

PB2001-101903



Design and Development of an Automated  
NEMA Traffic Signal Controller Tester  
State Job No. 14646 (0)

Final Report

Prepared by:

James C. Gilfert  
Principal Investigator  
and  
Ted A. Gilfert

August, 2000

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



Athens Technical Specialists, Inc.  
8157 US Route 50  
Athens, Ohio 45701

REPRODUCED BY: **NTIS**  
U.S. Department of Commerce  
National Technical Information Service  
Springfield, Virginia 22161



Design and Development of an Automated  
NEMA Traffic Signal Controller Tester  
State Job No. 14646 (0)

Final Report

Prepared by:

James C. Gilfert  
Principal Investigator  
and  
Ted A. Gilfert

August, 2000

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



Athens Technical Specialists, Inc.  
8157 US Route 50  
Athens, Ohio 45701



1. Report No. FHWA/OH-2000/013	2. Government Accession No.	3. Recipient's Catalog No.	
4. Title and Subtitle DESIGN AND DEVELOPMENT OF AN AUTOMATED NEMA TRAFFIC SIGNAL CONTROLLER TESTER		5. Report Date August, 2000	
		6. Performing Organization Code	
7. Author(s) James C. Gilfert, Ted A. Gilfert		8. Performing Organization Report No.	
		10. Work Unit No. (TRIS)	
9. Performing Organization Name and Address Athens Technical Specialists, Inc. 8157 US Route 50 Athens, OH 45701		11. Contract or Grant No. State Job No. 14646(0)	
		13. Type of Report and Period Covered Final Report	
12. Sponsoring Agency Name and Address Ohio Department of Transportation 1980 West Broad Street Columbus, OH 43223		14. Sponsoring Agency Code	
		15. Supplementary Notes Prepared in cooperation with the U.S. Department of Transportation, Federal Highway Administration	
16. Abstract Traffic signal system controllers designed to meet the NEMA TS1 Standards are one of the most prevalent types in use in the US and Canada as of this writing. A PC-based tester for testing NEMA TS1 traffic signal controllers has been developed as a part of this program. This tester may be used to verify compliance of controllers to the NEMA TS1 Standards. Test selection, data management, and data processing are controlled via familiar point-and-click menus and buttons on the display of the PC. Precise timing of the test conditions and test result reporting are handled by a separate microprocessor, located in the hardware interface. The user interface offers multiple levels of test result processing, including comparison of the results to user-specified tolerance limits for acceptable performance.			
17. Key Words Traffic, Controller, NEMA TS1, Traffic Control, Tester, Test <sup>2</sup> Equipment		18. Distribution Statement No Restrictions. This document is available to the public through the National Technical Information Service, Springfield, Virginia 22161	
19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages	22. Price



## Executive Summary

"Design and Development of an Automated NEMA Traffic Signal Controller Tester"  
FHWA Report No. FHWA/OH-2000/013

August, 2000

Agency: Athens Technical Specialists, Inc.

State Job No. 14646(0)

Principal Investigator: Dr. James C. Gilfert

Notice: For copies of this report, contact: Ohio Department of Transportation, Mr. Roger L. Green,  
(614) 995-5993, roger.green@dot.state.oh.us

### 1.0 Objectives of the Project

The overall objective of this project was to design and develop an automated tester for traffic signal controllers which verifies the compliance of the controller to the applicable portions of the NEMA TS1-1989 standard. The tester is capable of automatically presenting test conditions to the controller and recording the responses of the controller. The responses are analyzed by the system software to compute the response times and other performance characteristics, and these results are stored in a database. The system software will also compare the results to a set of performance criteria, established by the user, and create a "score card" report to indicate if the controller passed or failed the test. The anticipated uses of the NEMA Traffic Signal Controller Tester are to diagnose damaged controllers, certify repaired controllers, and acceptance testing of new product.

### 2.0 Description of Work Performed

Previous work performed for the Ohio Department of Transportation was studied to capitalize on the pertinent existing base of knowledge. Discussions were held with traffic signal controller repair technicians to understand the types of information that are important to diagnosis and to achieving compatibility between controllers and related equipment in the traffic control system. The NEMA TS1 standard was studied to define the actual test conditions required. The electrical interface to the tester and the control electronics were designed and fabricated. Then the actual test routine software was written and tested. The system utilizes a notebook PC with the Microsoft Windows 95 operating system to act as an operator interface, send instructions to the tester, and receive test results and data back from the tester. The user interface and tester control software was also written and tested. The entire system was then assembled and de-bugged and operating manuals were written.

### 3.0 Conclusions and Recommendations

One significant conclusion, reached by observing the results of tests upon various makes and models of controllers, is that significant deviations from the NEMA standards can exist in controllers which appear to operate in a satisfactory fashion. The tester designed under this project can detect and document these deviations from the standard, and should be used for acceptance testing of new product by public agencies that are purchasing new product with public funds. The result should be that, over time, a higher level of interchangeability between makes of product should be achieved, which will benefit all purchasers of NEMA TS1 traffic system controllers. The techniques developed under this project for managing the amount of data required to be manually reviewed have led to the acceptance of the prototype unit by the controller technicians. The contractor for this project is planning to utilize this design and the technology developed to produce a commercial product for the traffic signal test equipment market.



## Acknowledgements

The authors would like to acknowledge the support and assistance provided by the members the Ohio Department of Transportation, Office of Traffic Engineering during the course of this project. Specifically, Rodger Dunn, Don Snyder, Homer Suter, Wally Richardson, Bob Duemmel, Mike Teaford, Bobbie Williams, Terry Martin, Mike Wotring, and Ed Vincent should be recognized as contributors to the overall result.

## Disclaimer Statement

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the data presented herein. The contents do not necessarily reflect the official views or policies of the Ohio Department of Transportation or the Federal Highway Administration. This report does not constitute a standard, specification, or regulation.

## Copyright Notice

This report and all software developed under this program are Copyright 2000, Athens Technical Specialists, Inc. All rights are reserved, and duplication without written permission (except by the State of Ohio or the U. S. Federal Highway Administration for official purposes) of Athens Technical Specialists, Inc. is prohibited by Federal law.

## Acquisition Notice

For copies of this report, contact:

Ohio Department of Transportation  
Mr. Roger L. Green  
(614) 995-5993  
roger.green@dot.state.oh.us

**PROTECTED UNDER INTERNATIONAL COPYRIGHT  
ALL RIGHTS RESERVED  
NATIONAL TECHNICAL INFORMATION SERVICE  
U.S. DEPARTMENT OF COMMERCE**

Reproduced from  
best available copy.





# Table of Contents

I. Introduction .....	6
II. Testing Philosophy and Implementation .....	7
III. Controller Parameters Measured or Tested.....	9
IV. Overview of Tester Operation.....	10
V. Conclusions .....	18

## Appendix A

### NEMA TS1 Controller Tester Operating Manual

<b>General.....</b>	<b>A 4</b>
Licensing Agreement.....	A 4
Computer System Requirements .....	A 4
Software Installation/What Gets Installed Where .....	A 5
Getting Started.....	A 6
NEMA Controller Tester Software .....	A 6
Creating and Selecting Data Sources.....	A 6
The Application Interface.....	A 7
The Menus.....	A 7
The Tabs.....	A 8
Selecting Communications Options.....	A 8
<b>Using the NCT Software .....</b>	<b>A 9</b>
Running a New Test .....	A 9
The User Interface .....	A 10
File Menu .....	A 10
New Test.....	A 10
Load Test.....	A 10
Import Test File.....	A 10
Export Test File.....	A 10
Close.....	A 10
Exit.....	A 10
Test Menu.....	A 11
Run Next Test.....	A 11
Cancel Run .....	A 11
Execute Tests .....	A 11
Abort Tests .....	A 11
Reset Software .....	A 11
Templates.....	A 11
Comm Settings .....	A 11
Check Tester .....	A 11
Analysis Menu.....	A 11
Re-Analyze Test .....	A 11

Manage Criteria Specs .....	A 11
Test Setup Tab .....	A 11
Creating and Using Templates .....	A 13
Functional Display Tab .....	A 13
Timing Display Tab .....	A 14
Tester Output Tab .....	A 15
Raw Data Tab .....	A 16
Reduced Data Tab .....	A 17
Scorecard Tab .....	A 18
<b>Loading an Existing Test.....</b>	<b>A 18</b>
<b>Scoring a Test .....</b>	<b>A 19</b>
Interpreting the Scorecard For a Functional Test .....	A 20
Interpreting the Scorecard For a Timing Test.....	A 21
<b>Creating Criteria Files .....</b>	<b>A 22</b>
<b>Exporting and Importing Specific Tests .....</b>	<b>A 22</b>
<b>Test Modes .....</b>	<b>A 23</b>
Normal Mode .....	A 22
Diagnostic Mode .....	A 23
Verify Mode.....	A 23
<b>Test Descriptions .....</b>	<b>A 24</b>
Functional Tests .....	A 24
Timing Tests .....	A 26

# IMPLEMENTATION OF AN AUTOMATED NEMA CONTROLLER TESTER

## I. Introduction

The past fifty years have seen a dramatic increase in the capabilities of traffic signals. These new capabilities became possible primarily as a result of an unending stream of technological advances in electronic device fabrication starting from the first transistor in 1947, up to the multi-million-transistor chips that comprise modern microprocessors, memories, and related complex devices. The fundamental change from analog timing and storage to the corresponding digital implementations which occurred in the late sixties and seventies allowed accurate realization of signal storage and processing concepts which at best were poorly approximated with prior technology.

The availability of precise timing and parameter storage lent credibility to the concept of interchangeability of like assemblies (controllers, monitors, etc.) from various manufacturers. The National Electrical Manufacturers Association (NEMA) started to develop a set of standards for traffic signals in the early seventies, and it was released in 1976 as the NEMA Standards Publication No. TS1. A separate group, consisting primarily of representatives of the Federal Highways Administration (FHWA), California Department of Transportation (Caltrans), and New York State Department of Transportation (NYDOT), released a separate set of standards as Implementation Package FHWA-IP-78-16 in 1978. These two sets of standards agreed on most performance parameters, although the hardware implementations differed significantly. Both sets of standards have experienced revisions in these ensuing years to improve the safety and efficiency of vehicular movements. All modern manufacturers of signal components claim that their products are built to conform to the prevailing standards.

Unfortunately, it is no simple task to verify conformance with standards for a modern signal component such as a controller or monitor. The complexity of setting up precise tests, and then applying such tests to a multitude of phases or channels, then evaluating and reporting the results in readily-comprehended format, virtually mandates an automated tester. Recognizing the importance of an automated controller tester for all agencies which maintain signals, ODOT contracted with ATSI to develop a tester for the NEMA TS1 controllers that were commonly used at that time. That program, initiated in mid-1996, is now complete, and this report relates the results obtained under that program. This report consists of two sections, an executive summary and an operating manual for the tester developed. The continuation of this section is the executive summary, while the operating manual is a separable text which identifies all necessary setup and operating procedures, and provisions for extraction of the desired information in accordance with the current test objectives.

## II. Testing Philosophy and Implementation

An 8-phase NEMA TS1 Controller has 64 signal input lines and 92 signal output lines in addition to its AC input power, earth ground, logic ground, and 24VDC output power lines. All controller input and output signals are based on 24V negative-true logic. A tester must be capable of driving each of the controller's input lines with valid logic signals and capable of sensing the logical state of each of the controller's output lines. The NEMA TS1 Standard states that the durations of all controller-generated intervals are to lie within 100mS of the programmed values, as defined by an AC-line-based timing standard.

It was recognized early on that testing the controller was a major task. For each desired test, a given set of inputs had to be presented to the controller, the resulting output from the controller had to be collected over some seconds following the presentation of new input data, and finally the data had to be compared against the standards-specified values to determine whether the standards were satisfied.

Unlike the testing of monitors and detectors, the controller proceeds in accordance with its own internal programming. Once a set of input conditions is applied to place it into motion, the controller will come to rest again only after all requests for service have been satisfied. This behavior mandates a control structure which can provide inputs and modify the controller's configuration on-the-fly, while normal controller sequences are in progress. To obtain predictable responses from the controller, a special testing program must be used in the controller during the testing process. The special testing program uses short timing intervals for each phase, to reduce the time required for a given set of tests.

The magnitude of this task suggested partitioning the testing task into a "supervisor-slave" configuration. The supervisor provides the user interface, fully compatible with contemporary Windows 95 and Windows 98 operating systems. In this familiar Windows environment, the user first defines the characteristics of the controller to be tested, then selects the desired test(s) from lists of available tests. When this test setup and test selection process is complete, the supervisor sends along a list of the selected test number(s) to the slave processor. The slave then uses these test numbers and conditions to access appropriate sections of a large look-up table which provides sequences of specific inputs to be applied to the controller. The slave presents these inputs, then gathers the resulting controller outputs, formats them, and sends them back to the supervisor. The supervisor then parses the slave's report strings and enters the components into a data base for future recall and compliance checks. During the collection process, the supervisor provides the user's choice of two displays, either the text strings as received directly from the slave, or the corresponding display of signal-head lamps (both phases and overlaps) and pedestrian heads, as derived from the text strings. The huge gain of this partitioning is that both processors can be doing their assigned tasks simultaneously, almost independently of the other, tasks for which each processor is optimized, affording major savings in the time required to achieve comprehensive testing.

The slave function was first addressed, to verify the implementation concept. A Motorola 68HC11 processor was chosen primarily for its rich set of timer-based interrupts, several of

which are used in this design. The testing philosophy selected was one in which a new set of input signals would be presented simultaneously to the controller, and a sequence of "snapshots" of the controller's response to these inputs would be taken at uniform time intervals following that set of new inputs. The first tester built to this philosophy implemented several basic tests with data collected at a rate of 60 snapshots per second, synchronized to the AC power line, as required by the NEMA standards. Very gratifying results were obtained, along with some clear indications of where this concept should be improved. Logical conditions were applied to the slave's raw data input to strip out those report lines in which no lamp changes occurred, thereby removing all redundancy and reducing the requirements for processing new data. Further testing now confirmed that the slave could successfully test at a rate of 120 samples per second, still synchronized with the power line. This very fast data sampling allows a determination of controller-generated time intervals to an accuracy of 8.33 milliseconds. Since new input data can be applied to the controller by the same interrupt structure which reads the controller status, the timing of external inputs to the controller is also known to a precision of 8.33 milliseconds.

