STOP2IN	Non-key stop 1 dir, segment 2, stop id
2,2,STOP3IN	Non-key stop 1 dir, segment 2, stop id
3,3,STOP4IN	Key stop 1 dir, segment 3, stop id
2,4,STOP5IN	Non-key stop 1 dir, segment 4, stop id
2,4,STOP6IN	Non-key stop 1 dir, segment 4, stop id
3,5,CBDSTOP	Key stop 1 dir, segment 5, stop id

If the same route were to be constructed as a single direction, direction 0 for example, the stoplist for route 3 would be as follows:

Stoplist
3,1,0
10,0,12,0
1023,0,0,0,0,1,12
1,1,CBDSTOP
0,2,STOP2OUT
1,3,STOP3OUT
0,4,STOP4OUT
1,5,STOP5OUT
1,6,STOP6OUT
0,7,STOP7OUT
0,7,STOP8OUT
1,8,STOP9OUT
0,9,STOP10OU
0,9,STOP11OU
1,10,CBDSTOP

Boardings, alightings and time will be recorded at the same stops as the bi-directional route, but the checker will not have to restart the RIDECHEC program when changing directions.

TRANSFERRING STOPLIST TO EPSON PORTABLE COMPUTER

The stoplist file then needs to be transferred to the Epson portable computer so that the RIDECHec program will have a list of all the stops along a route. The stoplist file can be transferred to the Epson using a communications program such as PROCOMM, PCPLUS or PC-TALK. If PROCOMM or PCPLUS is to be used to transfer the stoplist from the PC to the Epson, the following sequence of steps should be followed:

Turn on the Epson portable computer, position the cursor over the CONVERTA file name press <RETURN>.

Press <S> for stoplist.

Make sure the communications cable is connected to the RS-232 port on the Epson and to the serial port of the PC.

For stop file name enter <COM0:>, the Epson is now ready to receive the stoplist file from the host PC.

The stoplist should now be sent from the host PC using the communications program, type <PROCOMM> or <PCPLUS> and then <RETURN>.

From the main help menu, select the send files option.

From the upload (send) menu select the ASCII protocol.

Enter the file name, example: <A:STOPLIST1> and enter <RETURN>.

The file transfer will terminate at the Epson, at which point the user must press the space bar and type <SYSTEM> to return to the main menu.

RUNNING THE RIDECHec PROGRAM

Once a stoplist has been transferred to the Epson Portable Computer, the RIDECHec program may be invoked. To initiate the RIDECHec program and begin bus ridership data collection, turn the computer on and use the arrow keys to highlight the RIDECHec program area and press <RETURN>.

The stoplists that are available are listed on the screen, along with a question of which route number is to be used (by convention, the extension of the stoplist filename is the route number). Enter the route number and press <RETURN>. Range - number between 1 and 999.

Enter a Checker Identification number (unique to each person collecting ridership data) and press <RETURN>. Range - number between 1 and 999.

Enter a Direction Code and press < RETURN >. Remember, one stoplist may have 2 directions. (For convenience 0 is usually outbound and 1 is inbound.)

Enter a Trip Identifier and press < RETURN >. The Trip Identifier is an 8 digit entry that is unique for each direction of each route and is the first line in the output bus ridership data. Range - number between 10000001 and 99999999. If the RIDECHEC data is to be loaded into the TIM package, this identifier should contain an entry for the following: route, branch, period, direction and trip.

For example the following trip identifier could be defined as:

3101029

3	Route #3
1	Branch #1
0	Direction 0
1	Period #1
0029	Trip #0029

Enter a Pattern Code and press < RETURN >. As described previously, a pattern is a unique set of stops over which a service is operated. The screen will display the number of patterns that are available for the route and direction that have been chosen. Range - number specified in stoplist file.

Enter the number of passengers that were already on board when the data collector boards the bus (this may be prior to or at the first stop) and press < RETURN >.

The first stop will appear on the screen along with an indicator as to the number of passengers that board (get on), alight (get off) and the total number of passengers remaining on the bus. The checker then records the number of passengers alighting and boarding. Boarding passengers are entered by pressing the key labeled "on" (right arrow) and alighting passengers are entered by pressing the key labeled "off" (tab). Inadvertent inputs can be corrected using the "-off" (w) and "-on" (left arrow) keys. In addition to the keys used for counting passengers, 6 other keys are also used.

ARRIVE (PF1) -

This key is used when the bus stops to board or discharge passengers. Three actions occur when this key is pressed: the current time is stored, data for the previous stop is saved in memory, and the next stop is displayed on the screen.

DEPART (PF2) -

This key is used when the bus departs from the first stop. The depart time is stored. If it is pressed more than once for the same stop (i.e., when a bus starts to pull out and then stops again), the departing time is updated to the new time.

NEXT (PF3) -

This key is used to scroll through the stop list to display the appropriate stop label on the screen. It can be used at any time before the ARRIVE key is pressed at the next stop. The stop that is displayed on the screen when the ARRIVE key is pressed is saved in the record for the previous stop.

PREVIOUS (PF4) -

This key should be used to scroll backwards through the stoplist. It can be used only when the next key has been pressed too many times.

INSERT (PF5) -

This key is used when the bus stops at a point not included in the stop list. For example, when the bus is diverted from its regular route because a street is temporarily closed or a bus stop has been relocated but the stoplist was not updated. The stop that will be displayed on the screen will be that of "OFF & LINE".

END (SHIFT/PF5) -

This key is used at the end of a trip, either at the last stop in the pattern or when a trip is terminated abnormally (e.g., due to a vehicle breakdown). Use of this key requires pressing the SHIFT and END keys simultaneously.

As the bus starts to depart the stop, press the DEPART (PF2) key. The blinking cursor will move to the upper right hand corner of the screen while awaiting the arrival of the next stop. When the bus arrives at the next stop press the ARRIVE (PF1) key and that stop will be displayed. Passenger boardings and alightings may now be recorded for this stop.

At the last stop, passenger boarding and alighting will be recorded and the screen will display a message to "Press SHIFT and PF5 to end trip." After these keys have been entered, the checker may continue with the present stoplist, select a new stoplist or return to the menu. If returning to the menu, the space bar should be pressed and "SYSTEM" typed to exit BASIC. The original main menu will then be displayed.

CREATING A BUSLIST

The LOADCHEC program is used for collection vehicle times and passenger loads at specific locations. A buslist (a list of buses sorted in the order which they should pass the checkpoint) must be created on the PC and transferred to the Epson. The buslist, just as the stoplist, should be saved in ASCII format and contains only two types of records. The first record simply contains a location code for the checkpoint. The remainder of the buslist contains the scheduled arrival times, the route number and the direction code for each passing bus. The buslist may be created using a word processing program such as WordPerfect and when completed will have the following form:

N	Location code	
HH:MM,NNN,D		Scheduled arrival time, Route #, Direction Bus #1
HH:MM,NNN,D		Scheduled arrival time, Route #, Direction Bus #2
HH:MM,NNN,D		Scheduled arrival time, Route #, Direction Bus #3
.	(etc)	
.		
.		

TRANSFERRING THE BUSLIST TO THE EPSON PORTABLE COMPUTER

The buslist file needs to be transferred to the Epson computer so that the LOADCHEC program will have a list of buses that are scheduled to arrive at a checkpoint. The buslist file can be transferred from the PC using a communications program. If the PROCOMM communications program is used, follow the same steps as described previously for the transfer of a stoplist. However, when the CONVERTA program on the Epson prompts the user, the user must enter a for a buslist transfer.

RUNNING THE LOADCHEC PROGRAM

Once the buslist has been transferred to the Epson computer, the LOADCHEC program may be invoked. To start the LOADCHEC program and begin collection of bus arrival and passenger load data, turn the Epson on, use the arrow keys to position the cursor over the LOADCHEC program area and press <RETURN>.

The last buslist that has been loaded will appear on the screen.

For example:

```
06:00 007 0      Bus# PF1
06:05 007 1
06:10 007 1
```

The scheduled arrival times are displayed as well as the route number (007) and the direction code (0 or 1). The cursor will initially be just to the right of the direction code, and Bus# and PF1 will be displayed to the right. The checker uses the arrow keys to position the cursor on the line which matches an arriving trip. The PF1 key stores the time for the arriving trip and an asterisk(*) is displayed for that bus. The checker will enter the bus number (a 4 character field) and then press the right arrow key. The cursor will move to the right and the load or passenger count is entered (a 3 character field). When the checker has completed the list (or must quit) PF5 should be pressed to copy the data to RAM memory (drive I on the Epson).

TRANSFERRING RIDECHEC OR LOADCHEC DATA TO PC

The output from the RIDECHEC and LOADCHEC program needs to be transferred to a PC so the it can be summarized and reported using the Transit Information Manager (TIM) software. To transfer either RIDECHEC or LOADCHEC data from the Epson computer to a PC requires the following steps:

Use a file transfer program such a PROCOMM on the PC to download or receive the data file from the Epson.

Make sure the communications cable is connected to the RS-232 port on the Epson and to the serial port on the PC.

Turn on the Epson, position the cursor over the XFRCKDAT.BAS file and press <RETURN>.

Select the appropriate response for the data to be transferred ("R" for RIDECHEC, "L" for LOADCHEC).

The data transfer will begin and end automatically. To delete the data from memory press <D>. It is recommended that the data be deleted on the Epson computer after it is successfully transferred to the PC. If data is not deleted on the Epson, the next set of data that is collected will be added to the bottom of the previous data set. Pressing any other key will return the Epson computer to the menu, press the space bar and type <SYSTEM> to return to the main menu.

A sample of a RIDECHec data file is shown below.

```
010489 10 31010029 ALRDOND 0 0 0
010489 10 31010029 CBDSTOP 5 0 6:15
010489 10 31010029 STOP2OUT 4 0 0
010489 10 31010029 STOP3OUT 2 0 6:16
010489 10 31010029 STOP4OUT 0 2 0
010489 10 31010029 STOP5OUT 2 3 6:20
010489 10 31010029 STOP6OUT 1 0 6:22
010489 10 31010029 STOP7OUT 4 1 0
010489 10 31010029 STOP8OUT 6 0 0
010489 10 31010029 STOP9OUT 0 7 6:25
010489 10 31010029 STOP10OU 0 2 0
010489 10 31010029 STOP11OU 0 3 0
010489 10 31010029 STOP12OU 0 0 6:30
010489 10 31010029 REMONBRD 0 0 0
```

A print out of the RIDECHec data should be made for later use. To do so simply use the DOS "print" command (example: print ridedat.14)

TRANSIT INFORMATION MANAGER

Transit Information Manager (TIM) is a software package developed by Multisystems Inc. for the processing of bus and bus ridership data. The menu driven program consists of four modules: data entry, analysis functions, report generation and data management.

The data entry module is used to: input unverified data (data that has not been checked for errors) either interactively or by a data transferral, verify the data and check for consistency, modify the data if needed and load the verified data into the master database. The data entry module will handle four types of inputs: load counts, boarding counts, ridechecks and fare box readings.

The analysis module provides a statistical means of processing the data.

The report generation module is used to generate and print up to 17 predefined reports.

The data management module allows the user to move verified data from on-line to inactive storage and retrieve verified data from inactive storage. It is also used to delete records, define parameters of the service network and schedules, and define the formats of the transferred bus data files.

Once the checker returns from the field and transfers the RIDECHec or LOADCHec data to the PC, the TIM software is used to check, summarize, store and report the data. To use the TIM software the system must first be initialized, to do this type <MDBSINIT>. A batch file will initialize the system automatically.