Following this initial success with some basic timing measurements, the NEMA TS1 Standard text was studied for all testable conditions specified for signal controllers. Most timing parameters are programmable on a phase-by-phase basis, and clearly called for compliance testing on every phase. Examples here are timing of initial greens, yellows, vehicle clearances, red-clearances, walks, pedestrian clearances, maximum green 1 and 2, added-initial green timings, gap reduction, etc. Some non-timing functions are also selectable on a phase-by-phase basis, such as hold, phase-omit, and ped-omit, and similarly require testing on every phase. Other functions are unit-wide, such as power-on start and external restart, manual control and interval advance, overlap programming, etc. To facilitate the early stages of test development, a primitive supervisor structure was implemented within the slave's program.

Concurrent with the expansion of test options, the Windows-based supervisor structure took form. A serial-data communications format was selected, and a command-line structure was devised appropriate to the information needed to define each test uniquely.

As the specifications grew into tests, these many timing and non-timing conditions were grouped into timing tests and functional tests. Thirteen timing tests were developed, several of which combine two different parameter measurements within a single test. All timing tests and those functionals which are activated on a phase-by-phase basis are executed for every phase present in the controller. Several ring-wide and unit-wide control functions, such as manual enable and interval advance, were also tested on a phase-by-phase basis where full verification was deemed important. Eight-phase controllers are tested as dual-ring controllers, with identical test conditions applied to the paired phases of the two rings. Ten functional tests were developed for evaluating the controller's conformance with non-timed standards.

A special test program is used for all controllers of a given number of phases, so that a single set of results, as obtained from a perfect controller, serves as the standard for compliance with the NEMA specifications. Therefore, the controller must be configured to a

corresponding set of timings and related conditions prior to testing. These timings and other conditions are not necessarily representative of conditions which would be found at typical signal installations: Rather, they are chosen to allow testing to progress at a reasonably brisk pace, to minimize testing time, but without sacrificing accuracy or completeness. Following the compliance testing, the controller is restored to the timings and other settings appropriate to its target intersection, and the tester can be requested to report those timings found, allowing the technician to verify that the controller was properly re-programmed for its target intersection.

### III. Controller Parameters Measured or Tested for Functional Operation

#### A. Timing Parameters, tested for every phase

1. Minimum Initial Green, tested at 7 seconds.
2. Maximum Initial Green, tested at 11 seconds.
3. Added Initial Green, tested at 1 second/actuation
4. Vehicle Passage Time (Green Extension), tested at 5 seconds.
5. Yellow Clearance, tested at 3 seconds.
6. Red-Clearance, tested at 1 second.
7. Pedestrian Walk, tested at 3 seconds.
8. Pedestrian Clearance Interval, tested at 7 seconds.
9. Maximum Green 1, tested at 20 seconds.
10. Maximum Green 2, tested at 25 seconds.
11. Red Revert, tested at 2 seconds
12. Time-To-Reduce, tested at 8 seconds
13. Time-Before-Reduction, tested at 25 seconds.
14. Minimum Gap, tested at 2 seconds.

#### B. Functional Operation, tested for every phase

1. Vehicle Calls, Detection and Storage
2. Pedestrian Calls, Detection and Storage
3. Phase Hold Function
4. Phase Omit Function
5. Red-Clearance-Omit Function
6. Manual Enable Function
7. Interval Advance Function
8. Force-Off Function.
9. Minimum Recall Function
10. Call to Non-Actuated Mode 1
11. Call to Non-Actuated Mode 2
12. Walk-Rest-Modifier Function
13. Pedestrian Recycle Function
14. Red-Rest Function
15. Inhibit Maximum-Green Function

## 16. Phase leaves Recall on Max-out Termination

### C. Functional Operation, tested on a unit basis

1. Power-Interrupt, Timing-box Test
2. Measure and Report Actual Voltage from 24VDC source
3. Check for CVM True in normal operation.
4. Check for Proper Response to External Restart Signal.
5. Check Duty Cycle of Flashing Don't-Walk
6. Measure and Report Minimum Detectable Duration of Call Signal.
7. Verify that Call Signal Recognizes Voltage Threshold Standards.
8. Measure and Report Symmetry and Duration of Flasher Output
9. Measure and Report Phase Difference between AC line and Flasher.
10. Compare Actual Overlap Displays against Program Card, Report Errors.

### D. Verify User Programming for every phase

1. Provide Data to allow User to Verify Actual Values of Min Initial Green, Yellow, Red-Clearance, and Actual Overlap Displays.
2. Provide Data to allow User to Verify Actual Values of Walk, Pedestrian Clearance Interval, Verify Red-Clearance-Omit Function.
3. Provide Data to allow User to Verify Actual Values of Max Green 1 Interval.
4. Provide Data to allow User to Verify Actual Values of Max Green 2 Interval.
5. Provide Data to allow User to Verify Actual Values of Green Extension Interval.
6. Provide Data to allow User to Associate Each Overlap uniquely with its corresponding phase(s). (8-phase controllers only.)

## IV. Overview of Tester Operation.

This section is intended to provide the reader with a slightly more intimate sense of the tester's operation by showing some typical monitor screens as seen by the user during the test selection, execution, and data extraction phases.

When the program boot is completed, the user is asked to select a database to work within, or to create a new database. The establishment of multiple databases allows the segregation of test reports into groups to meet the needs of the user. Once a data base for the task at hand is selected, the user is offered the choice of working with existing data from prior test sessions, or of performing new tests on untested units. Each test session has its own session number, and can be accessed by that number. On completion of a test session, the user has the option of saving or discarding the data of that session. Test data are saved to the selected database on the PC's hard drive, and remain there until they are deleted by the user, or the database is copied off to another location.

Assuming a new test session is desired, the user defines the controller to be tested, and selects the tests to be applied to that controller. It is assumed that the controller has already been programmed to the tester's standard test parameters. The Test Setup form from the Operator Interface is shown in Figure A below:

**NEMA Controller Tester**

File Edit Test Analysis Help

Test Setup | Func. Display | Timing Display | Tester Output | Raw Test Data | Reduced Data | Score Card

Record #: 8

Load Template | Save As Template

Phases: 8 | Laps: 1

Test Mode: Normal

Functional Tests | **Timing Tests**

- Veh. Timings/Storage, Min Recall
- Ped Timings and Storage
- Redclear Omit and Red Revert
- Hold Function on Actuated Phases
- Non-Actuated Hold, Force-Off, Etc
- Non Act. Wlk-Rst-Mod/Ped Recycle
- Volume Density1: Added Initial
- Volume Density2: Gap Reduction
- Green Extension with Max Inhibit
- Green Extension Timing with Max1
- Green Extension Timing with Max2
- Stop-Time and Interval Advance
- Manual Enable/Int- Adv/Ped Omit

Notes:

Offer Termination Options

Following Functional Tests     Following Each Timing Test

Select All | Clear All | Run Tests

New Session Created    9:01 PM    7/20/98

Start | Exploring - C:\ATSIwork\S... | **NEMA Controller Tester** | 9:01 PM

Figure A: Test Setup Form

The controller is identified by manufacturer's model and serial numbers, as well as the agency's inventory number. Selected timing tests are shown, and selected functional tests would be shown by clicking the "Functional Tests" tab. The user can add any special comments pertinent to this controller or the circumstances of this particular test.

After confirming correct entry of the controller description and test selections, the user commands the tester to execute the indicated tests. For functional tests, the only viewable data are those text strings coming from the slave, either a pass-fail indication or a numerical measure of the tested parameter. Refer to the Functional Display in Figure B below.

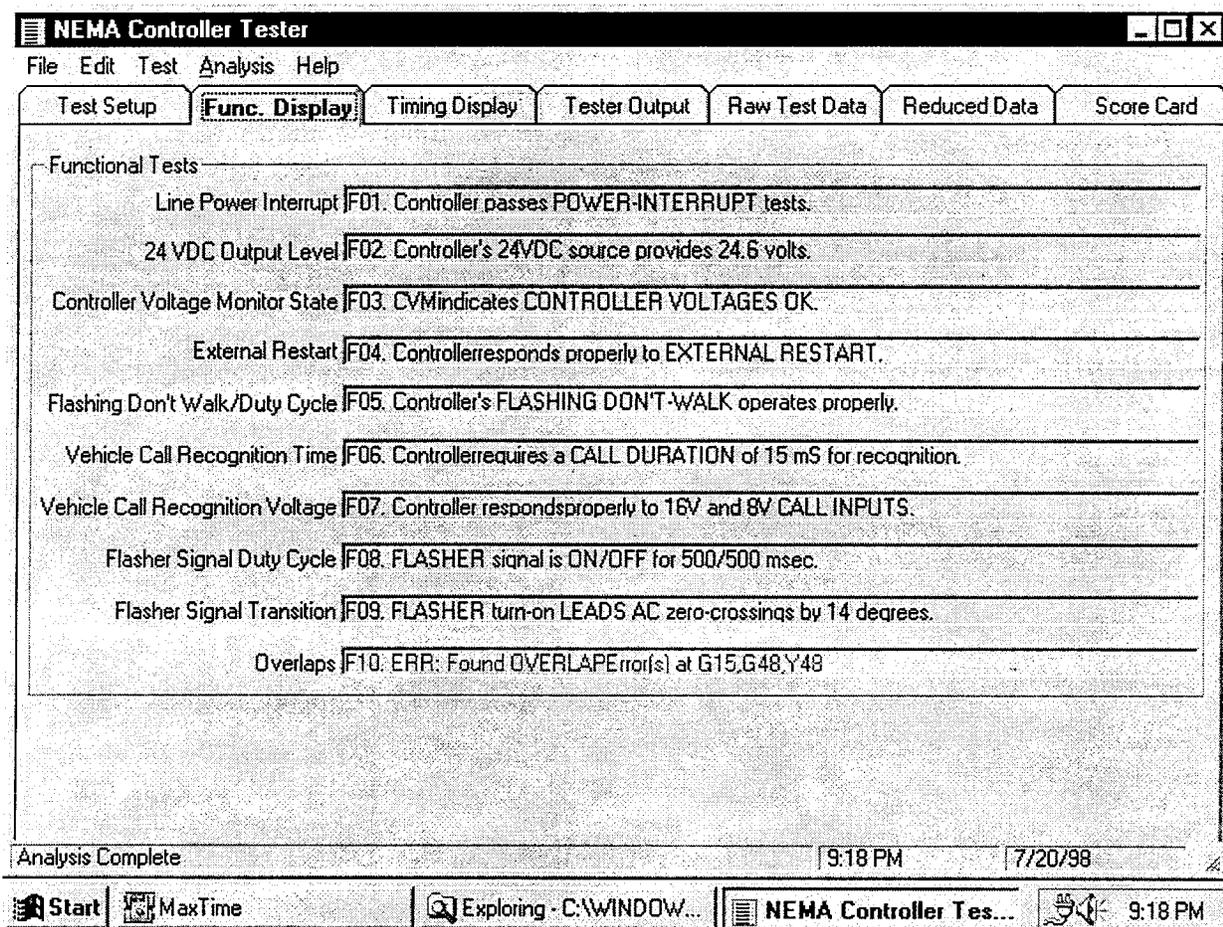


Figure B: Functional Display Form

The functional tests are those which do not relate explicitly to the controller's individual phases or their timing.

On completion of all selected functional tests, the tester proceeds to the timing tests, if any are to be performed. One user option is to observe the lamps as the various intervals are executed for each phase. An example of the Timing Display form is shown in Figure C below. As stated earlier, the tester sends a series of data strings to the Master PC running the operator interface program which describes the real-time state of all the output lines from the controller. This allows the operator interface to maintain a time-stamped record of the controller's behavior and to display a real-time status of what the signals would look like if the controller were running in a normal installation.

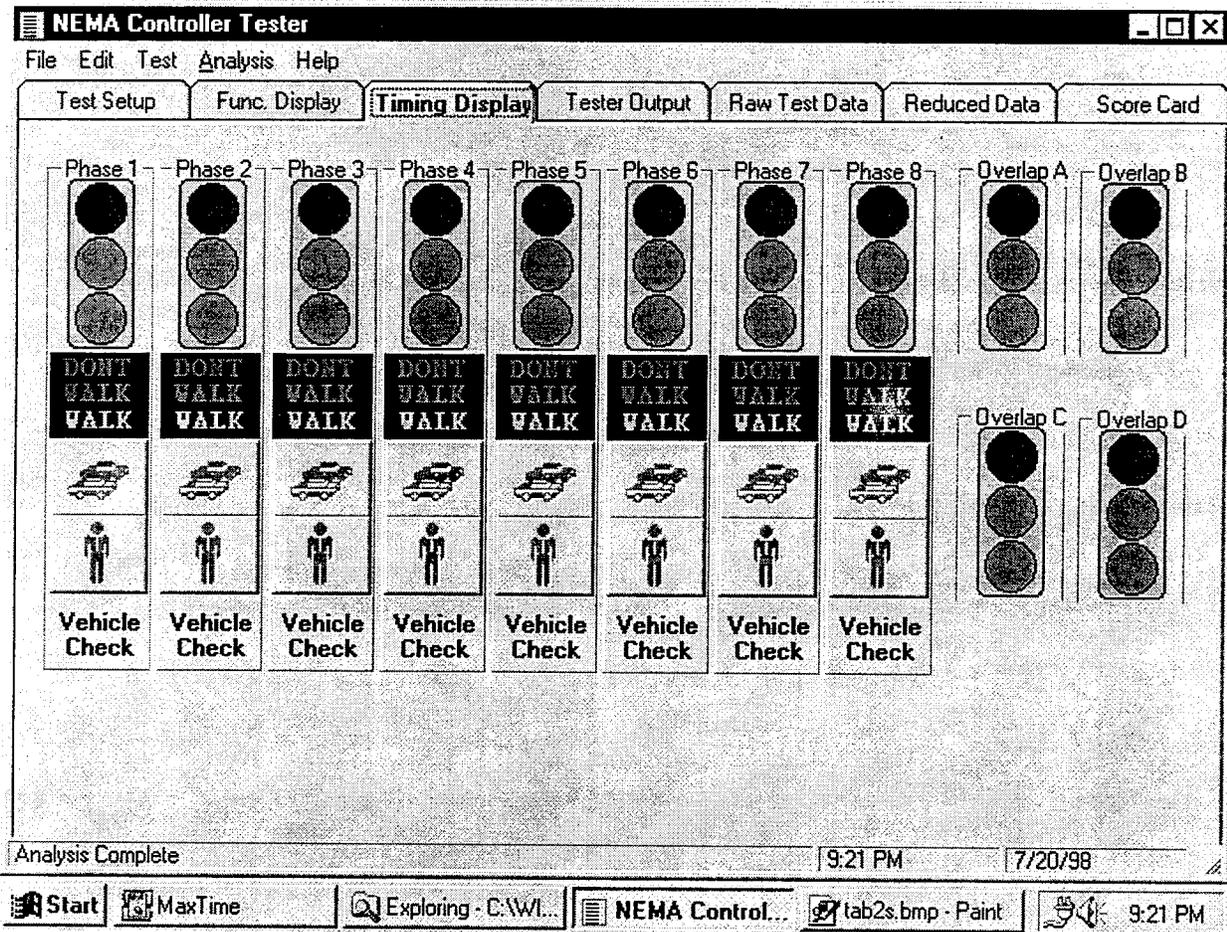


Figure C: Timing Display Form

This display presents, in real-time, the status of all vehicle and pedestrian signals, as well as vehicle and pedestrian calls. While this figure does not indicate colors, all lamp representations (on the computer) display their current colors at all times. The vehicle-call and pedestrian-call icons are similarly illuminated while the tester applies calls to these inputs. Vehicle check indicators show vehicle calls are logged to the associated phase which are not yet serviced.