INITIALIZE TIM/ADD USERS

Prior to entering TIM, the user must set up a user id and password, this is accomplished by using the DMU utility contained in the TIM package. The user should type <DMU>, at the data base prompt <TIM> should be entered, at the user name prompt <TIM> should be entered and finally at the password prompt <TIM> should again be entered. The user should select the Add, change, or delete users option by typing <A>. Typing <A> again will select the add user option. Enter a new user name and new password when prompted. The user will then be asked to enter a read and write access code. These should be left blank. When finished, exit to the DMU menu then exit DMU. Additional information on the DMU utility can be found in Appendix A of the Transit Information Manager (TIM) Functional Specifications and User's Guide.

The next step to log onto TIM is to type <TIM> and press <RETURN>. The current drive on the PC must be set to the hard disk upon which the TIM program and database are resident. Screen 0.0 will appear and the user will be asked to provide a user's identification, a password and the current date. The ID and the password must be entered exactly as they have been defined in DMU, upper and lower case letters are not considered to be the same by TIM. After the date has been entered press <RETURN>, the cursor returns to the user identification line and the user should press the F10 key to accept the screen.

NETWORK DEFINITION

The user must define the bus network and schedule. To do so, select the data management option from the main menu and then the network/schedule definition option on the next screen. All of the characteristics of the network and service are defined starting with "1. Define Periods" and ending at "9. Define Trips." Follow the directions in the TIM User's Guide for completing the network/schedule definition and note the following:

When defining the periods, use a numeric character for the periods (the same one used in the trip identifier of RIDECHEC).

Route, branch, direction and trip code must correspond to those used in the trip identifier of RIDECHEC.

When defining the stops, if the street or cross street name is less than 7 characters, use leading blank spaces. The entry will accept 12 characters but only the first 7 are used in the reports and are right justified.

Segments are portions of a route that extend from one time point to the next.

When defining the trip code, each direction of a single run must have a different trip code.

Trip boardings must equal trip alightings, all passengers must depart at the last stop.

When defining segments, the user should print out a copy of each segment (F7), and compare the print out of the RIDECHec data with the segments. Check to make sure that the stops are spelled correctly and especially that they are in the correct order. Always make a copy of the TIM.DB file after a network has been completed. Using the Archive Network option would do the same thing (make a backup of the coded network) however, when a network is archived the period data is lost. The user must then re-enter the period, run time classes, service and trips, otherwise a fatal error will occur. This is one of the problem areas in TIM that needs to be attended to.

LOAD RAW DATA

To load a raw data file (RIDECHec data for example) an input format must be constructed. In the data maintenance subsystem, option 6 defines the input file format. If a RIDECHec data file is used as input, a date and format identifier must be added to the file so that TIM will recognize it. Above the first line of data, the date and format id must be entered in the form: MM/DD/YYYY A where "A" is an alphanumeric character that must agree with the format id input into the RIDECHec format input screen. The first entry from a RIDECHec data file is the date that the data was collected, but this entry will not be used in TIM, only the date that is added at the top of the file is used. The trip identifier should have been constructed so as to contain a code for each of the following: route, branch, direction, period and trip.

VERIFY DATA

After data has been loaded, it is to be verified and checked for consistency with the network and schedule that has been defined. The printer should be turned on, loaded with paper and connected to the PC at this time, since any discrepancies in the data will be printed automatically on an Unverified Data Error Report. If any errors are found (usually errors are in the network/schedule definition), they should be corrected immediately. Check the data that has been loaded by scrolling through the data while in the modify input data section. If there is only a partial data set or if there are too many errors to correct, the data can be deleted using the modify input data option of the data entry subsystem. If the data has been totally deleted, it should then be reloaded and reverified after corrections. After data for a trip has been verified and no errors are found the network should be archived, this simply saves a "good" network and schedule for future use. The verification process does not check the correctness of the order of stops. If a fatal error number of 141.1.255 is displayed when loading a verified data set to the master data base, this means that one of the stops is misspelled or that one or more of the stops is out of order.

LOAD VERIFIED DATA TO MASTER DATABASE

After the data has been verified it should be loaded to the master database. Data must be loaded to the master database before reports can be produced. The master database should be backed up periodically to prevent the possible loss of the systems network and data. Make a copy of the TIM.DB file onto a flexible disk.

GENERATE REPORTS

Five different reports may be generated from the data that is in the master database. Note that the route segment that is requested is only one of the segments that make up a route and if no run number was input in the network/schedule definition then it should be left blank on the report form. Some of the reports require an input of a range of days to report, others only require a single date.

SAMPLE RIDECHEC FILES AND TIM SCREENS

THE UNIVERSITY OF CHICAGO

PH.D. THESIS

BY

DR. [Name]

IN

THE DEPARTMENT OF [Department]

CHICAGO, ILLINOIS

19[Year]

PH.D. THESIS

BY

DR. [Name]

IN

THE DEPARTMENT OF [Department]

CHICAGO, ILLINOIS

19[Year]

PH.D. THESIS

BY

DR. [Name]

IN

THE DEPARTMENT OF [Department]

CHICAGO, ILLINOIS

19[Year]

PH.D. THESIS

BY

DR. [Name]

IN

THE DEPARTMENT OF [Department]

CHICAGO, ILLINOIS

19[Year]

1,1,0
13,0,76,0
8191,0,0,0,0,0,1,76

Sample stoplist for route #1 with only one (0) direction
13 segments and 76 stops