When all the user-specified tests are completed, the user has the option of saving the session or discarding it. When the test session is saved, the software moves the timing data into a database which groups the data strings by the test name. The data can then be viewed in the Raw Test Data form, shown below in Figure E.

Assuming the test session data is saved by the user, it can now be manually reviewed and spot checked for abnormal sequences or to check a repaired function.

The results from the data analysis are stored in a Microsoft Access ® database by phase of the controller and lap (if the controller is tested multiple times) and can be viewed by opening the Raw Test Data form, as shown in Figure E below.

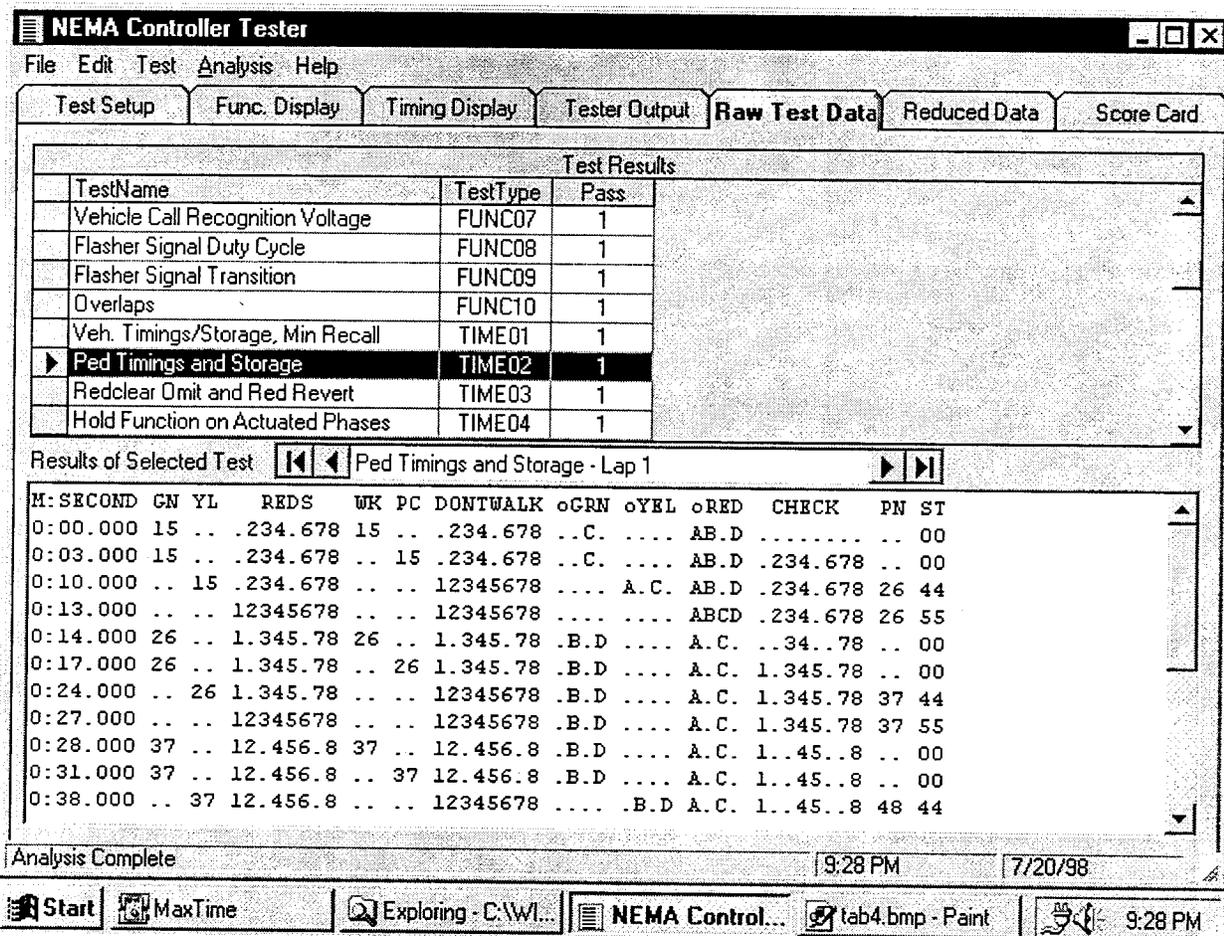


Figure E: Raw Test Data Form

These time-stamped lines of data have been recalled from the database after the test session has been concluded and saved. The results are grouped by test name, and can be easily recalled by clicking on the test name, shown on the upper half of the display.

When the test session is saved, the software performs an analysis on the timing data to time certain events for each of the Timing Tests, for instance the Phase 1 Yellow signal duration. By looking at Figure E, it is seen that Phases 1 and 5 Yellow come on at 10.000 seconds and go off at 13.000 seconds. The software times the events of the controller by analyzing the same data.

The results of the analysis are called Reduced Data and these results can be checked for conformance with the NEMA TS1 Standards or other performance criteria established by the user.

The Reduced Data are also stored in the database by phase of the controller and lap (if the controller is tested multiple times) and can be viewed by opening the Reduced Data form, as shown in Figure F below.

Analysis of Data Collected					
Test	Test Type	Phase	Lap	Result Category	Value
Flasher Signal Duty Cycle	FUNC08	0	1	Flash On (mS)	500
Flasher Signal Duty Cycle	FUNC08	0	1	Flash Off (mS)	500
Flasher Signal Duty Cycle	FUNC08	0	1	Flash Period (mS)	1000
Flasher Signal Duty Cycle	FUNC08	0	1	Duty Cycle (%)	50
Flasher Signal Transition	FUNC09	0	1	Transition Lead/Lag (degrees)	14
Overlaps	FUNC10	0	1	Implicit Result From Tester	FAILED
Veh. Timings/Storage, Min Recall	TIME01	9	1	Dual Indication Overlaps (mS)	3000
Veh. Timings/Storage, Min Recall	TIME01	9	1	Dual Indication Overlaps (mS)	3000
Veh. Timings/Storage, Min Recall	TIME01	1	1	Short Yellow Timing (mS)	3000
Veh. Timings/Storage, Min Recall	TIME01	1	1	Short Yellow Timing (mS)	3000
Veh. Timings/Storage, Min Recall	TIME01	2	1	Short Yellow Timing (mS)	3000
Veh. Timings/Storage, Min Recall	TIME01	3	1	Short Yellow Timing (mS)	3000
Veh. Timings/Storage, Min Recall	TIME01	4	1	Short Yellow Timing (mS)	3000

Analysis Complete 9:36 PM 7/20/98

Figure F: Reduced Data Form

This information has been calculated from the test data saved to the database, shown in Figure E (prior page). Phase 9 is Overlap A, Phase 10 is Overlap B, etc.

The Reduced Data can be automatically compared to a set of criteria that is established by the user. The criteria sets high and low limits for the numerical Reduced Data results and

passes or fails the test results based on those limits. The software comes with a default set of criteria, which may be edited by the user and saved under a new name. The same Reduced Data set can then be compared to different criteria sets, with varying tolerance levels for imperfect timing on the part of the controller. The ability to create various sets of criteria allows the user to vary the timing tolerances allowed based on the age and type of controller under test.

The report that is generated by comparing the Reduced Data to the criteria set is called the Score Card, and is set up and viewed in the Score Card form, as shown in Figure G below. On the left-hand side of the form is an index to the report, by test number, that allows the user to see the desired part of the report with a mouse-click.

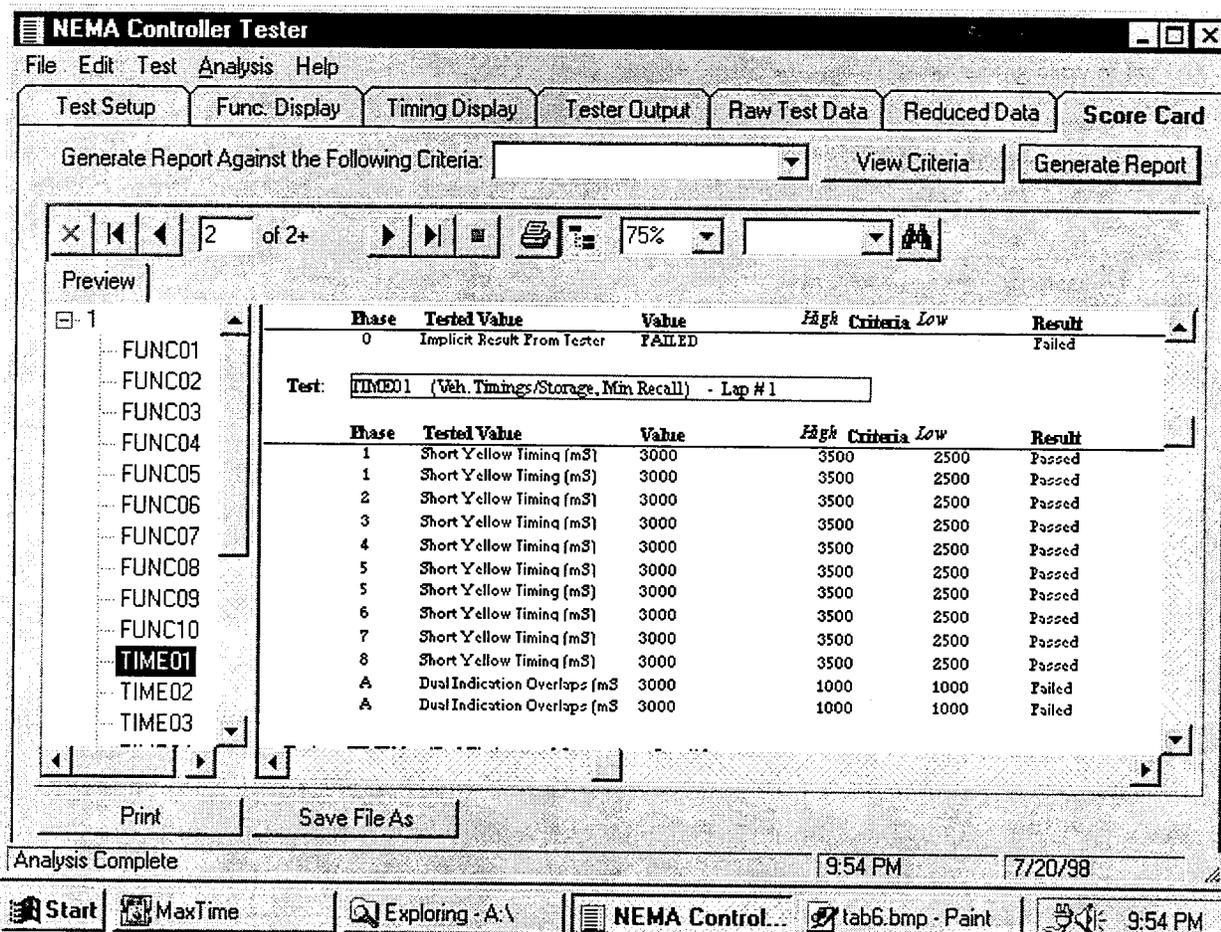


Figure G: Score Card Form

The Reduced Data (Figure F, prior page) are compared to sets of user-defined limits and judged as having passed or failed.

Considerably more detail on the features and operation of the NEMA Controller Tester can be found in Appendix A of this report. Appendix A is the User's Operating Manual for the NEMA Controller Tester.

## V. CONCLUSIONS

This summary has described an automated tester for use with the huge number of NEMA TS-1 controllers in use across the U. S. and Canada. Prior to the development of this tester, existing test equipment can, at best, offer a functional verification of the controller's compliance to standards, with no meaningful verification of the interval timings. Advanced NEMA features, such as volume-density timings, are totally beyond the capability of existing test equipment. The tester developed in this project not only addresses the shortcomings of existing instrumentation, but it also provides analysis software to verify compliance at a user-specifiable level, and facilitates performance-tracking of any controller, or any controller model, across months or years of service.



**Appendix A**  
**Operating Manual**  
**For the**  
**Automated NEMA Traffic Signal Controller Tester**

**August, 2000**

# Table of Contents

<b>General.....</b>	<b>A 4</b>
Licensing Agreement.....	A 4
Computer System Requirements .....	A 4
Software Installation/What Gets Installed Where .....	A 5
Getting Started.....	A 6
NEMA Controller Tester Software .....	A 6
Creating and Selecting Data Sources.....	A 6
The Application Interface .....	A 7
The Menus.....	A 7
The Tabs.....	A 8
Selecting Communications Options.....	A 8
<b>Using the NCT Software .....</b>	<b>A 9</b>
Running a New Test .....	A 9
The User Interface.....	A 10
File Menu .....	A 10
New Test.....	A 10
Load Test.....	A 10
Import Test File.....	A 10
Export Test File.....	A 10
Close.....	A 10
Exit.....	A 10
Test Menu.....	A 11
Run Next Test.....	A 11
Cancel Run .....	A 11
Execute Tests.....	A 11
Abort Tests .....	A 11
Reset Software .....	A 11
Templates.....	A 11
Comm Settings.....	A 11
Check Tester .....	A 11
Analysis Menu.....	A 11
Re-Analyze Test.....	A 11
Manage Criteria Specs .....	A 11
Test Setup Tab .....	A 11
Creating and Using Templates.....	A 13
Functional Display Tab .....	A 13
Timing Display Tab .....	A 14
Tester Output Tab.....	A 15
Raw Data Tab.....	A 16
Reduced Data Tab.....	A 17
Scorecard Tab .....	A 18
<b>Loading an Existing Test.....</b>	<b>A 18</b>
<b>Scoring a Test .....</b>	<b>A 19</b>
Interpreting the Scorecard For a Functional Test.....	A 20
Interpreting the Scorecard For a Timing or Verify Test .....	A 21

<b>Creating Criteria Files .....</b>	<b>A 22</b>
<b>Exporting and Importing Specific Tests .....</b>	<b>A 22</b>
<b>Test Modes .....</b>	<b>A 23</b>
Normal Mode .....	A 23
Diagnostic Mode .....	A 23
Verify Mode.....	A 24
<b>Test Descriptions .....</b>	<b>A 24</b>
Functional Tests .....	A 24
Timing Tests .....	A 26

# General

## Licensing Agreement

### Software License Agreement (SLA)

By using this software, you are agreeing to the terms of this SLA. If you cannot agree to these terms, please return the software to the vendor within 7 days of receipt for a refund.

Athens Technical Specialists, Inc. retains all rights of ownership to this copy of the software. It is licensed to the customer of record (you) under the conditions stated herein.

You may install the software on multiple computers as needed to efficiently utilize the tester hardware which it is designed to operate. You may make copies of this software for backup purposes. You may transfer this software license to another party by removing all copies from your system(s) and transferring the associated tester hardware along with the software, without retaining copies of the software. The receiving party of the transfer must agree to comply with the terms of the SLA.

You may not distribute, rent, sublicense, or lease the software or documentation or associated tester hardware. You may not translate, de-compile, dis-assemble, or reverse-engineer the software or associated tester hardware.

This license and your right to use this software automatically terminates if you fail to comply with any provision of this license agreement. Nothing in this license agreement constitutes a waiver of Athens Technical Specialists, Inc. rights under the U.S. copyright laws.

All software and media are covered by a 12 month limited warranty. Athens Technical Specialists, Inc. will replace the media or documentation at no charge to the original purchaser during the warranty period. Athens Technical Specialists, Inc. excludes any and all implied warranties, and limits your remedy to returning the software and documentation for replacement (or refund during the first 7 days after receipt). This limited warranty is void if failure of the software has resulted from accident, abuse, or misapplication. Any replacement software will be warranted for the remainder of the original warranty period or thirty (30) days, whichever is longer.

In no event will Athens Technical Specialists, Inc. be liable for direct, indirect, special, incidental, or consequential damages arising out of the use or inability to use the software, documentation, or associated tester hardware.

The limited warranty and remedies set forth herein are exclusive and in lieu of all others, oral or written, expressed or implied. Athens Technical Specialists, Inc. reserves exclusive authority to amend or modify this SLA, and may do so from time to time.

## Computer System Requirements

If the computer system is NOT connected to the Tester unit, then the system requirements are the recommended system requirements of the operating system. For more information on the operating system requirements, see your operating system manual.

If the computer system IS connected to the Tester unit, then it is necessary to have an installed Serial port configured as COM1 through COM4. It is highly recommended that the serial port contain a UART 16550 Chip or higher. The serial port needs to be able to support a sustained data rate of 9600 baud.

**The minimum computer hardware required to run this software is as follows:**

120 MHz Pentium or equivalent processor

16 MB RAM (32 MB recommended)

140 MB Hard Disk space

800 X 600 resolution display (recommended)

1 9-pin RS232 serial port (16550 UART)

4X CDROM (for software installation)

Microsoft Windows 95 operating system

## **Software Installation/What Gets Installed Where**

To install the NCT1-4000 tester software on the user's computer, first put the NCT1-4000 CD into the computer's CDROM drive according to the computer system instructions. After the drive is closed, click the Start button on the desktop and click the Run button on the menu. Using the correct drive letter for the CDROM drive (typically D: or E: ), type in the drive letter and the command "setup" similar to the following:

D: setup

Then click the OK button and follow the instructions that are displayed on the computer screen to complete the installation of the software.

During installation, the user designates where the software application should be installed, or can elect to let the setup program install to the default directory: C:\Program Files\Atsi\NEMA Controller Tester. If the default is selected, and does not exist on the host computer, the setup program will create the directories automatically. The file NCT1.EXE, System.M and Results.rpt are all placed in the application directory. It is very important that these three files stay together. In addition, several supporting system files are placed in the Windows System folder. Because of this, it is recommended that uninstalling this application, if ever required, be performed by using the Add/Remove Program Files icon in the Control Panel directory to insure no unneeded system files are left on your computer. The setup program will create a shortcut to the NCT1 program and install it on the desktop of the computer. To start the software, double-click the shortcut icon on the desktop. If you have questions on the use of the operating system of the computer, refer to the manuals or help files that came with the computer.

# Getting Started: Background

## NEMA Controller Tester Software

The NCT1-4000 software is a Windows 32 bit application requiring a computer running Microsoft Windows 95, or Microsoft Windows 98.

This software is designed to be used with a companion piece of equipment (the tester unit) that tests NEMA TS1 traffic signal controllers for compliance with published standards. The computer, with the software properly installed, will communicate with the tester, sending out commands and receiving test results, for the execution of the tests specified by the operator.

The application can be run on a computer not connected to the tester unit, for purposes of analyzing and consolidating the data. The NCT software interfaces with the tester unit using a standard serial port. The system stores data in a standard Microsoft Access database (Microsoft Access version 7 and up).

Users of the NCT1-4000 software are bound by the Software License Agreement (SLA) described at the beginning of this manual.

### To start the NCT1-4000 software:

- Double-click the “NEMA Controller Tester” Icon on the desktop or the shortcut on the Programs menu. This starts the application and begins by asking the user to select a data source as shown below (Figure 1).

### Creating and Selecting Data Sources

The NCT1 software stores the Tests, Results, Evaluation Criteria and Templates in a standard Microsoft Access .MDB file. The software allows for the creation of any number of database files and allows the user to select which database to use each time the application is started.

Every time the software is started the user is presented with the following dialog screen:

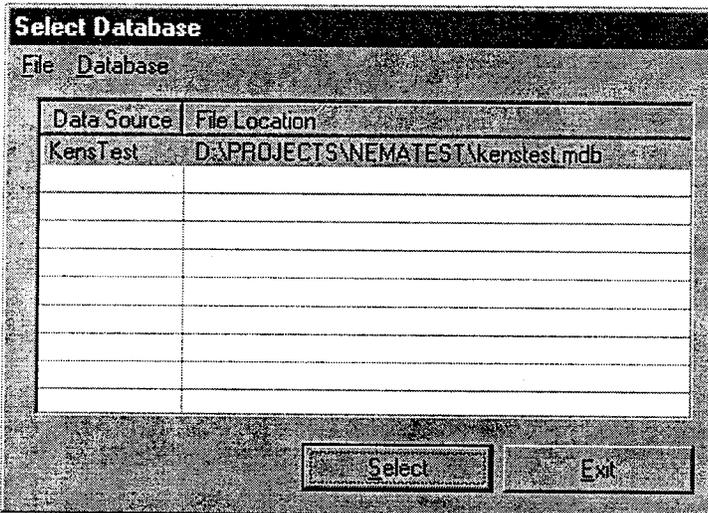


Fig 1. The Data Source Manager

### To select an existing database:

Highlight the desired database by clicking on it and press the Select button.

### To create a new database:

From the Database Menu, choose Create New Database. The Assign Data Source screen will come up (see Figure 2). Enter a Name for the data source, which is not (necessarily) the same as the database filename, but will make sense to the users of the tester. For example: "12-2001 Acceptance Tests" or "District 9 Repairs" or something else which meets the needs of the users. Then select the "Browse" button to assign the location on your system you would like the database to be stored. The location can either be an actual path\filename or a UNC Path to a network share. The most common location is in the default directory where the application is stored: C:\Program Files\Atsi\NEMA Controller Tester. The database filename must use the .mdb extension.

Once the data source entry is created, moving the file will cause the system to generate an error. If you want to change where your files are stored, create a new source with a new name and path to the new location. See "To add an existing file to the list:" below.

### To remove an entry from the list:

From the Database Menu, choose Remove Entry. This will remove the file from the entry. It **DOES NOT** delete the underlying file. The user must do this manually.

### To add an existing file to the list:

If it becomes necessary to move an existing file you must delete the entry for the file from the list (see above) then relocate the file to the location where you would like it stored. Then from the Database menu choose Add New Entry. The Assign Data Source dialogue will appear (fig 2). Enter in any name you would like and set the Database Path\Filename to the existing mdb file.

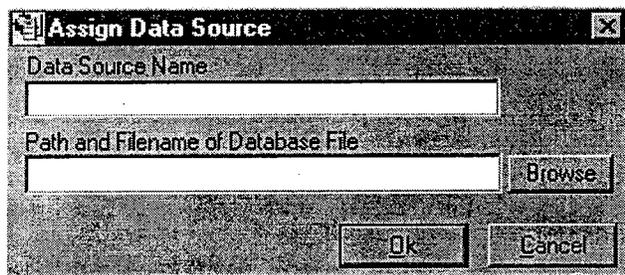


Figure 2. The Assign Data Source Dialogue

Once the Data Source selection is complete, the application opens the main user interface, described below.

## The Application Interface

The application runs primarily from a single form containing a tab control that has seven pages and a standard menu system. Refer to Figure 5.

## The Menus

The Main form contains the application menu system. The Edit and Help menus are the standard Microsoft Windows menus. For help on these, consult your Microsoft help documentation

The File Menu

The Test Menu

The Analysis Menu

## The Tabs

Page 1 - Test Setup (fields for user-entered data and test selection)

Page 2 - Functional Display (real-time display during test execution)

Page 3 - Timing Display (real-time display during test execution)

Page 4 - Tester Output (real-time display during test execution)

Page 5 - Raw Test Data (display of data as recalled from database)

Page 6 - Reduced Data (results and time intervals computed from test data)

Page 7 - Score Card (results and time intervals compared against user's criteria)

## Selecting Communications Options

The first item of business after the program is started and a data source set, it will be necessary to let the software know which serial port you wish to use to connect to the Tester unit.

**Before Setting the communications options, you must first connect the Tester unit to the computer and turn the tester unit power on.**

To set the Comm Settings, open the Test Menu of the Main Screen, and select the Comm Settings menu. The following Dialog will appear:

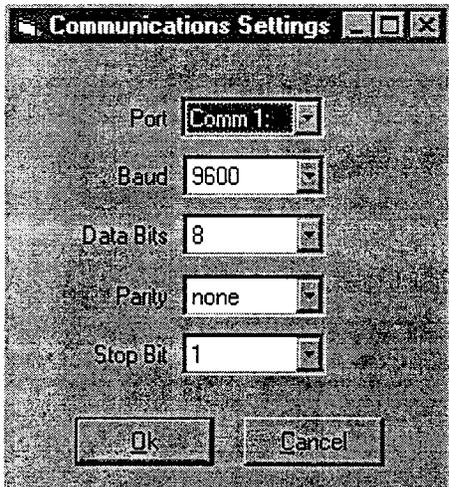


Fig 3. The Communication Settings dialog form

It is important to note that the only setting that should be changed is the Port entry unless instructed by ATSI. After setting the Comm port choose OK, a connection test will automatically start and the following screen will appear.

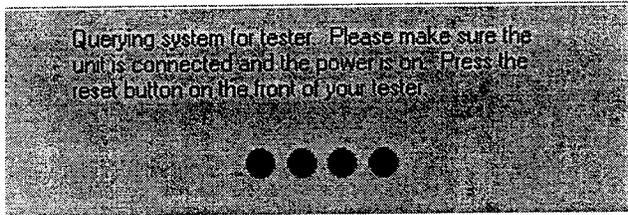


Fig 4. The Connection Message box

Press the reset button on the tester firmly. If the software is able to detect the Tester unit the dialogue will disappear automatically. If the NCT software cannot detect the tester unit it will continue to try for 30 seconds. Press the reset button on the Tester unit again. If the system still cannot detect the Tester, wait for it to time out, recheck your cables and Port Settings, and try again.

## Using the NCT1 Software

### Running a New Test

- Select the New Test menu entry from the File menu.