1,1,BROAAVEH
0,2,BROAAVEG
0,2,BROAAVEF
0,2,BROAAVEE
0,2,AVEE13TH
0,2,AVEE14TH
0,2,AVEE15TH
0,2,AVEE16TH
0,2,AVEE17TH
0,2,17THAVED
1,3,17THAVEC
0,4,17THAVEB
0,4,17THAVEA
0,4,AVEA18TH
0,4,19THAVEA
0,4,19THBIRC
0,4,BIRC20TH
0,4,CEDA20TH
0,4,CEDA21ST
0,4,CEDA22ND
0,4,CEDA23RD
0,4,CEDA24TH
0,4,CEDA26TH
0,4,CEDACORO
0,4,COROIVOR
0,4,IVOR FIR
0,4,FIR HICK
0,4,FIR GLOB
1,5,FIR 26TH
0,6,26THHICK
0,6,24TH OAK
0,6,OAK 25TH
0,6,25THWEBE
0,6,WEBE OAK
0,6,WEBE27TH
0,6,WEBE28TH
0,6,WEBE29TH
0,6,29THQUIR
0,6,QUIR30TH
1,7,30THTEAK
0,8,TEAK29TH
0,8,TEAK28TH
0,8,TEAK27TH
0,8,TEAKREDB
0,8,26THQUIR
0,8,QUIR25TH
0,8,QUIR24TH
0,8,24TH OAK
0,8,24THHICK
1,9,26THGLOB
0,10,26TH FIR
0,10,26TH ELM
0,10,26THDATE
0,10,26THCEDA
0,10,CEDA24TH
0,10,CEDA23RD
0,10,CEDA22ND

0,10,CEDA21ST
0,10,CEDA20TH
0,10,BIRC20TH
0,10,BIRC19TH
0,10,19THAVEA
0,10,AVEA18TH
0,10,17THAVEA
0,10,17THAVEB
1,11,17THAVEC
0,12,17THAVED
0,12,17THAVEE
0,12,AVEE16TH
0,12,AVEE15TH
0,12,AVEE14TH
0,12,AVEE13TH
0,12,AVEEBROA
0,12,BROAAVEF
0,12,BROAAVEG
1,13,BROAAVEH

011789	12	12345678	BROAAVEH	6	0	1559	← Delete using Wordperfect
011789	12	12345678	BROAAVEG	4	0	0	
011789	12	12345678	BROAAVEF	2	0	0	
011789	12	12345678	BROAAVEE	2	1	0	
011789	12	12345678	AVEE13TH	4	3	0	
011789	12	12345678	AVEE14TH	0	0	0	
011789	12	12345678	AVEE15TH	1	2	0	
011789	12	12345678	AVEE16TH	4	0	0	
011789	12	12345678	AVEE17TH	2	0	0	
011789	12	12345678	17THAVED	0	2	0	
011789	12	12345678	17THAVEC	0	3	1604	
011789	12	12345678	17THAVEB	7	0	0	
011789	12	12345678	17THAVEA	1	4	0	
011789	12	12345678	AVEA18TH	3	5	0	
011789	12	12345678	19THAVEA	2	0	0	
011789	12	12345678	19THBIRC	1	1	0	
011789	12	12345678	BIRC20TH	0	0	0	
011789	12	12345678	CEDA20TH	0	5	0	
011789	12	12345678	CEDA21ST	3	1	0	
011789	12	12345678	CEDA22ND	3	4	0	
011789	12	12345678	CEDA23RD	5	1	0	
011789	12	12345678	CEDA24TH	0	3	0	
011789	12	12345678	CEDA26TH	0	2	0	
011789	12	12345678	CEDACORO	1	0	0	
011789	12	12345678	COROIVOR	3	1	0	
011789	12	12345678	IVOR FIR	3	2	0	
011789	12	12345678	FIR HICK	0	6	0	
011789	12	12345678	FIR GLOB	1	0	0	
011789	12	12345678	FIR 26TH	0	0	1610	
011789	12	12345678	26THHICK	0	0	0	
011789	12	12345678	24TH OAK	2	0	0	
011789	12	12345678	OAK 25TH	0	4	0	
011789	12	12345678	25THWEBE	2	1	0	
011789	12	12345678	WEBE OAK	4	1	0	
011789	12	12345678	WEBE27TH	3	1	0	
011789	12	12345678	WEBE28TH	0	5	0	
011789	12	12345678	WEBE29TH	0	0	0	
011789	12	12345678	29THQUIR	0	1	0	
011789	12	12345678	QUIR30TH	1	2	0	
011789	12	12345678	30THTEAK	3	0	1614	
011789	12	12345678	TEAK29TH	3	2	0	
011789	12	12345678	TEAK28TH	1	1	0	
011789	12	12345678	TEAK27TH	3	1	0	
011789	12	12345678	TEAKREDB	1	2	0	
011789	12	12345678	26THQUIR	5	3	0	
011789	12	12345678	QUIR25TH	0	5	0	
011789	12	12345678	QUIR24TH	1	0	0	
011789	12	12345678	24TH OAK	2	1	0	
011789	12	12345678	24THHICK	1	3	0	
011789	12	12345678	26THGLOB	2	1	1620	
011789	12	12345678	26TH FIR	0	5	0	
011789	12	12345678	26TH ELM	2	0	0	
011789	12	12345678	26THDATE	1	1	0	
011789	12	12345678	26THCEDA	1	0	0	
011789	12	12345678	CEDA24TH	2	0	0	
011789	12	12345678	CEDA23RD	2	3	0	
011789	12	12345678	CEDA22ND	2	1	0	
011789	12	12345678	CEDA21ST	6	0	0	
011789	12	12345678	CEDA20TH	0	2	0	