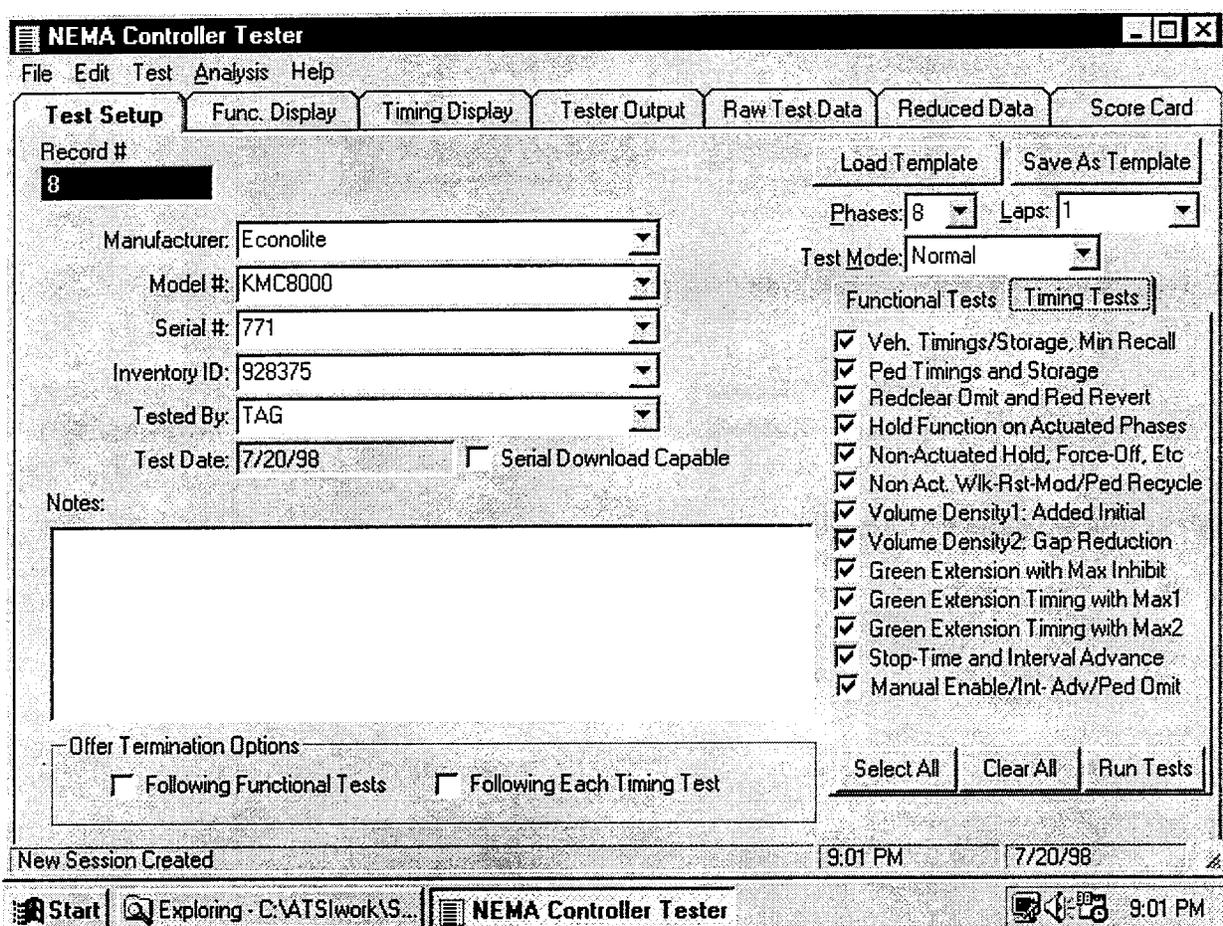


Fig 5. The Main User Interface

- The system will create a new session and assign it an ID. These IDs are used by the system to manage the session and are informational only.
- If you wish to load a template choose the Load Template button on the Test Setup tab. Otherwise enter the information about the Controller being tested, set the number of phases, the number of times to repeat the tests (Laps) and the test mode (See Test Modes).
- Select Normal, Diagnostic, or Verify Mode from the dropdown menu and select the tests you wish to perform (if Normal or Diagnostic mode) from the Functional/Timing lists (refer to Figure 5 above).
- Determine if you want the system to prompt you to continue after each Timing Test or each Function Test and check or uncheck the appropriate box in the Offer Termination Options area.
- Press the Run Tests Button (lower right) to begin the test session
- The NCT1 software will command the tester unit to execute each of the selected tests. The status bar on the bottom of the screen will tell you which test is currently executing.
- To watch the test run, use the Functional Display, Timing Display or Tester Output tabs.
- After the test is complete the NCT1 software will reduce the raw data into timing measurements of certain events. You may then view these events and times using the Reduced Data tab or apply a criteria set against the data to determine whether these timings are satisfactory using the Score Card Tab.

## The User Interface

### File Menu

Click on the File menu to display the dropdown list.

**New Test:** Initializes the software and prepares for a new test run (Figure 5).

**Load Test:** Opens the Load Test dialog (Figure 13) to allow stored data to be viewed.

**Import Test File:** This will import and save a test from a file that was created using the Export Test File Menu entry.

**Export Test File:** Takes the currently loaded test and exports it to an ASCII based file for loading into another database either on the current machine or on another copy of the NCT system elsewhere.

**Close:** Closes the current open test

**Exit:** Exits the NCT software

## Test Menu

Click on the Test menu to display the dropdown list.

**Run Next Test:** If you select the Prompting options, and the system prompts you to continue on, selecting this option will tell the software to continue.

**Cancel Run:** Terminates the current test series. If the system is running a session containing 10 timing tests and after the first test you wish to stop, selecting this option will cause the software to stop after finishing the current test. The current session will be closed and the data discarded. Unlike the Abort Tests menu entry (see below) , you are not prompted to save the data.

**Execute Tests:** Starts the current session. This menu option is identical to Run Tests button on the main form.

**Abort Tests:** Terminates the current test series. If the system is running a session containing 10 timing tests and after the first test you wish to stop, selecting this option will cause the software to stop after finishing the current test. The operator is then prompted to save the data collected so far.

**Reset Software:** This commands the software to ignore the Tester and cancel the test session, discarding all data. This option doesn't wait for the current test to finish its run, it simply ignores the Tester. This option is useful if the Controller 'locks up' during testing.

**Templates:** Brings up the Load Templates Dialog.

**Comm Settings:** Activates the Communications Settings Dialog (Figure 3).

**Check Tester:** Performs a communications test between the Tester and the Software (see Communications Settings)

## Analysis Menu

**Re-Analyze Test:** Causes the software to reduce the Data contained in the currently loaded test. This option is primarily used to reduce the data in an imported test, as only the raw data is imported and exported.

**Manage Criteria Specs:** Used to manage the criteria that is applied to the reduced data to produce the report. See Scoring a Test.

## Test Setup Tab

The Test Setup tab is where the information is entered that applies to the entire test session. It is also where the user specifies what tests will be executed.

- The Record # is the Session indicator and is a number assigned by the system
- The Manufacturer, Model #, Serial #, Inventory ID, Serial Download capable and Tested by fields are used to enter identifying information about the particular controller being tested and who is doing the test. These fields are drop down combo lists, so the user may either type in a value or select one from the list. The lists are generated dynamically from previous entries.
- The Test Date is assigned by the system and cannot be changed.

- The Notes field is used to add any comments the user wishes to enter. This is the only field that can be edited after a test has been run.
- Marking the Offer Termination Options checkboxes pauses the test session and allows the user to terminate or continue testing at the indicated junctures.
- The Load Template and Save Template buttons are for applying or saving templates to the current test, see Creating and Using Templates (next section) for more information.

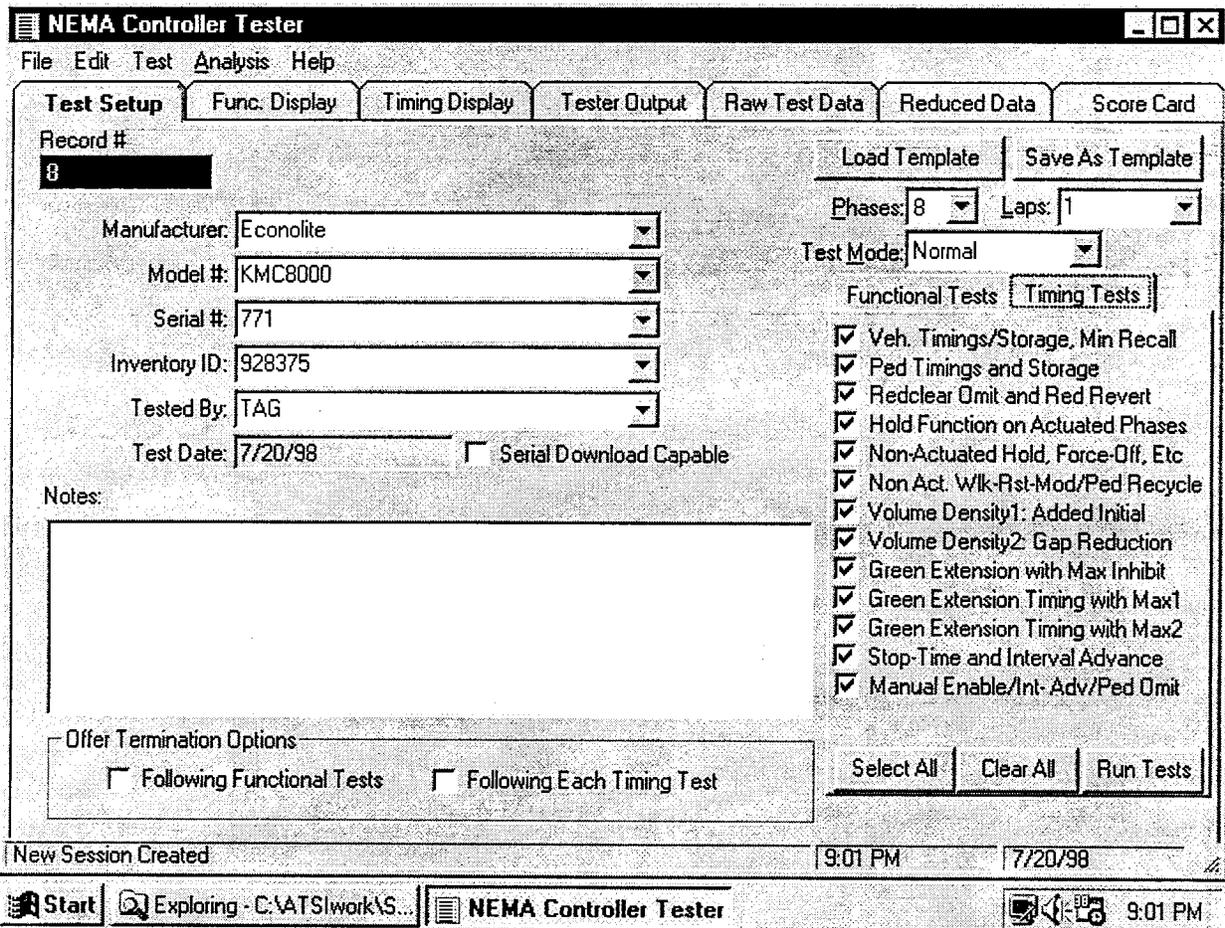


Fig 6. The Test Setup form

- The Phases dropdown must be set to the number of phases on the controller.
- The Laps box designates how many times the user wishes to repeat the selected tests.
- The Test Mode dropdown allows the user to select the desired test mode: Normal, Diagnostic or Verify. See descriptions in the Test Modes section.
- The Timing and Function Test Tabs allow the user to select the desired tests. See Timing Tests and Function Test for a description of what each test measures.
- Run Tests button prompts the user to press the reset button on the tester to verify communications then launches the selected tests.
- Select All and Clear All buttons are used to quickly select and deselect all tests, functional and timing.

## Creating and Using Templates

Templates store configuration data for test setup. They are a convenient way to run the same tests without having to set all the options in the Test Setup tab each time. Templates are created and recalled from the Test Setup form (Figure 6, above)

### To Create a Template:

Create a new test, using the File menu, then configure the options for the test just the way you want them. Then, before running the test, choose the Save as Template button from the Test Setup tab. The software will open a dialog box for the user to name the template.

### To load a Template

Create a new test, then choose the Load Template button on the Test Setup tab. The software opens a dialog box showing the available templates. Select the template you desire from the list. All the settings saved with that template are applied to the current test session, and the user may edit them as needed.

## Functional (Func.) Display Tab

The Functional Results tab displays the results of the most recent functional tests. So, if the same tests are being repeated for 8 laps, the results displayed reflect the most recently run lap. Implicitly provided errors are colored red. Test results are only visible in this tab during the actual test execution, before the results are saved into the database.

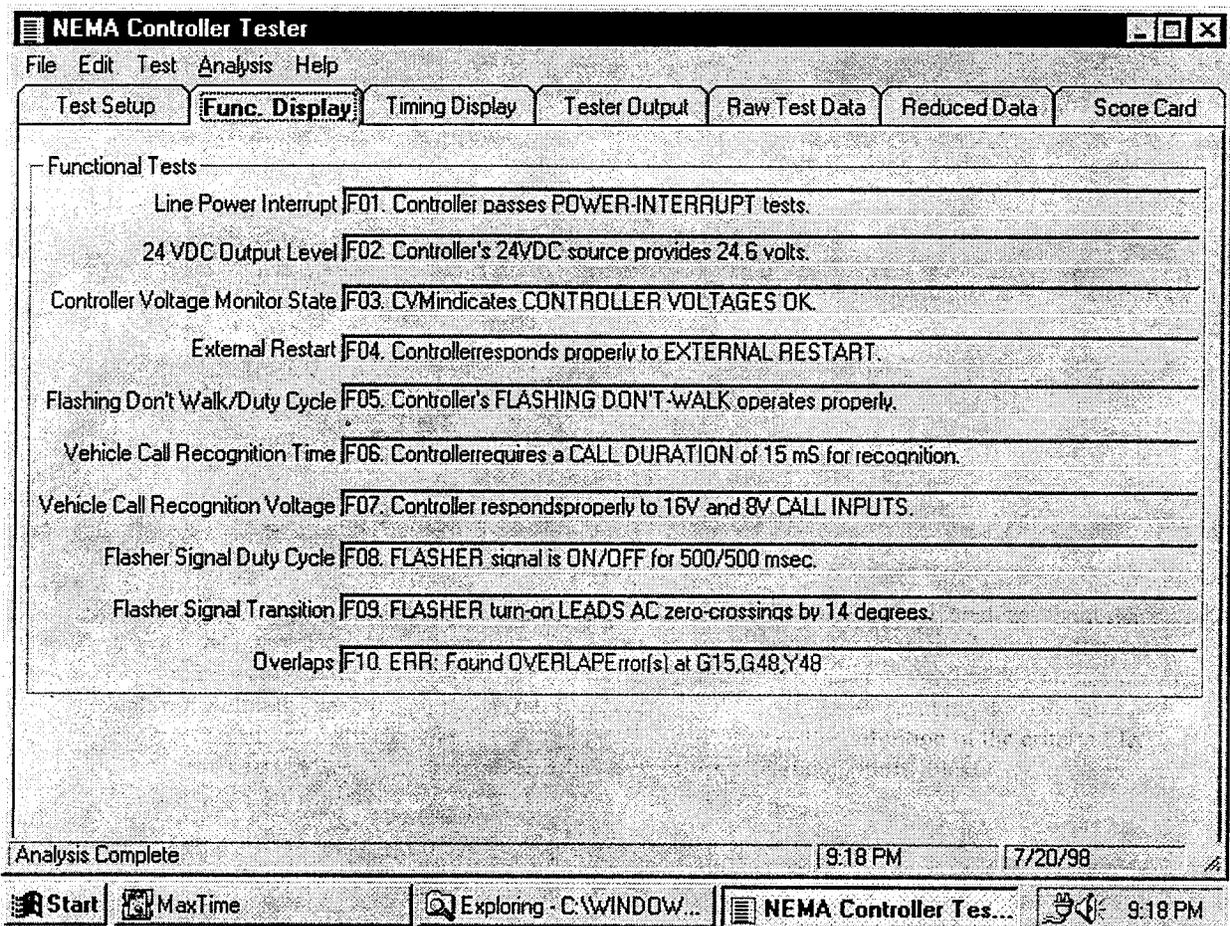


Fig 7. The Functional Display form

## Timing Display Tab

The time display provides a graphical representation of what the controller is doing at the specific point in time. Each phase is represented by a signal, Ped indicator, Vehicle and Ped Call indicators and a vehicle check status. Don't Walk displayed in YELLOW represents the Ped signal for Flashing Don't Walk. If any overlaps are present, they are displayed by a signal only. Test results are only visible in this tab during the actual test execution, before the results are saved into the database.

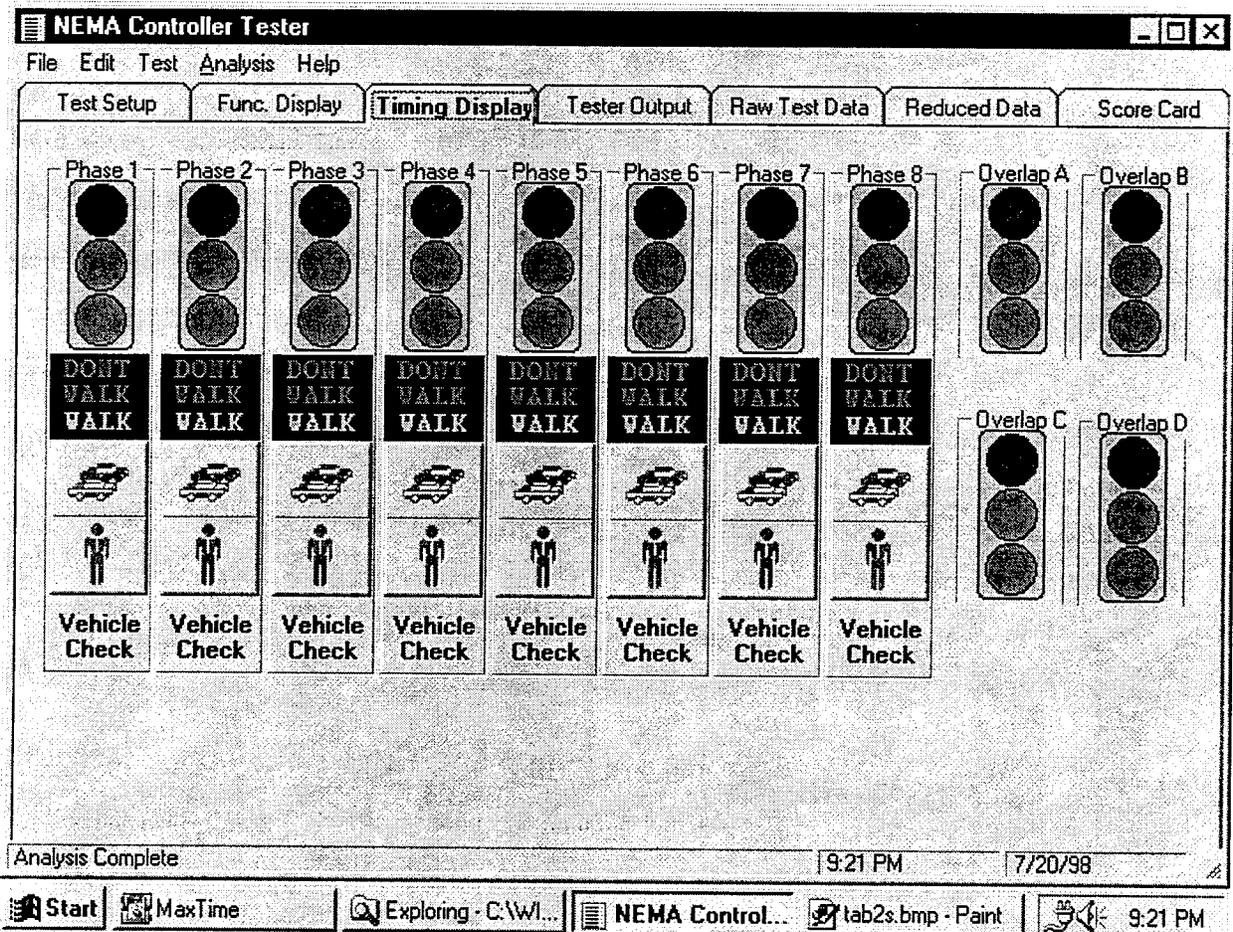


Fig 8. The Timing Display form

## Tester Output Tab

The Tester Output tab displays the actual data the PC and software is receiving from the tester for the currently running test. Test results are only visible in this tab during the actual test execution, before the results are saved into the database.

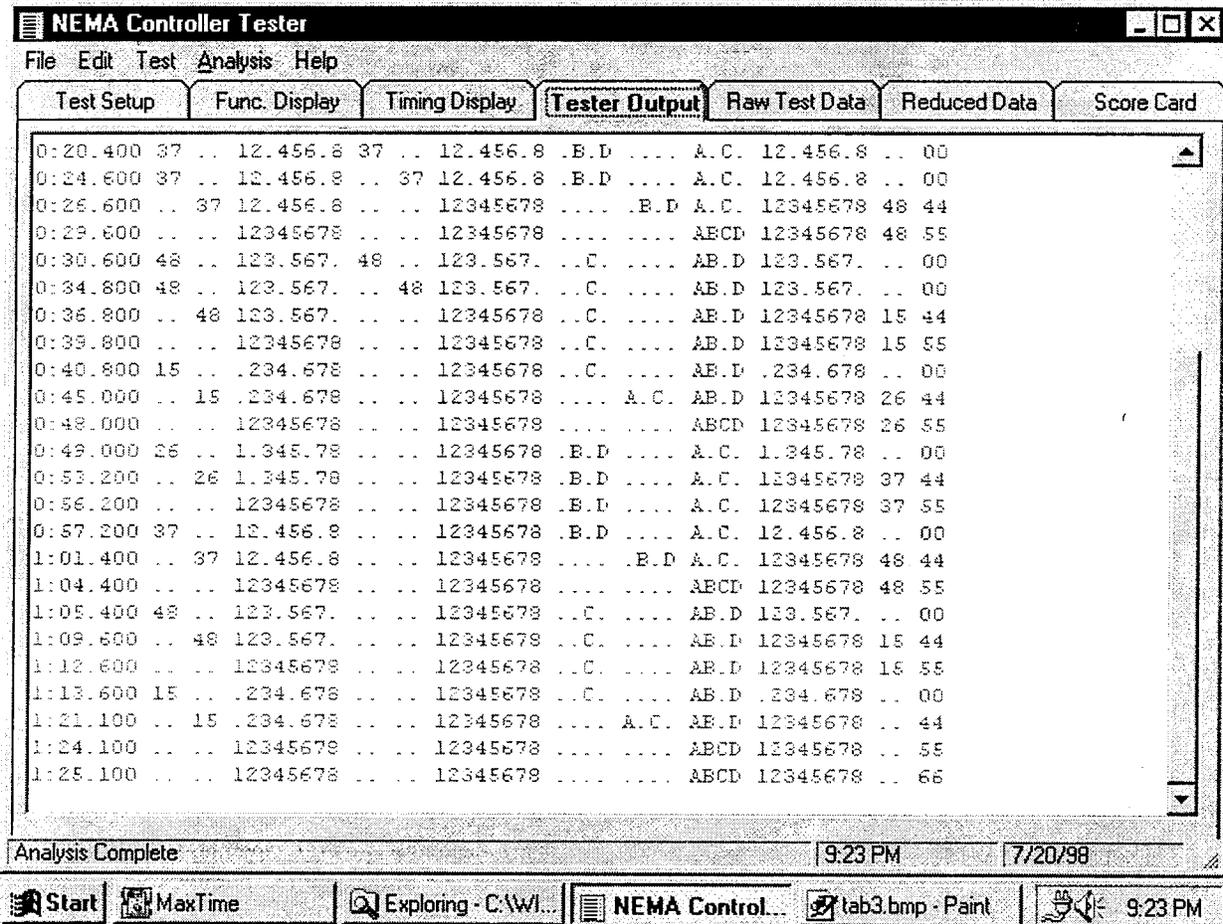


Fig 9. The Tester Output form

## Raw Test Data Tab

The Raw Data represents the actual data received from the tester and stored in the database for each test over the entire session. During the actual test execution, no data will be visible on this form. The grid on the top of the screen indicates how the data is grouped and stored in the database by test type. Selecting a specific test in the upper half of the screen will display the raw data on the lower half of the screen.

**Test Results**

TestName	TestType	Pass
Vehicle Call Recognition Voltage	FUNC07	1
Flasher Signal Duty Cycle	FUNC08	1
Flasher Signal Transition	FUNC09	1
Overlaps	FUNC10	1
Veh. Timings/Storage, Min Recall	TIME01	1
▶ Ped Timings and Storage	TIME02	1
Redclear Omit and Red Revert	TIME03	1
Hold Function on Actuated Phases	TIME04	1

Results of Selected Test: Ped Timings and Storage - Lap 1

M:SECOND	GN	YL	REDS	WK	PC	DONTWALK	oGRN	oYEL	oRED	CHECK	PN	ST
0:00.000	15	..	.234.678	15	..	.234.678	..C.	....	AB.D	.....	..	00
0:03.000	15	..	.234.678	..	15	.234.678	..C.	....	AB.D	.234.678	..	00
0:10.000	..	15	.234.678	..	..	12345678	....	A.C.	AB.D	.234.678	26	44
0:13.000	..	..	12345678	..	..	12345678	....	....	ABCD	.234.678	26	55
0:14.000	26	..	1.345.78	26	..	1.345.78	.B.D	....	A.C.	..34.78	..	00
0:17.000	26	..	1.345.78	..	26	1.345.78	.B.D	....	A.C.	1.345.78	..	00
0:24.000	..	26	1.345.78	..	..	12345678	.B.D	....	A.C.	1.345.78	37	44
0:27.000	..	..	12345678	..	..	12345678	.B.D	....	A.C.	1.345.78	37	55
0:28.000	37	..	12.456.8	37	..	12.456.8	.B.D	....	A.C.	1..45..8	..	00
0:31.000	37	..	12.456.8	..	37	12.456.8	.B.D	....	A.C.	1..45..8	..	00
0:38.000	..	37	12.456.8	..	..	12345678	....	.B.D	A.C.	1..45..8	48	44

Analysis Complete 9:28 PM 7/20/98

Start MaxTime Exploring - C:\Wl... NEMA Control... tab4.bmp - Paint 9:28 PM

Fig 10. The RawTest Data form

## Reduced Data Tab

The reduced data tab shows the actual events and times extracted from analyzing the raw data. Each row represents a timed event. Certain events shown in the reduced data that are applied across all tests are not shown on the scorecard if they meet the criteria. For a list of these tests, see Scoring a Test.

Analysis of Data Collected					
Test	Test Type	Phase	Lap	Result Category	Value
Flasher Signal Duty Cycle	FUNC08	0	1	Flash On (mS)	500
Flasher Signal Duty Cycle	FUNC08	0	1	Flash Off (mS)	500
Flasher Signal Duty Cycle	FUNC08	0	1	Flash Period (mS)	1000
Flasher Signal Duty Cycle	FUNC08	0	1	Duty Cycle (%)	50
Flasher Signal Transition	FUNC09	0	1	Transition Lead/Lag (degrees)	14
Overlaps	FUNC10	0	1	Implicit Result From Tester	FAILED
Veh. Timings/Storage, Min Recall	TIME01	9	1	Dual Indication Overlaps (mS)	3000
Veh. Timings/Storage, Min Recall	TIME01	9	1	Dual Indication Overlaps (mS)	3000
Veh. Timings/Storage, Min Recall	TIME01	1	1	Short Yellow Timing (mS)	3000
Veh. Timings/Storage, Min Recall	TIME01	1	1	Short Yellow Timing (mS)	3000
Veh. Timings/Storage, Min Recall	TIME01	2	1	Short Yellow Timing (mS)	3000
Veh. Timings/Storage, Min Recall	TIME01	3	1	Short Yellow Timing (mS)	3000
Veh. Timings/Storage, Min Recall	TIME01	4	1	Short Yellow Timing (mS)	3000

Analysis Complete 9:36 PM 7/20/98

Start MaxTime Exploring - A:\ NEMA Control... tab5.bmp - Paint 9:36 PM

Fig 11. The Reduced Data form

## Score Card Tab

The Score Card tab allows the user to compare the reduced data against a criteria file. The report that is generated is called the Score Card, and is set up and viewed in the Score Card form, as shown in Figure 12 below. On the left-hand side of the form is an index to the report, by test number, that allows the user to see the desired part of the report with a mouse-click. For detailed instructions, see the Scoring a Test section, which follows the next section, Loading an Existing Test.