Sample RIDECHEC output

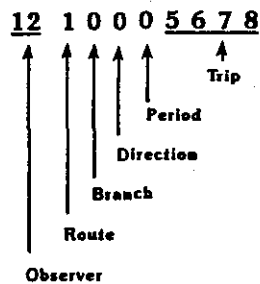
011789	12	12345678	20THBIRC	0	3	0
011789	12	12345678	BIRC19TH	0	2	0
011789	12	12345678	19THAVEA	0	1	0
011789	12	12345678	AVEA18TH	0	2	0
011789	12	12345678	17THAVEA	0	0	0
011789	12	12345678	17THAVEB	0	1	0
011789	12	12345678	17THAVEC	2	0	1625
011789	12	12345678	17THAVED	0	1	0
011789	12	12345678	17THAVEE	0	2	0
011789	12	12345678	AVEE16TH	0	3	0
011789	12	12345678	AVEE15TH	0	0	0
011789	12	12345678	AVEE14TH	0	1	0
011789	12	12345678	AVEE13TH	0	0	0
011789	12	12345678	AVEEBROA	2	0	0
011789	12	12345678	BROAAVEF	0	1	0
011789	12	12345678	BROAAVEG	2	0	0
011789	12	12345678	BROAAVEH	0	5	1630
011789	12	12345678	REMOVED	0	0	0

← Delete using Wordperfect

01/17/1989 A ← Add date and format using Wordperfect

011789	12	10005678	BROAAVEH	6	0	1559
011789	12	10005678	BROAAVEG	4	0	0
011789	12	10005678	BROAAVEF	2	0	0
011789	12	10005678	BROAAVEE	2	1	0
011789	12	10005678	AVEE13TH	4	3	0
011789	12	10005678	AVEE14TH	0	0	0
011789	12	10005678	AVEE15TH	1	2	0
011789	12	10005678	AVEE16TH	4	0	0
011789	12	10005678	AVEE17TH	2	0	0
011789	12	10005678	17THAVED	0	2	0
011789	12	10005678	17THAVEC	0	3	1604
011789	12	10005678	17THAVEB	7	0	0
011789	12	10005678	17THAVEA	1	4	0
011789	12	10005678	AVEA18TH	3	5	0
011789	12	10005678	19THAVEA	2	0	0
011789	12	10005678	19THBIRC	1	1	0
011789	12	10005678	BIRC20TH	0	0	0
011789	12	10005678	CEDA20TH	0	5	0
011789	12	10005678	CEDA21ST	3	1	0
011789	12	10005678	CEDA22ND	3	4	0
011789	12	10005678	CEDA23RD	5	1	0
011789	12	10005678	CEDA24TH	0	3	0
011789	12	10005678	CEDA26TH	0	2	0
011789	12	10000678	CEDACORO	1	0	0
011789	12	10005678	COROIVOR	3	1	0
011789	12	10005678	IVOR FIR	3	2	0
011789	12	10005678	FIR HICK	0	6	0
011789	12	10005678	FIR GLOB	1	0	0
011789	12	10005678	FIR 26TH	0	0	1610
011789	12	10005678	26THHICK	0	0	0
011789	12	10005678	24TH OAK	2	0	0
011789	12	10005678	OAK 25TH	0	4	0
011789	12	10005678	25THWEBE	2	1	0
011789	12	10005678	WEBE OAK	4	1	0
011789	12	10005678	WEBE27TH	3	1	0
011789	12	10005678	WEBE28TH	0	5	0
011789	12	10005678	WEBE29TH	0	0	0
011789	12	10005678	29THQUIR	0	1	0
011789	12	10005678	QUIR30TH	1	2	0
011789	12	10005678	30THTEAK	3	0	1614
011789	12	10005678	TEAK29TH	3	2	0
011789	12	10005678	TEAK28TH	1	1	0
011789	12	10005678	TEAK27TH	3	1	0
011789	12	10005678	TEAKREDB	1	2	0
011789	12	10005678	26THQUIR	5	3	0
011789	12	10005678	QUIR25TH	0	5	0
011789	12	10005678	QUIR24TH	1	0	0
011789	12	10005678	24TH OAK	2	1	0
011789	12	10005678	24THHICK	1	3	0
011789	12	10005678	26THGLOB	2	1	1620
011789	12	10005678	26TH FIR	0	5	0
011789	12	10005678	26TH ELM	2	0	0
011789	12	10005678	26THDATE	1	1	0
011789	12	10005678	26THCEDA	1	0	0
011789	12	10005678	CEDA24TH	2	0	0
011789	12	10005678	CEDA23RD	2	3	0
011789	12	10005678	CEDA22ND	2	1	0
011789	12	10005678	CEDA21ST	6	0	0
011789	12	10005678	CEDA20TH	0	2	0

Sample RIDECHEC output with
modifications necessary
for use in TIM



011789	12	10005678	BIRC20TH	0	3	0
011789	12	10005678	BIRC19TH	0	2	0
011789	12	10005678	19THAVEA	0	1	0
011789	12	10005678	AVEA18TH	0	2	0
011789	12	10005678	17THAVEA	0	0	0
011789	12	10005678	17THAVEB	0	1	0
011789	12	10005678	17THAVEC	2	0	1625
011789	12	10005678	17THAVED	0	1	0
011789	12	10005678	17THAVEE	0	2	0
011789	12	10005678	AVEE16TH	0	3	0
011789	12	10005678	AVEE15TH	0	0	0
011789	12	10005678	AVEE14TH	0	1	0
011789	12	10005678	AVEE13TH	0	0	0
011789	12	10005678	AVEEBROA	2	0	0
011789	12	10005678	BROAAVEF	0	1	0
011789	12	10005678	BROAAVEG	2	0	0
011789	12	10005678	BROAAVEH	0	5	1631

SCREEN 0.0

Sample TIM Input screens

Version 1.2

TIM
TRANSIT INFORMATION MANAGER

USER IDENTIFICATION: KIRK

USER PASSWORD: KIRK

TODAY'S DATE: 1/17/89

SCREEN 0.1

VERSION 1.2

TIM
TRANSIT INFORMATION MANAGER

MASTER MENU

1. DATA ENTRY
2. REPORT GENERATOR
3. ANALYSIS FUNCTIONS
4. DATA MANAGEMENT

ENTER DESIRED FUNCTION:

SCREEN 4

Version 1.2

TIM
DATA MAINTENANCE SUBSYSTEM

MENU

1. DATA ARCHIVAL/DELETION
2. DATA RETRIEVAL
3. NETWORK/SCHEDULE DEFINITION
4. NETWORK ARCHIVAL
5. NETWORK RETRIEVAL
6. INPUT FILE FORMAT DEFINITION

ENTER DESIRED FUNCTION:

SCREEN 4.3

Version 1.2

TIM
DATA MAINTENANCE SUBSYSTEM

NETWORK DEFINITION MENU

1. DEFINE PERIODS
2. DEFINE STOPS
3. DEFINE SEGMENTS
4. DEFINE ROUTES
5. DEFINE BRANCHS
6. DEFINE DIRECTIONS
7. DEFINE RUN TIME CLASSES
8. DEFINE SERVICE
9. DEFINE TRIPS

ENTER DESIRED FUNCTION:

SCREEN 4.3.1

VERSION 1.2

TIM
DATA MAINTENANCE SUBSYSTEM
PERIOD DEFINITION INPUT SCREEN

PERIOD CODE: 0
DAYS COVERED: WKD
PERIOD STARTS: 06:00
PERIOD ENDS: 18:00
DESCRIPTION: TEST

SCREEN 4.3.2

VERSION 1.2

TIM
DATA MAINTENANCE SUBSYSTEM
STOP DEFINITION INPUT SCREEN

STOP CODE: BROAAVEH
STREET STOP IS ON: BROADWAY
CROSS STREET AT STOP: AVE H
ANALYSIS ZONE:

SCREEN 4.3.3

TIM
DATA MAINTENANCE SUBSYSTEM
SEGMENT DEFINITION INPUT SCREEN

VERSION 1.2

SEGMENT CODE: 0101 SEGMENT NAME: OUT RT1 SEG1 TOTAL LENGTH: 0.91
ORIENTATION: 0 FROM TIMEPOINT: BROAAVEH TO TIMEPOINT: 17THAVEC

STOPS: BROAAVEG BROAAVEF BROAAVEE AVEE13TH AVEE14TH AVEE15TH AVEE16TH AVEE17TH
DIST: 0.090 0.090 0.090 0.090 0.090 0.090 0.090 0.10

STOPS: 17THAVED
DIST: 0.090

STOPS:
DIST:

STOPS:
DIST:

STOPS:
DIST:

SCREEN 4.3.4

TIM
DATA MAINTENANCE SUBSYSTEM
ROUTE DEFINITION INPUT SCREEN

Version 1.2

ROUTE CODE: 1
ROUTE NAME: DUNBAR EAST
ROUTE TYPE: R

SCREEN 4.3.7

VERSION 1.2

TIM
DATA MAINTENANCE SUBSYSTEM
RUN TIME CLASS INPUT SCREEN

EFFECTIVE DATE: 1/17/89

ROUTE CODE: 1

BRANCH CODE: 0

DIRECTION CODE: 0

PERIOD START: 6:00

DAYS COVERED: WKD

SEGMENT:	1	2	3	4	5	6	7	8	9	10
RUN TIME:	5	5	5	5	5	5				
SEGMENT:	11	12	13	14	15	16	17	18	16	20
RUN TIME:										

SCREEN 4.3.8

VERSION 1.2

TIM
DATA MAINTENANCE SUBSYSTEM
SERVICE DEFINITION INPUT SCREEN

EFFECTIVE DATE: 1/17/89

ROUTE CODE: 1

BRANCH CODE: 0

DIRECTION CODE: 0

PERIOD CODE: 0

BASE OPERATING COST PER TRIP: 15.00

SCREEN 4.3.9

VERSION 1.2

TIM
DATA MAINTENANCE SUBSYSTEM
TRIP DEFINITION INPUT SCREEN

EFFECTIVE DATE: 1/17/89

ROUTE CODE: 1

BRANCH CODE: 0

DIRECTION CODE: 0

PERIOD CODE: 0

TRIP CODE: 5678

DEPART TIME: 16:00

RUN CODE:

BLOCK CODE:

SCREEN 4.4

VERSION 1.2

TIM
DATA MAINTENANCE SUBSYSTEM
NETWORK ARCHIVE SPECIFICATIONS

NETWORK ARCHIVE FILE NAME: ROUTE1.NET

SCREEN 4.6

Version 1.2

TIM
DATA MAINTENANCE SUBSYSTEM
INPUT FILE FORMAT DEFINITION MENU

1. RIDE CHECK FORMAT
2. POINT CHECK FORMAT
3. FARE CHECK FORMAT

ENTER DESIRED FUNCTION:

SCREEN 4.6.1

TIM
RIDE CHECK FORMAT INPUT

Version 1.2

FORMAT ID: A

DATA ELEMENT	# OF OCCURENCE	SEQUENCE #	START COLUMN	FIELD WIDTH
OBSERVER		1	10	2
ROUTE		2	13	1
BRANCH		3	14	1
DIRECTION		4	15	1
PERIOD		5	16	1
TRIP		6	17	4
STOP		7	22	8
RUN NUMBER				
START TIME				
WEATHER				
ARRIVE TIME		10	37	4
DEPART TIME				
ON BOARD				
BOARDINGS	1	8	31	2
ALIGHTINGS	1	9	34	2
BUS CAPACITY				

SCREEN 1

Version 1.2

TIM
DATA ENTRY SUBSYSTEM
MENU

1. INPUT UNVERIFIED DATA
2. DATA VERIFICATION AND CONSISTENCY CHECK
3. MODIFY INPUT DATA
4. LOAD VERIFIED DATA TO MASTER DATABASE

ENTER DESIRED FUNCTION:

SCREEN 1.1

Version 1.2

TIM
DATA ENTRY SUBSYSTEM
INPUT MENU

1. LOAD FROM FILE: RIDEDAT.1
2. MANUAL RIDE CHECK ENTRY
3. MANUAL POINT CHECK ENTRY
4. MANUAL FAREBOX ENTRY

DATE OF DATA TO BE INPUT: 1/17/89

ENTER DESIRED FUNCTION: 1

SCREEN 1.2

Version 1.2

TIM
DATA ENTRY SUBSYSTEM
DATA VERIFICATION INPUT

DATE OF DATA TO BE VERIFIED: 1/17/89

SCREEN 1.4

Version 1.2

TIM
DATA ENTRY SUBSYSTEM
DATA LOADING INPUT

DATE OF DATA TO BE LOADED: 1/17/89

SCREEN 2

Version 1.2

TIM
REPORT GENERATOR SUBSYSTEM
MENU

1. BASIC REPORTS
2. SUMMARY REPORTS
3. PROFILE REPORTS
4. TREND REPORTS
5. COMPARISON REPORTS

ENTER DESIRED FUNCTION:

SCREEN 2.1

VERSION 1.2

TIM
REPORT GENERATOR SUBSYSTEM
BASIC REPORTS MENU

1. BASIC RIDE CHECK REPORT
2. BASIC POINT CHECK REPORT
3. BASIC FARE CHECK REPORT

ENTER DESIRED FUNCTION:

SCREEN 2.1.1

VERSION 1.2

TIM
REPORT GENERATOR SUBSYSTEM
BASIC RIDE CHECK REPORT

ROUTE: 1

DIRECTION: 0

DATE: 1/17/89

SCHEDULED START TIME: 16:00

ROUTE SEGMENT: 0101

RUN NUMBER:

Sample TIM reports

RPT 2.1.1 01/17/1989

TEXAS A&M UNIVERSITY

VERSION 1.2 PAGE 1

ROUTE: 1 DUNBAR EAST

SEGMENT: 0101 OUT RT1 SEG1

DIRECTION: 0

RUN #:

* BASIC RIDE CHECK REPORT *

START TIME: 16:00

DATE: TU 01/17/1989

		-----BOARDINGS BY FARE CATEGORY-----									
LOCATION	TIME	BOARD INGS	ALIGHT INGS	ON BOARD	FARE CAT1	FARE CAT2	FARE CAT3	FARE CAT4	FARE CAT5	FARE CAT6	FARE CAT7
BROADWA &	AVE H	6	0	6	6	0	0	0	0	0	0
BROADWA &	AVE G	4	0	10	4	0	0	0	0	0	0
BROADWA &	AVE F	2	0	12	2	0	0	0	0	0	0
BROADWA &	AVE E	2	1	13	2	0	0	0	0	0	0
AVE E &	13TH	4	3	14	4	0	0	0	0	0	0
AVE E &	14TH	0	0	14	0	0	0	0	0	0	0
AVE E &	15TH	1	2	13	1	0	0	0	0	0	0
AVE E &	16TH	4	0	17	4	0	0	0	0	0	0
AVE E &	17TH	2	0	19	2	0	0	0	0	0	0
17TH &	AVE D	0	2	17	0	0	0	0	0	0	0
17TH &	AVE C	0	3	14	0	0	0	0	0	0	0
TOTAL		25	11		25	0	0	0	0	0	0

RPT 2.1.1 01/17/1989

TEXAS A&M UNIVERSITY

VERSION 1.2 PAGE 1

ROUTE: 1 DUNBAR EAST

SEGMENT: 0102 OUT RT1 SEG2

DIRECTION: 0

RUN #:

START TIME: 16:00

DATE: TU 01/17/1989

 * BASIC RIDE CHECK REPORT *

LOCATION	TIME	BOARD INGS	ALIGHT INGS	ON BOARD	BOARDINGS BY FARE CATEGORY						
					FARE CAT1	FARE CAT2	FARE CAT3	FARE CAT4	FARE CAT5	FARE CAT6	FARE CAT7
17TH & AVE B		7	0	21	7	0	0	0	0	0	0
17TH & AVE A		1	4	18	1	0	0	0	0	0	0
AVE A & 18TH		3	5	16	3	0	0	0	0	0	0
19TH & AVE A		2	0	18	2	0	0	0	0	0	0
19TH9TH & BIRCH A		1	1	18	1	0	0	0	0	0	0
BIRCH & 20TH		0	0	18	0	0	0	0	0	0	0
CEDAR & 20TH		0	5	13	0	0	0	0	0	0	0
CEDAR & 21ST		3	1	15	3	0	0	0	0	0	0
CEDAR & 22ND		3	4	14	3	0	0	0	0	0	0
CEDAR & 23RD		5	1	18	5	0	0	0	0	0	0
CEDAR & 24TH		0	3	15	0	0	0	0	0	0	0
CEDAR & 26TH		0	2	13	0	0	0	0	0	0	0
CEDARAR & CORONAD		1	0	14	1	0	0	0	0	0	0
CORONAD & IVORYAD		3	1	16	3	0	0	0	0	0	0
IVORY & FIR		3	2	17	3	0	0	0	0	0	0
FIR & HICKORY		0	6	11	0	0	0	0	0	0	0
FIR & GLOBE		1	0	12	1	0	0	0	0	0	0
FIR & 26TH		0	0	12	0	0	0	0	0	0	0
TOTAL		33	35		33	0	0	0	0	0	0

RPT 2.1.1 01/17/1989

TEXAS A&M UNIVERSITY

VERSION 1.2 PAGE 1

ROUTE: 1 DUNBAR EAST

SEGMENT: 0103 OUT RT1 SEG3

DIRECTION: 0

RUN #:

START TIME: 16:00

DATE: TU 01/17/1989

* BASIC RIDE CHECK REPORT *

LOCATION	TIME	-----BOARDINGS BY FARE CATEGORY-----									
		BOARD INGS	ALIGHT INGS	ON BOARD	FARE CAT1	FARE CAT2	FARE CAT3	FARE CAT4	FARE CAT5	FARE CAT6	FARE CAT7
26TH & HICKORY		0	0	12	0	0	0	0	0	0	0
24TH & OAK		2	0	14	2	0	0	0	0	0	0
OAK & 25TH		0	4	10	0	0	0	0	0	0	0
25TH & WEBER		2	1	11	2	0	0	0	0	0	0
WEBER & OAK		4	1	14	4	0	0	0	0	0	0
WEBER & 27TH		3	1	16	3	0	0	0	0	0	0
WEBER & 28TH		0	5	11	0	0	0	0	0	0	0
WEBER & 29TH		0	0	11	0	0	0	0	0	0	0
29TH & QUIRT		0	1	10	0	0	0	0	0	0	0
QUIRT & 30TH		1	2	9	1	0	0	0	0	0	0
30TH & TEAK		3	0	12	3	0	0	0	0	0	0
TOTAL		15	15		15	0	0	0	0	0	0

RPT 2.1.1 01/17/1989

TEXAS A&M UNIVERSITY

VERSION 1.2 PAGE 1

ROUTE: 1 DUNBAR EAST

SEGMENT: I101 IN RT1 SEG1

DIRECTION: 0

RUN #:

START TIME: 16:00

DATE: TU 01/17/1989

 * BASIC RIDE CHECK REPORT *

LOCATION	TIME	---BOARDINGS BY FARE CATEGORY---									
		BOARD INGS	ALIGHT INGS	ON BOARD	FARE CAT1	FARE CAT2	FARE CAT3	FARE CAT4	FARE CAT5	FARE CAT6	FARE CAT7
TEAK & 29TH		3	2	13	3	0	0	0	0	0	0
TEAK & 28TH		1	1	13	1	0	0	0	0	0	0
TEAK & 27TH		3	1	15	3	0	0	0	0	0	0
TEAK & REDBUD		1	2	14	1	0	0	0	0	0	0
26TH & QUIRT		5	3	16	5	0	0	0	0	0	0
QUIRT & 25TH		0	5	11	0	0	0	0	0	0	0
QUIRT & 24TH		1	0	12	1	0	0	0	0	0	0
24TH & OAK		2	1	13	2	0	0	0	0	0	0
24TH & HICKORY		1	3	11	1	0	0	0	0	0	0
26TH & GLOBE		2	1	12	2	0	0	0	0	0	0
TOTAL		19	19		19	0	0	0	0	0	0

RPT 2.1.1 01/17/1989

TEXAS A&M UNIVERSITY

VERSION 1.2 PAGE 1

ROUTE: 1 DUNBAR EAST

SEGMENT: I103 IN RT1 SEG3

DIRECTION: 0

RUN #:

START TIME: 16:00

DATE: TU 01/17/1989

* BASIC RIDE CHECK REPORT *

LOCATION		TIME	BOARD INGS	ALIGHT INGS	ON BOARD	---BOARDINGS BY FARE CATEGORY---						
						FARE CAT1	FARE CAT2	FARE CAT3	FARE CAT4	FARE CAT5	FARE CAT6	FARE CAT7
17TH &	AVE D		0	1	8	0	0	0	0	0	0	0
17TH &	AVE E		0	2	6	0	0	0	0	0	0	0
AVE E &	16TH		0	3	3	0	0	0	0	0	0	0
AVE E &	15TH		0	0	3	0	0	0	0	0	0	0
AVE E &	14TH		0	1	2	0	0	0	0	0	0	0
AVE E &	13TH		0	0	2	0	0	0	0	0	0	0
AVE E &	BROADWA		2	0	4	2	0	0	0	0	0	0
BROADWA &	AVE F		0	1	3	0	0	0	0	0	0	0
BROADWA &	AVE G		2	0	5	2	0	0	0	0	0	0
TOTAL			4	8		4	0	0	0	0	0	0

RPT 2.1.1 01/17/1989

TEXAS A&M UNIVERSITY

VERSION 1.2 PAGE 1

ROUTE: 1 DUNBAR EAST

SEGMENT: I102 IN RT1 SEG2

DIRECTION: 0

RUN #:

START TIME: 16:00

DATE: TU 01/17/1989

* BASIC RIDE CHECK REPORT *

		-----BOARDINGS BY FARE CATEGORY-----									
LOCATION	TIME	BOARD INGS	ALIGHT INGS	ON BOARD	FARE CAT1	FARE CAT2	FARE CAT3	FARE CAT4	FARE CAT5	FARE CAT6	FARE CAT7
26TH &	FIR	0	5	7	0	0	0	0	0	0	0
26TH &	EIM	2	0	9	2	0	0	0	0	0	0
26TH &	DATE	1	1	9	1	0	0	0	0	0	0
26TH &	CEDAR	1	0	10	1	0	0	0	0	0	0
CEDAR &	24TH	2	0	12	2	0	0	0	0	0	0
CEDAR &	23RD	2	3	11	2	0	0	0	0	0	0
CEDAR &	22ND	2	1	12	2	0	0	0	0	0	0
CEDAR &	21ST	6	0	18	6	0	0	0	0	0	0
CEDAR &	20TH	0	2	16	0	0	0	0	0	0	0
BIRCH &	20TH	0	3	13	0	0	0	0	0	0	0
BIRCH &	19TH	0	2	11	0	0	0	0	0	0	0
19TH &	AVE A	0	1	10	0	0	0	0	0	0	0
AVE A &	18TH	0	2	8	0	0	0	0	0	0	0
17TH &	AVE A	0	0	8	0	0	0	0	0	0	0
17TH &	AVE B	0	1	7	0	0	0	0	0	0	0
17TH &	AVE C	2	0	9	2	0	0	0	0	0	0
TOTAL		18	21		18	0	0	0	0	0	0