Generate Report Against the Following Criteria:

Preview

Phase	Tested Value	Value	High Criteria	Low Criteria	Result
0	Implicit Result From Tester	FAILED			Failed
Test: TIME01 (Veh. Timings/Storage, Min Recall) - Lap #1					
Phase	Tested Value	Value	High Criteria	Low Criteria	Result
1	Short Yellow Timing (mS)	3000	3500	2500	Passed
1	Short Yellow Timing (mS)	3000	3500	2500	Passed
2	Short Yellow Timing (mS)	3000	3500	2500	Passed
3	Short Yellow Timing (mS)	3000	3500	2500	Passed
4	Short Yellow Timing (mS)	3000	3500	2500	Passed
5	Short Yellow Timing (mS)	3000	3500	2500	Passed
5	Short Yellow Timing (mS)	3000	3500	2500	Passed
6	Short Yellow Timing (mS)	3000	3500	2500	Passed
7	Short Yellow Timing (mS)	3000	3500	2500	Passed
8	Short Yellow Timing (mS)	3000	3500	2500	Passed
A	Dual Indication Overlaps (mS)	3000	1000	1000	Failed
A	Dual Indication Overlaps (mS)	3000	1000	1000	Failed

Analysis Complete 9:54 PM 7/20/98

Fig 12. The Score Card form

## Loading an Existing Test

Loading an existing test allows the user to reload data from a test session executed in the past and have the NCT1 software re-analyze the data and perform a comparison to a criteria set, if desired. Since the database only records the Raw Data, this function must be performed whenever a report based on old data is needed. Loading an existing test is an alternative to running a new test and is done from the File menu (present on every tabbed form).

The Load Data dialog contains a list of every test stored in the database. To select a test, highlight the desired test in the list and choose open. To filter the list use the filter query at the lower part of the screen. These filter conditions will shorten the list of available tests to load.

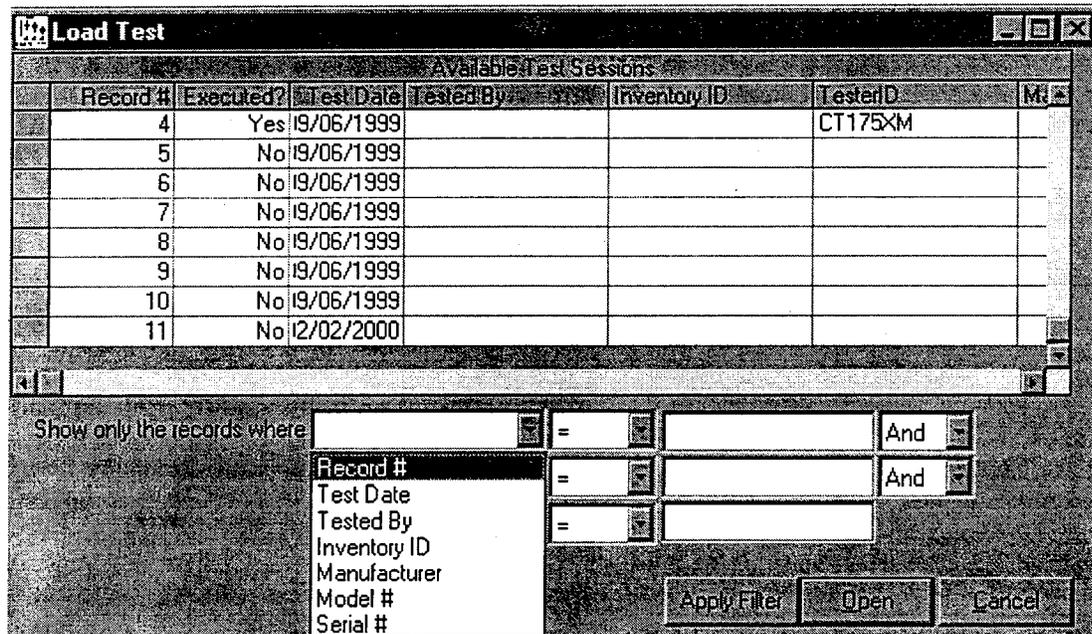


Fig 13. The Load Test form

### To load an existing test:

- Select the Load Test menu entry from the File menu.
- After selecting the test you wish to load the system will display the test data on the Test Setup, Raw Test Data, and Reduced Data forms.
- After the test is loaded into NCT1 software you may then view the timed events and times using the Reduced Data tab or score the test using the Score Card Tab.

## Scoring a Test

When the test is complete, or the desired existing test data has been loaded, the system analyzes the tester output turning the raw data into distinct timed events. These events vary according to the test selected. After the data is reduced, you may now apply a set of criteria against the events and produce a report that determines whether these events are within the specifications you set in the criteria. To do this, after reducing the data, go to the Score Card tab. Select the criteria set from the drop down menu. To see the definition of the criteria file, choose the View Criteria button. For help on creating criteria, see Creating Criteria Files.

Once you have chosen the Criteria, select the Generate Report button. The NCT1 Software will display an onscreen report that displays all the results. The Report is ordered in the format of Lap, Test, Criteria. So for example it will display:

Lap 1 Timing Test 1, Green Timing

Lap 1 Timing Test 1, Yellow Timing

Lap 1 Timing Test 1, Red Timing

Lap 1 Timing Test 2, Result...

Lap 2 Timing Test 1, Green Timing

Lap 2 Timing Test 1, Yellow Timing

Lap 2 Timing Test 1, Red Timing

Lap 2 Timing Test 2, Result...

Etc..

**To Print the Report:**

Select the Print button on the bottom of the screen.

**To Export the Report to a Specific File Format:**

Select the Save As button on the bottom of the screen. You may export the test in a variety of formats including Comma, Tab, or space delimited, or as HTML, Microsoft Word, Microsoft Excel, Text, RTF and Lotus 123. Once the report has been saved to a file, you may open it and print with your preferred application.

**Interpreting the Scorecard For a Function Test**

A given line in the report looks like the following for a function test

**Test:** FUNC07 (Vehicle Call Recognition Voltage) - Lap # 1

Phase	Tested Value	Value	High	Criteria	Low	Result
0	Implicit Result From Tester	PASSED	0		0	Passed

**Test:** FUNC08 (Flasher Signal Duty Cycle) - Lap # 1

Phase	Tested Value	Value	High	Criteria	Low	Result
0	Flash On (mS)	500	600		400	Passed
0	Flash Off (mS)	500	600		400	Passed
0	Flash Period (mS)	1000	1100		900	Passed
0	Duty Cycle (%)	50	55		45	Passed

The phase for function tests is 0 since the test applies to the controller as a whole and isn't phase specific.

When the tested value indicates Implicit Result from tester, this means that the tester determines the result and there is no corresponding value in the criteria file.

## Interpreting the Scorecard For a Timing or Verify Test

The various timing test reports appear similar to the one shown below.

### Timing Tests

Test: TIME01 (Veh. Timings/Storage, Min Recall) - Lap # 1

Phase	Tested Value	Value	High	Criteria Low	Result
1	Min Green Timing (mS)	7108	7100	6900	Failed
1	Yellow Duration Timing (mS)	3042	3100	2900	Passed
1	Red Clear Timing (mS)	9	1100	900	Failed
2	Min Green Timing (mS)	7116	7100	6900	Failed
2	Yellow Duration Timing (mS)	3050	3100	2900	Passed
2	Red Clear Timing (mS)	1017	1100	900	Passed
3	Min Green Timing (mS)	7117	7100	6900	Failed
3	Yellow Duration Timing (mS)	3050	3100	2900	Passed
3	Red Clear Timing (mS)	1017	1100	900	Passed
4	Min Green Timing (mS)	7117	7100	6900	Failed
4	Yellow Duration Timing (mS)	3050	3100	2900	Passed
4	Red Clear Timing (mS)	1016	1100	900	Passed
5	Min Green Timing (mS)	7108	7100	6900	Failed
5	Yellow Duration Timing (mS)	3042	3100	2900	Passed
6	Min Green Timing (mS)	7116	7100	6900	Failed

Overlaps are indicated by their letter in the phase column (A-D). The value is the actual timing recorded from the controller. The High Low values are from the criteria file.

Certain analyses are performed system wide and are not specific to a particular test. Whenever these conditions exist, it is considered an error and the condition is reported.

These Error Conditions include:

- Dual Indication Vehicle: If two vehicle indications are on at the same time for any single phase, the condition is timed and reported.
- Dual Indication Peds: If two ped indications are on at the same time for any single phase, the condition is timed and reported.
- Dual Indication Overlap: If two overlap indications are on at the same time for any single overlap phase, the condition is timed and reported.
- Red Fail Overlap: If no vehicle signals are active, the condition is timed and reported.
- Red Fail Vehicle: If no vehicle signals are active, the condition is timed and reported.
- Sequence Failure: If the vehicle signals are activated out of the correct sequence an error message is generated.

The Yellow Timing measurement occurs during any timing test, but is only reported as an error when the measured value is outside the criteria used for the Score Card report.

- Yellow Timing: Yellow signal times are measured and reported. If the value is outside the criteria used in the ScoreCard report, the condition is reported as an error.

# Creating Criteria Files

The criteria are used to evaluate the reduced data and determine whether a specific event is within an acceptable range determined by the user.

To create a criteria file, choose Manage Criteria Specs from the Analysis menu. The Scoring Criteria form will come up (Figure 14).

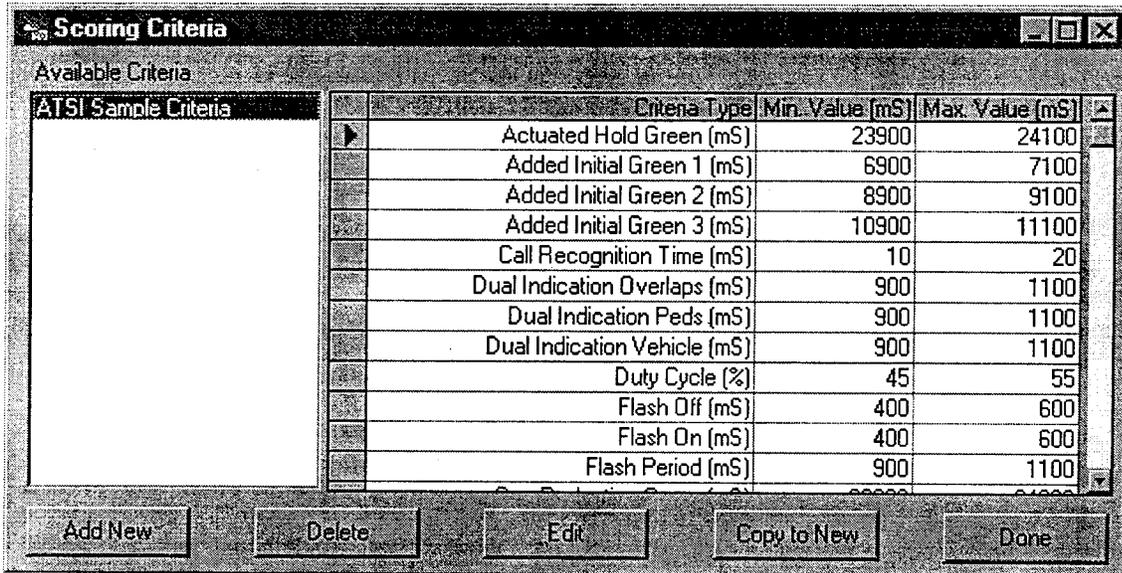


Fig 14. The Scoring Criteria editing form

The list on the left indicates the name of all the criteria files stored in the system. To create a new criteria file you may either select the Add New button or select an existing criteria set and choose the Copy to New button. The system will prompt you for a new name for the criteria. Once you have entered a name, the system will either create a criteria file with zeros in all the fields if you selected Add New or the values from the copied criteria.

To enter in the new values or edit an existing criteria, select the Edit button and begin entering in the desired values. When you are done updating the criteria, Select the Done Button.

# Exporting and Importing Specific Tests

The system allows for the transfer of a specific test from machine to machine or database to database through the Import Test and Export Test methods. To export a test, first load it into the system using the Load Test Menu entry of the File Menu. Then choose Export Test from the file menu, the system will prompt you to select a location to save the file. The system will create a specially formatted ASCII file that can be read in by the software. To import a file, choose the Import File from the File menu. The system will prompt you for the name of the file to import then load the file and create a new test session for it and save it. It will then re-analyze the data since only the actual tester output is exported. You may then use the Score Card tab to grade the test results.

# Test Modes

## Normal Mode

This mode is supported by all the tester's automated data reduction and analysis functions, and is the required mode for certification of a controller's performance to the NEMA Standards. The controller must be programmed to the Special Test Program (described in the Timing Tests section on page A26 of this manual) to work properly with the Normal Mode tests.

- \* A series of 10 Functional tests can be performed to verify non-channel related functions (External Restart, 24 V Output, Call Recognition Timing, etc.) conform to NEMA Standards.
- \* A series of 13 Timing Tests can be performed in which the tester records all signal change events (Red, Green, Yellow, Walk, Ped Clear, Don't Walk) and changes in the controller's status bits with a precise time-stamp.
- \* With every output record, the state of all signals, including Ped Clear states and overlaps, pending vehicle calls, phase next, and status bits is recorded.
- \* The sequential record of the states of the controller outputs is available for interpretation by the signal technician, and can be analyzed by the tester's software to compute the actual times of events (such as Green extension) controlled by the controller.
- \* The event timings computed by the tester software can be compared to a set of criteria, established by the user, that determines if the measured times are deemed passing or failing.

## Diagnostic Mode

This mode is supported by most of the tester's automated data reduction and analysis functions, and provides some different information from the Normal mode to help the signal technician pinpoint the problem that is detected during a Normal Mode testing session. The controller must be programmed to the Special Test Program (described in the Timing Tests section on page A26 of this manual) to work properly with the Diagnostic Mode tests. A Diagnostic Mode test report can become lengthy, because every call placed to the controller from the tester will cause another data record to be generated.

- \* A series of 10 Functional Tests can be performed, same as in the Normal Mode.
- \* A series of 13 Timing Tests can be performed in which the tester records all signal change events (Red, Green, Yellow, Walk, Ped Clear, Don't Walk) and changes in the controller's status bits with a precise time-stamp. In addition, each Ped call and Vehicle call is recorded with a time-stamp.
- \* With every output record, the state of all signals (minus overlaps), Ped Clear states, pending Vehicle calls, pending Pedestrian calls, phase next, and status bits are recorded. Overlaps are omitted to provide space to report vehicle calls in a readable format.
- \* The sequential record of the states of the controller outputs is available for interpretation by the signal technician, and can be analyzed by the tester's software to compute the actual times of events (such as Green extension) controlled by the controller.
- \* The event timings computed by the tester software can be compared to a set of criteria, established by the user, that determines if the measured times are deemed passing or failing.

## Verify Mode

This mode is supported by the tester's automated data reduction functions, but not by the comparison functions. This mode will test the controller that is running a "street" program, and allows manual verification of the timings that the controller generates prior to operating in the cabinet. Since the controller had to be programmed to a Special Test Program for testing the controller's conformance to NEMA, the tester also offers this mode to verify the "street" program that will be re-installed after the performance test is passed. This mode of operation can be used to verify and document the controller's timing for any number of other reasons as well.

\* All controller phases must be set to "Memory" or "Locking" mode to retain vehicle calls after the call drops. Leaving any phases in "Non-Locking" or "Recall" mode could result in erroneous test results.

\* No Functional tests are performed. The results of Functional Tests performed with the Special Test Program are valid with the "street" program installed.

\* A series of 6 Timing Tests can be performed in which the tester records all signal change events (Red, Green, Yellow, Walk, Ped Clear, Don't Walk) and changes in the controller's status bits with a precise time-stamp.

\* The sequential record of the states of the controller outputs is available for interpretation by the signal technician, and can be analyzed by the tester's software to compute the actual times of events (such as Green extension) controlled by the controller.

## Test Descriptions

There are 2 Categories of tests, Functional and Timing.

### FUNCTIONAL TESTS

The functional tests include tests where timing in a phase is of no particular significance, and tests which are unrelated to the controller's progression thru phases. Functional tests are not repeated for all phases. Each functional test is run as a single test and is started from a red-rest condition.

#### **FUNC 01. Line-power interrupt.**

From red-rest, the tester interrupts the AC line to controller for 450mS, then for 1050mS. Controller should remain in red-rest following the 450 ms interrupt, but should execute a normal restart sequence on the interrupt of 1050mS. One of three test results is reported in text: pass, restarts too early (goes into restart with 450mS), restarts too late (didn't go into restart with 1050mS). NEMA TS1 states: Restart on 1000 mS power interrupt, ignore 500mS power interrupt.

#### **FUNC 02. 24VDC Output Level.**

Measure 24v signal from controller; reports actual voltage to 0.1V, and pass or fail. NEMA TS1 states: Voltage must be > 22 VDC and < 26 VDC.

#### **FUNC 03. Controller voltage monitor.**

Sense logic state of CVM line from controller, reports state in terms of controller's indicated status, and pass or fail. NEMA TS1 states: Controller shall maintain a valid CVM output state when no faults are present.

#### **FUNC 04. External restart.**

From red-rest, the tester applies a vehicle call to phase 2, and waits for green 2 to appear. Then the tester asserts an external restart signal (0.1sec) and verifies that green jumps to phase 1, and that vehicle and pedestrian calls are placed to all available phases (except active phase 1). Reports pass or fail result.

#### **FUNC 05. Flashing don't-Walk.**

From red-rest, the tester asserts a pedestrian call to phase 1, then waits for pedestrian clear on phase 1. Monitoring the don't-walk on phase 1, the tester waits for a rising edge, then waits for a falling edge, then starts a timer. When the timer reaches 550mS, it tests the state of don't-walk. If now high, it has executed the half-cycle and returned. This is a passing result, else fail. Reports a pass or fail result.

#### **FUNC 06. Input Signal duration for recognition.**

The tester applies a red-rest to both rings, and waits for controller to reach red-rest. Tester asserts a 5mS vehicle call to phase 1. Wait 500mS for controller to respond. If no response, assert a 10mS vehicle call. Again wait 500mS for response. Repeat this pattern by 5mS increments (up to 50mS max) until the controller responds to the input. Report the signal duration at which call was first detected, or ">50mS" if not detected by that time (NEMA TS1 states 30mS max.). Errors are reported if the time exceeds 30mS.

#### **FUNC 07. Signal voltage level for recognition.**

The tester applies a red-rest to both rings, and waits for controller to reach red-rest. Tester asserts and holds a 16.5V signal to phase 1 vehicle call. Tester waits 500mS and verifies that the controller is still in red-rest. Tester asserts and holds a 7.5V signal to phase 1 vehicle call. Tester waits 500mS and verifies that controller is in phase 1 green. Reports pass if both tests pass, fail otherwise. NEMA TS1 states that signals must be recognised as on at less than 8V, and off at greater than 16V.

#### **FUNC 08. Flasher signal from controller.**

Measure and report the duration of the flasher's high signal, and of flasher's low signal. These should add to 1000mS, with asymmetry of 20mS or less. Tester reports on-time, off-time, period, and duty cycle of flasher output and indicates pass or fail.

#### **FUNC 09. Flasher transitions vs AC line crossings.**

Measure and report the phase difference between transitions of the flasher signal and the zero-crossings of the AC line. Flasher transitions should be within 5 degrees of a line crossing. Result is reported in degrees of phase angle and pass or fail.

#### **FUNC 10. Overlap Functions**

The tester cycles the controller through all the available phases and records the presence of the overlap outputs present at each phase and compares this to the expected overlap matrix. Any discrepancies cause a failure. Results are reported as pass or fail. The controller's overlaps must be set to the Special Test Program settings for this test (see below).

# TIMING TESTS

The Timing test group requires that a Special Test Program be installed in the controller under test. The controller's Special Test Program uses phase timing values that are selected to allow reasonably fast and comprehensive testing, and may not reflect values used in typical installations. Every phase receives the same tests and has the same settings. Single-ring controllers are tested in sequence starting from phase 1, i.e., 1-2, 1-2-3, 1-2-3-4, depending on the number of phases present or selected. For eight-phase controllers, both rings are tested simultaneously, i.e., 1&5-2&6-3&7-4&8. Some tests require multiple loops thru the phases to utilize conditions set up in earlier loops. All tests start from red-rest in all active phases.

## Controller Special Test Program Parameters:

Min Green, 7sec; Yellow clear, 3sec,; Red clear, 1 sec

Walk, 3 sec; Pedestrian clear, 7 sec; Passage, 5 sec

Max1, 20 sec; Max2, 25 sec; Red Revert, 2 sec

Max Initial, 11 sec; Added Initial, 1 sec/act;

TBR, 25 sec; TTR, 8 sec; Min gap, 2 sec

CNA1, phases 1,3,5,7; CNA2 phases 2,4,6,8

Phases 1 green and 5 green are starting phases, normal progression.

## Overlap Settings:

### For 3 and 4 phase controllers:

Overlap A: Phases 2 and 3

Overlap B: Phases 1 and 3

Overlap C: Phases 1 and 2

Overlap D: Phases 1 and 4

### For 8 phase controllers

Overlap A: Phases 1 and 8

Overlap B: Phases 6 and 7

Overlap C: Phases 4 and 5

Overlap D: Phases 2 and 3

**Note:** In all test descriptions to follow, any phase indicated (1,2,3...) also implies the corresponding Ring 2 phase, if present.

#### **TIME 01. Vehicle Timings and Storage, and Minimum Recall.**

From red-rest, start with a vehicle call to phase 1, then place minimum recall. Let the controller run to completion (back to red-rest condition) and verify that all phases are used with a final phase 1. The Tester will measure Minimum green, yellow, and red clear timings for all phases.

#### **TIME 02. Pedestrian Timings and Storage, Omit Red-Clear.**

From red-rest, start with a pedestrian call to phase 1, then place pedestrian calls to remaining phases, along with omit-red-clear to both rings. During phase 2 green, assert a pedestrian call to phase 1, so that storage of the call will be verified. Tester verifies that the controller runs through all phases, returning to red-rest after a second pass through phase 1. Tester verifies that red-clear interval is omitted for all phases, and measures the Walk and Pedestrian Clearance times for each phase.

#### **TIME 03. Passage Time and Red Revert.**

From red-rest, start with a vehicle call to phase 1, then apply vehicle calls to phases 1 and 2, so that the call to phase 1 is maintained. The phase 1 call is dropped as phase 1 enters the extension interval. As phase 2 green is applied by the controller, a vehicle call is applied to cause a green extension. This pattern of calls is repeated for all phases, while the tester measures the green times for each phase. On the second cycle, red-rest is asserted with no conflicting call, so phase 1 executes a min green/yellow/red-rest sequence. After 0.5 sec in red-rest, a recall to phase 1 is issued. Since red-revert is set to 2.0 sec, an additional 1.5 sec in red should be observed before phase 1 can go to a green signal. This pattern is repeated for the remaining phases and the red-revert intervals are measured.

#### **TIME 04. Holds and Force-Offs in Actuated Phases.**

Start with vehicle call to phase 1; assert holds to all phases. Assert vehicle call to next phase at 0.2 sec, wait for active phase to go into green rest at 10 sec. At 12 sec, assert force-off (0.1 sec). Verify that force-off to current ring advances phase properly, and leave recall to that phase. Repeat for other phases. On a second loop, instead of force-off, remove the hold for 0.1 sec, verify phase is advanced with no recall.

#### **TIME 05. Non-Actuated Hold and Force-Off.**

Start with pedestrian call to phase 1; assert CNA1 and CNA2 at 0.2 sec. to provide continuous pedestrian and vehicle calls to all phases. Mask competing calls by placing inactive phases in phase-omit. When green-rest is reached, assert conflicting call by removing phase-omit from next phase, should advance to next phase. Repeat around loop. On second loop, apply hold at 0.2 sec, wait for walk-rest at 3 sec, apply conflicting call at 5 sec by removing phase-omit from phase 2. Verify no phase advance to green-rest until force-off (0.1 sec) is asserted at 7 sec. Repeat around the loop.

#### **TIME 06. Non-Actuated Pedestrian Recycle and Walk-Rest-Modifier.**

Start with pedestrian call to phase 1. At 0.2 sec, assert CNA1 and CNA2 to provide continuous pedestrian and vehicle calls to all phases. Mask competing calls by placing all inactive phases in phase-omit. When phase 1 green-rest is reached, assert pedestrian-recycle, verify new walk interval and turn off pedestrian-recycle. When

green rest is reached again, assert walk-rest-modifier, verify new walk interval, verify walk-rest at 4 sec. Assert vehicle call to phase 2 by removing phase omit from phase 2. After phase 2 is active, drop walk-rest modifier and remove phase 1's competing call by omitting phase 1. Repeat above control sequences in phase 2. Repeat around the loop. Verify conformance from timings and sequences.

#### **TIME 07. Volume Density 1: Added Initial.**

Start with vehicle call to phase 1, apply five calls at half-second intervals to phase 2 while phase 1 is active. While phase 2 is active, apply five calls to next phase, etc. When phase 1 comes around again, apply 9 calls to phase 2, etc. When phase 1 comes around again, apply 13 calls to phase 2, etc. Five calls should produce an init green of 7 sec (min-green), 9 calls should produce an init green of 9 sec (1 sec/actuation), and 13 calls should produce an init green of 11 sec (max init). Requires three loops plus a terminating phase 1.

#### **TIME 08. Volume Density 2: Gap Reduction.**

Start with vehicle call to phase 1. At 0.2 sec, inhibit max, add continuous vehicle call to phase 2. Assert vehicle calls to phase 1 at 6.5 sec, 11.1 sec, 15.7 sec, 20.3 sec, 24.9sec (TBR=25) and 28.9sec, while maintaining phase 2 call. Green should drop about 32sec (TTR=6, MinGap=2) into the phase. Repeat around loop. Verify green duration from timings.

#### **TIME 09. Vehicle Extensions and Max-Inhibit.**

Start with vehicle call to phase 1, assert max-inhibit and vehicle call to phase 2. Apply same calls to active phase as in Test 08 above, but only one call (not continuous occupancy) to competing phase, so TBR timer should not run. Expect green drop about 34sec into the phase. Repeat around loop. Note timing of green drop to verify extension.

#### **TIME 10. Green Extension, Max 1.**

Start with vehicle call to phase 1, assert conflicting vehicle call to phase 2 at 0.2sec to start max timer. Starting at 6 sec, assert four calls at four sec intervals (extend to 23 sec), verify max-out at 20 sec (max 1). Repeat around the loop. Max-out termination should leave recalls.

#### **TIME 11. Green Extension, Max2.**

Similar to Test 10 but assert max 2 along with competing call at 0.2sec. Starting at 6 sec, assert five calls at four second intervals (extend to 27 sec), verify max-out at 25 sec. Repeat around the loop. Check for recalls.

#### **TIME 12. Stop-Time and Interval Advance.**

Start with vehicle call to phase 1, assert stop-time at 6 sec (in min green). Apply vehicle call to phase 2 at 8 sec, release stop-time at 10 sec. Verify that min green was extended by four sec, verify storage of call during stop-time by testing advance to phase 2. Repeat around the loop. On second loop, assert stop-time at 6 sec, verify immediate advance to yellow on interval-advance at 8 sec, immediate advance to red-clear on interval-advance at 10 sec. Release stop-time, allow red-clear to time-out and advance next phase. Repeat around the loop.

**TIME 13. Manual enable, Interval Advance, and Pedestrian Omits.**

Start with pedestrian call to phase 1, assert manual enable at 0.2 sec. At 3 sec, assert (0.1 sec) interval advance to pedestrian clear. Assert interval advance again at 5 sec, advance to yellow, where it will continue to terminate the phase and advance to phase 2. Repeat around the loop. On the second loop, assert and hold pedestrian omits on all phases, verify that interval advance at 3 sec in green advances directly to yellow, and the phase terminates and advances with no further inputs. Repeat around the loop.

